Software Libre y Educación:

servicios de red, gestores de contenidos y seguridad

José Ángel Bernal, Fernando Gordillo, Hugo Santander y Paco Villegas

18 de octubre de 2005

Índice general

I Redes TCP/IP **17**

1. Introducción al Curso **19**
1.1. apt . . . 19
1.1.1. /etc/sources.list . . . 20
1.1.2. apt-get . . . 21
1.1.3. Desinstalando paquetes . . . 22
1.1.4. Eliminando archivos de paquete no utilizados . . . 22

2. Redes de Ordenadores **25**
2.1. Modelo de Referencia OSI . . . 25
2.1.1. Capa de Aplicación . . . 28
2.1.2. Capa de Presentación . . . 28
2.1.3. Capa de Sesión . . . 28
2.1.4. Capa de Transporte . . . 28
2.1.5. Capa de Red . . . 29
2.1.6. Capa de Enlace de Datos . . . 29
2.1.7. Capa Física . . . 30
2.2. Comunicación entre capas . . . 30
2.2.1. Tipos de Servicios . . . 31

3. Conjunto de Protocolos TCP/IP. **33**
3.1. Niveles de la Arquitectura TCP/IP . . . 33
3.1.1. Nivel de Aplicación. . . . 34
3.1.2. Nivel de Transporte. . . . 34
3.1.3. Nivel de Red. . . . 35
3.1.4. Nivel de Enlace o de Acceso a la red. . . . 35
3.1.5. Nivel Físico. . . . 36
3.2. Añadiendo cabeceras y colas . . . 36

4. TCP/IP: desde los pulsos hasta los datos **37**
4.1. Nivel Físico . . . 37
4.1.1. Ethernet e IEEE 802.3 . . . 37
4.2. Nivel de Red . . . 40
4.2.1. Direccionamiento IP . . . 40
4.2.2. Protocolo ARP . . . 44
4.2.3. Protocolo IP . . . 46
4.3. Nivel de transporte: TCP . . . 49
4.3.1. Puertos y Sockets . . . 49
4.3.2. Protocolo TCP . . . 50
4.4. Nivel de Aplicación . . . 54
4.4.1. Protocolo HTTP . . . 54
4.5. Ver para creer: Ethereal . . . 55

4.5.1. tcpdump . . . 56
4.5.2. Arrancando ethereal . . . 58
4.5.3. Capturando paquetes . . . 59
4.5.4. Filtrado durante la captura . . . 61
4.5.5. Visualización de los datos capturados . . . 61
4.5.6. Filtrado durante la visualización . . . 63
4.5.7. Capturando una sesión telnet . . . 64
4.5.8. Estadísticas acerca de la captura . . . 66
4.6. Monitorizacion de red con EtherApe . . . 67
4.6.1. Instalación . . . 68
4.6.2. Configuración . . . 68
4.6.3. Funcionamiento . . . 69

5. Conectando al mundo exterior 71
5.1. Routing o encaminamiento IP . . . 71
5.1.1. Estático y dinámico . . . 74
5.2. Vale, pero yo quería un curso de Linux. . . . 74
5.2.1. Activando interfaces . . . 75
5.2.2. Estableciendo rutas . . . 77
5.2.3. Resolución de nombres. . . . 78
5.3. Configuración gráfica . . . 78
5.3.1. Con Fedora . . . 78
5.3.2. Con Guadalinex (Debian) . . . 82
5.4. Conexión a Internet: RTB y ADSL. . . . 85
5.4.1. Conexión con módem . . . 85
5.4.2. ADSL . . . 94

6. Linux como Router y Cortafuegos 99
6.1. Router Linux . . . 99
6.2. Cortafuegos Linux . . . 100
6.2.1. Clasificación de cortafuegos . . . 100
6.2.2. Terminología de cortafuegos . . . 100
6.2.3. Arquitecturas de cortafuegos . . . 101
6.2.4. Firewalls sobre linux . . . 102
6.2.5. ¿Qué es iptables? . . . 104

7. Configuración de DHCP 113
7.1. Introducción . . . 113
7.2. Instalación . . . 115
7.3. Configuración . . . 115
7.3.1. De la máquina Linux . . . 115
7.3.2. Configuración de los clientes: . . . 119

8. IPv6 123
8.1. Introducción histórica . . . 123
8.1.1. ¿Cómo son las direcciones IPv6? . . . 124
8.1.2. Tipos de direcciones . . . 125
8.1.3. ¿Estamos preparados para IPv6? . . . 128

Prácticas 133

II Linux como servidor 139

9. Demonios y Superdemonios 141
9.1. inetd . . . 142
9.2. xinetd . . . 144
9.3. Parando y arrancando demonios . . . 147
9.3.1. Debian . . . 147
9.3.2. Fedora . . . 148
9.3.3. Algunos servicios de red usuales . . . 150
9.4. TCP-Wrappers . . . 151
9.4.1. Reglas de acceso . . . 152

10. Terminal remoto. Telnet y SSH 155
10.1. Visión general . . . 155
10.1.1. Acceso remoto: telnet . . . 155
10.1.2. Copia remota: ftp . . . 157
10.1.3. Una solución más segura . . . 161
10.2. La Criptografía llega en nuestra ayuda . . . 163
10.3. SSH como cliente . . . 166
10.3.1. Sesiones remotas con SSH . . . 166
10.3.2. Autenticación por clave criptográfica . . . 168
10.3.3. Y ahora que tenemos las claves ... ¿qué hacemos con ellas? . . . 170
10.3.4. El agente ssh . . . 171
10.3.5. Uso del agente SSH en GNOME . . . 172
10.4. Configuración del servidor SSH . . . 173
10.4.1. Instalación . . . 173
10.4.2. Configuración . . . 174

11. Servidor de nombres DNS 177
11.1. ¿Qué necesito del DNS? . . . 178
11.2. Recursos del Servidor de Nombres . . . 178
11.3. Servidores de Nombres . . . 182

12. Servicio de Directorio LDAP 187
12.1. Estructura del Directorio . . . 188
12.2. Servidor OpenLDAP . . . 190
12.2.1. Configuración del Servidor OpenLDAP . . . 190
12.3. Clientes LDAP . . . 195
12.4. Caso práctico - LDAP . . . 198
12.4.1. Crear un directorio para autenticación . . . 199
12.4.2. Configuración de Name Service Switch . . . 200
12.4.3. Módulos PAM . . . 202
12.4.4. nscd . . . 203

13. Compartir impresoras:Cups 205
13.1. Introducción . . . 205
13.2. Instalación . . . 206
13.2.1. Fedora . . . 207
13.2.2. GuadaLinex . . . 207
13.3. Configuración de CUPS . . . 208
13.3.1. client.conf . . . 209
13.3.2. cupsd.conf . . . 209
13.4. Interfaz Web . . . 218
13.4.1. Añadir una impresora . . . 219

13.4.2. Añadir una clase . . . 223
13.5. Un poco de comandos . . . 224
13.5.1. lpadmin . . . 225
13.6. ➥ Para Practicar . . . 226

14.Samba **229**
14.1. ¿Qué es Samba? . . . 229
14.2. Instalación . . . 231
14.2.1. Fedora . . . 231
14.2.2. Debian . . . 232
14.2.3. Programas . . . 232
14.3. Configuración . . . 233
14.3.1. Configuración de las máquinas Güindows . . . 233
14.3.2. Configuración de la máquina Linux . . . 234
14.3.3. Swat . . . 243
14.4. A "bailar" la Samba . . . 244
14.4.1. Acceder desde una máquina Linux a una Windows . . . 244
14.4.2. Acceder desde Windows a la máquina Linux . . . 250

15.Servicio de compartición de ficheros NFS **255**
15.1. Servidor NFS . . . 255
15.1.1. Fichero /etc/exports . . . 257
15.1.2. RPC y portmap . . . 258
15.2. Cliente NFS . . . 259
15.2.1. Montar sistemas de archivos NFS usando /etc/fstab . . . 259

16.Servicio de Proxy-caché **261**
16.1. ¿Qué es un proxy caché? . . . 261
16.2. Squid, un proxy caché para Linux . . . 261
16.2.1. Visión general . . . 261
16.2.2. Conceptos sobre cachés . . . 262
16.2.3. Instalación . . . 262
16.3. Configuración de Squid . . . 263
16.3.1. Configuración básica . . . 263
16.3.2. Configuración de jerarquía de caché . . . 265
16.3.3. Control de acceso . . . 265
16.4. Configuración de los clientes . . . 268
16.5. Acceso a internet autenticado contra ldap . . . 270
16.5.1. Métodos de autenticación de Squid . . . 270
16.5.2. Analizador de logs SARG . . . 272
16.6. ➥ Para practicar . . . 281
16.6.1. Castellanizar los errores de Squid . . . 281
16.6.2. Limitar ancho de banda para determinadas extensiones . . . 281
16.6.3. Proxy transparente . . . 283
16.7. DansGuardian . . . 283
16.7.1. Funcionamiento . . . 284
16.7.2. Instalación . . . 284
16.8. Configuración . . . 284

Prácticas **287**

III Servidor Web y Correo electrónico 293

17.Servidor Web Apache 295
17.1. Servidor Web (apache). . . . 295
17.2. Instalación . . . 297
17.2.1. Red Hat/Fedora . . . 297
17.2.2. Guadalinex (Debian) . . . 298
17.2.3. ¡A navegar! . . . 300
17.3. Configuración . . . 302
17.3.1. Document root . . . 302
17.3.2. Ficheros de configuración . . . 303
17.4. /etc/http/httpd.conf . . . 305
17.4.1. Debian . . . 309
17.5. Autenticación . . . 311
17.6. Host Virtuales . . . 314
17.7. Servidores Seguros . . . 316
17.7.1. Autenticación del cliente mediante certificados . . . 322
17.8. Reescribir las URL . . . 325
17.9. Loganalizadores . . . 326
17.9.1. webalizer . . . 326
17.9.2. awstats . . . 328

18.Correo electrónico 333
18.1. Introducción . . . 333
18.1.1. ¿Cuántos invitados tenemos para cenar? . . . 335
18.1.2. ¿Cómo se encamina el correo? . . . 337
18.1.3. Eso no es todo, aún hay más . . . 338
18.2. Agentes de Transporte . . . 341
18.2.1. Postfix . . . 341
18.2.2. Sendmail . . . 351
18.3. Agente de entrega: Fetchmail . . . 358
18.3.1. Configuración . . . 359
18.4. Mozilla Mail y Ximian Evolution . . . 362
18.4.1. Mozilla Mail . . . 362
18.4.2. Agente de Usuario: Ximian Evolution . . . 365
18.5. Luchemos contra el SPAM . . . 367
18.5.1. Instalación de SpamAssassin . . . 368
18.5.2. Instalación de Amavisd-new . . . 369
18.5.3. Modificaciones en Postfix . . . 371
18.6. Gestores de listas de correo: Mailman . . . 372
18.6.1. ¿Qué es una lista de correo? . . . 372
18.6.2. Mailman . . . 372
18.7. Correo Web: SquirrelMail . . . 378
18.7.1. Instalación . . . 378
18.7.2. Configuración . . . 380

Prácticas 385

IV Contenido dinámico 391

19.Páginas PHP 393
19.1. Introducción . . . 393
19.2. Instalación . . . 396

19.2.1. Configuración 398
19.3. Primeros pasos con php 406
19.3.1. Estructuras condicionales 410
19.3.2. Bucles 412
19.4. Ejemplos 413
19.4.1. Un ejemplo de Web con PHP 413
19.4.2. Representación gráfica de funciones con PHP 419

20.MySQL **425**
20.1. Introducción a las BDR 425
20.2. Instalación 428
20.2.1. Configuración del servidor 429
20.3. Inicio de MySQL 431
20.3.1. Aseguremos el servidor 431
20.3.2. Un poco de comandos 432
20.4. PHPMyAdmin 435
20.4.1. Instalación 435
20.4.2. ➥Base de datos cursolinux 437
20.5. PHP y MySQL: páginas web dinámicas. 447
20.5.1. Más sentencias de PHP 447
20.5.2. Un ejemplo 451

21.Moodle y PHP-Nuke **461**
21.1. Entorno virtual de aprendizaje: Moodle 461
21.1.1. Introducción. 461
21.1.2. Instalación 462
21.1.3. Primeros pasos en la administración. 468
21.1.4. Nuestro primer curso 470
21.1.5. Más configuración 473
21.2. PHP-Nuke 479
21.2.1. Introducción 479
21.2.2. Instalación de PHP-Nuke 479
21.2.3. Configuración básica del portal. 483
21.2.4. Coppermine 492

Prácticas **495**

V Administración Avanzada **501**

22.Copias de seguridad **503**
22.1. Visión general 503
22.2. Políticas de copias de seguridad 503
22.3. Dispositivos de almacenamiento 505
22.4. Utilidades de archivado 505
22.4.1. Utilidad tar 505
22.4.2. Utilidad dump/restore 506
22.5. Sincronización de sistemas de ficheros 508
22.5.1. Trabajando con rsync 509
22.5.2. Copias de seguridad con rsync 510
22.6. Copias de seguridad en CDROM 511
22.7. AMANDA 513
22.7.1. Características de AMANDA 514
22.7.2. Instalación de AMANDA 515

22.7.3. Configuración de clientes . . . 515
22.7.4. Configuración del servidor de cintas . . . 516
22.7.5. Salvaguarda de datos con AMANDA . . . 520
22.7.6. Recuperación de datos con AMANDA . . . 522

23.Logs del sistema **525**
23.1. Archivos de bitácora . . . 525
23.2. Archivos de log existentes en el sistema . . . 525
23.3. Bitácora del sistema: syslog . . . 530
23.3.1. ¿Qué podemos registrar en los ficheros de log? . . . 532
23.3.2. Acciones en respuesta a eventos. . . . 533
23.4. Gestión de los logs . . . 534
23.4.1. Registro de nuestros scripts . . . 534
23.4.2. Rotación de los logs . . . 535
23.5. Análisis de logs con logwatch . . . 536

24.Utilidades de administración **539**
24.1. Administración remota de sistemas . . . 539
24.2. ¿Por qué utilizar Webmin? . . . 539
24.3. Instalación de Webmin . . . 540
24.4. Primera toma de contacto . . . 542
24.5. Administración de Webmin . . . 544
24.5.1. Configuración de Webmin . . . 544
24.6. Un ejemplo: Apache . . . 551
24.6.1. Dónde configurar los servicios de nuestro sistema . . . 551
24.6.2. Módulo de configuración de Apache . . . 552
24.6.3. Consideraciones finales . . . 558
24.7. Gestión de varios servidores Webmin . . . 559

25.Monitorización de Sistemas **561**
25.1. Nagios . . . 561
25.1.1. ¿Qué es Nagios? . . . 561
25.1.2. Instalación de Nagios . . . 561
25.1.3. Configuración de Nagios . . . 562
25.1.4. Monitorizar un nuevo host . . . 564
25.2. Monitorización de redes con ntop . . . 568
25.2.1. Instalación . . . 569
25.2.2. Datos en ntop . . . 570

Prácticas **573**

VI Seguridad **579**

26.Blindaje del sistema **581**
26.1. Seguridad en UNIX . . . 581
26.2. Conceptos sobre seguridad . . . 582
26.3. Planificación de Seguridad . . . 583
26.4. Mecanismos de prevención . . . 584
26.4.1. Cierre de servicios innecesarios . . . 584
26.4.2. Instalación de envolventes (*wrappers*) . . . 585
26.4.3. Seguridad de las claves . . . 585
26.4.4. Seguridad de los usuarios . . . 586
26.5. SELinux . . . 590

26.5.1. ¿Qué es SELinux? . . . 590
26.5.2. Terminología SELinux . . . 591
26.5.3. Modos de uso de SELinux . . . 593

27. Vulnerabilidades del sistema **595**
27.1. Tipos de ataques y vulnerabilidades . . . 595
27.1.1. Ataques de negación de servicio (*denial of service*) . . . 595
27.1.2. Cracking de passwords . . . 598
27.1.3. E-mail bombing y spamming . . . 599
27.1.4. Seguridad en WWW . . . 600
27.2. Analizador de vulnerabilidades Nessus . . . 602
27.2.1. Instalación de Nessus . . . 603
27.2.2. Actualización de plugins . . . 605
27.2.3. Arrancando Nessus . . . 606
27.2.4. Usando Nessus . . . 607
27.3. Crackeadores de password: John the Ripper . . . 609
27.3.1. Instalacion . . . 610
27.3.2. Crackeando el fichero /etc/passwd . . . 610
27.4. Detección de intrusiones . . . 611
27.4.1. Razones para la detección de intrusiones . . . 611
27.4.2. Intentos de intrusión en el sistema (Portsentry) . . . 613
27.4.3. Integridad del sistema (Tripwire) . . . 618

28. Análisis Forense **623**
28.1. Recopilando evidencias . . . 624
28.2. Analizando datos . . . 625
28.3. Una ayuda al forense: Sleuthkit . . . 625

29. Detección de virus **629**
29.1. Virus y Troyanos en UNIX . . . 629
29.1.1. El problema de los virus . . . 630
29.2. Antivirus ClamAV . . . 630
29.2.1. Instalación . . . 630
29.2.2. Probemos la medicina . . . 633
29.2.3. Freshclam . . . 634
29.2.4. Funcionamiento . . . 634
29.2.5. Ejemplo . . . 636

Prácticas **637**

VII Apéndices 643

A. Soluciones **645**

B. httpd.conf **657**

C. Licencia de Documentación Libre GNU (traducción) **675**
C.1. GFDL . . . 675

Índice de figuras

4.1. Visualizar captura con `tcpdump` . . . 57
4.2. Pantalla de inicio ethereal . . . 59
4.3. Configuración de la captura . . . 60
4.4. Detalles de un paquete capturado . . . 62
4.5. Detalles de un paquete capturado en ventana independiente . . . 62
4.6. Creación de expresiones de filtrado . . . 63
4.7. Filtro de captura de sesión de telnet . . . 65
4.8. Captura de sesión de telnet . . . 65
4.9. Seguir la trama TCP de la sesión telnet . . . 66
4.10. Sumario de la captura . . . 67
4.11. Conversación IPv4 . . . 67
4.12. Cuadro de preferencias de Etherape . . . 68
4.13. Captura de Etherape . . . 69
4.14. Estadísticas de protocolos de Etherape . . . 70

8.1. Comparación paquete IPv4 y paquete IPv6 . . . 123

10.1. Configuración de sesión . . . 173

13.1. Herramientas gráficas de configuración . . . 206

16.1. Jerarquía de proxy . . . 262
16.2. Autenticación de Squid . . . 272
16.3. Error en la autenticación . . . 272
16.4. Página principal y listado de informes disponible . . . 280
16.5. Informes por usuario y por sitio visitado . . . 281

17.1. Estadísticas uso Apache . . . 295
17.2. Arquitectura en capas . . . 297

18.1. Instalacion Postfix - Tipo genérico de configuración . . . 343
18.2. Instalacion Postfix - Redirección correo root . . . 343
18.3. Instalación Postfix - Dominio de correo . . . 344
18.4. Instalación Postfix - Destinos para los que se acepta correo . . . 344
18.5. Instalación Postfix - Actualización síncrona cola correo . . . 345
18.6. Configuración Squirrelmail . . . 382

20.1. Tablas y phpMyAdmin . . . 444

21.1. Moodle de Mileto e IES Murgi . . . 462
21.2. Inicio Moodle . . . 467
21.3. Web del IES Murgi . . . 479
21.4. phpNuke admin . . . 482

22.1. Inicio de k3b . . . 513
22.2. Informe de copia de seguridad sin errores de AMANDA . . . 521
22.3. Informe de copia de seguridad de AMANDA . . . 521

23.1. Guadalinex, Bitácora del Sistema . . . 528
23.2. Guadalinex, Monitorizar ficheros de log . . . 528
23.3. Fedora, Registro del sistema . . . 529
23.4. Fedora, Archivos de log por defecto . . . 529
23.5. Fedora, Alarmas y Advertencias . . . 530
23.6. Fedora, Configuración de nuevos archivos de log . . . 530

24.1. Login en Webmin . . . 542
24.2. Configuración de Webmin . . . 543
24.3. Cambio de lenguaje en Webmin . . . 544
24.4. Configuración de Webmin . . . 544
24.5. Control de acceso por IP . . . 545
24.6. Configuración de dirección IP y puerto donde escucha Webmin . . . 546
24.7. Configuración de funciones de log . . . 546
24.8. Configuración de servidores Proxy . . . 547
24.9. Instalación de nuevos módulos . . . 548
24.10Creación de un nuevo usuario . . . 549
24.11Visión de Webmin del nuevo usuario . . . 550
24.12Grupos en Webmin . . . 550
24.13Asignar un grupo a un usuario . . . 551
24.14Categoría Servidores . . . 552
24.15Primera llamada al módulo Apache . . . 553
24.16Módulo gestión Apache - Configuración Global . . . 553
24.17Límites y Procesos . . . 554
24.18Redes y Direcciones . . . 554
24.19Tipos MIME . . . 555
24.20Varios . . . 555
24.21Programas CGI . . . 556
24.22Opciones por-directorio . . . 556
24.23Editar ficheros de configuración . . . 557
24.24Gestión de Servidores Virtuales . . . 557
24.25Creación de un nuevo servidor virtual . . . 558
24.26Configuración del servidor virtual . . . 558
24.27Configuración módulo gestión Apache . . . 559

25.1. Pantalla inicial de Nagios . . . 562
25.2. Monitorizar el nuevo host . . . 566
25.3. Habilitar chequeos y notificaciones . . . 567
25.4. Chequeos y notificaciones habilitados . . . 567
25.5. Forzar chequeo de un servicio . . . 568
25.6. Estado de servicio OK . . . 568
25.7. Configuración de ntop . . . 569
25.8. Inicio de ntop . . . 570
25.9. Usando ntop -About . . . 570
25.11Usando ntop - IP Summary . . . 571
25.10Usando ntop - Summary . . . 571
25.12Usando ntop - Admin . . . 572

26.1. Ejemplo de vulnerabilidad aparecida en CERT 582

27.1. Acceso de usuario en Nessus 607
27.2. Plugins de Nessus 608
27.3. Selección de objetivos 609
27.4. Informe de vulnerabilidades presentado por Nessus 609

28.1. Pantalla de inicio de Autopsy y análisis de datos 627

Índice de cuadros

19.1. Errores 400
19.2. Tipos de datos 407

20.1. Privilegios para usuarios 434
20.2. Tipos Numéricos 440
20.4. Tipos cadena 441
20.6. Tipos fecha y hora 442
20.7. Tipos `TIMESTAMP` 442

23.1. Tipos de procesos disponibles en `/etc/syslogd.conf` 532
23.2. Niveles de severidad 533
23.3. Opciones de la utilidad `logger` 535

Parte I

Redes TCP/IP

Capítulo 1

Introducción al Curso

> "Para Dan Halbert el camino hacia Tycho comenzó en la universidad, cuando Lissa Lenz le pidió prestado su ordenador. El suyo se había estropeado, y a menos que pudiese usar otro suspendería el proyecto de fin de trimestre. No había nadie a quien se atrevería a pedírselo, excepto Dan.
>
> Esto puso a Dan en un dilema. Tenía que ayudarla, pero si le prestaba su ordenador ella podría leer sus libros. Dejando de lado el riesgo de ir a la cárcel durante muchos años por dejar a otra persona leer sus libros, la simple idea le sorprendió al principio. Como todo el mundo, había aprendido desde la escuela que compartir libros era malo, algo que sólo un pirata haría."
>
> *El Derecho a Leer. Richard Stallman - GNU*

Este curso surge como una continuación lógica de los cursos sobre GNULinux convocados en años anteriores por la SAEM Thales y el CICA. En las encuestas realizadas al finalizar los mismos, siempre se ha puesto de manifiesto el interés de muchos de los participantes por ampliar los conocimientos adquiridos. El curso se dirige, por tanto, a aquellas personas que partiendo de unas nociones previas sobre Linux desean seguir profundizando en su conocimiento. Está enfocado sobre todo al profesorado y con él se pretende dar a conocer las herramientas necesarias para poner en funcionamiento un servidor de red centralizado (usando software libre), que permita compartir los recursos y la información en los centros educativos.

Antes de comenzar de lleno con los contenidos del curso debemos aclarar que las distintas entregas se enfocan bajo el uso de dos distribuciones[1]: Guadalinex 2004 (Debian) y Fedora Core 3, pudiendo los participantes optar por aquella que más les interese.

1.1. apt

> ¿Desea instalar o eliminar una aplicación? No hay problema. ¿Desea actualizar un programa que ya ha instalado? Muy fácil. Con un par de simples comandos o pulsando algunos botones, este proceso lo podrá realizar usted mismo. *Red Hat Linux 7.0: The Official Red Hat Linux Getting Started Guide*

Para garantizarnos que trabajamos siempre con la última versión disponible para los programas objeto de estudio, realizaremos la instalación de los paquetes bajo el supuesto de que estamos conectados a Internet. Si bien esto es "casi obligatorio" con Guadalinex, no lo es con Fedora ya que casi todos los paquetes comentados están en alguno de los CDs[2] que componen la distribución.

Aunque desde Fedora podemos usar la herramienta `yum` para instalar paquetes desde Internet, con el fin de homegeneizar el proceso de instalación, lo realizaremos usando los comandos en modo texto del paquete `apt`.

[1] Partimos de la base de que se dispone de un ordenador con alguna de estas distribuciones ya instalada.
[2] O en el DVD

Con los comandos del paquete `apt` instalaremos las últimas versiones de los programas, ya que siempre buscará la versión más reciente de los paquetes.

APT son las siglas de *Advanced Packaging Tool*, es decir, *herramienta avanzada de empaquetamiento*. El sistema APT es un sistema abierto, basado en la licencia GNU y desarrollado por el *APT Development Team*. Este paquete se instala por defecto en Debian (Guadalinex) y no así en Fedora. Así que antes de nada, si usamos Fedora o Red Hat debemos instalarlo.

Desde la página

`http://apt.freshrpms.net/`

podemos acceder a la última versión del programa para Fedora (o el resto de versiones de Red Hat). En la actualidad, y si trabajamos con Fedora Core 3, se trata de bajar:

```
$wget http://ftp.freshrpms.net/pub/freshrpms/fedora/linux/3/apt/apt-0.5.15cnc6-1.1.fc3.fr.i386.rpm
#rpm -ivh apt-0.5.15cnc6-1.1.fc3.i386.rpm
```

1.1.1. /etc/sources.list

Parte fundamental del funcionamiento de `apt` es el archivo en que están localizados los repositorios de paquetes. Este archivo es:

`/etc/apt/sources.list`

Su formato es similar en ambas distribuciones aunque no su contenido. En general las líneas de este fichero son del tipo:

Debian: `deb http://protocolo.site.org/debian distribución sección1 sección2 sección3`

Fedora: `rpm http://protocolo.site.org/ distribución sección1 sección2 sección3`

y para los comentarios se usa el carácter `#`.

GuadaLinex

El contenido de ese fichero en GuadaLinex es:

```
# Junta de íAndaluca (Repositorio raiz)
# éMtodo HTTP

deb http://http.guadalinex.org/debian sarge main contrib non-free
deb http://http.guadalinex.org/debian-non-US sarge/non-US main contrib non-free
deb http://http.guadalinex.org/debian-security sarge/updates main contrib non-free
deb http://http.guadalinex.org/repositorio muflon guada

# Fuentes
# deb-src http://http.guadalinex.org/debian sarge main contrib non-free
# deb-src http://http.guadalinex.org/repositorio muflon guada

# éMtodo FTP
#deb ftp://ftp.guadalinex.org/repositorio muflon main contrib non-free guada
#deb ftp://ftp.guadalinex.org/repositorio muflon/non-US main contrib non-free

# Mirror Oficial de Guadalinex: Centro áInformtico íCientfico de íAndaluca (CICA)
#deb ftp://ftp.cica.es/debian sarge main contrib non-free
#deb ftp://ftp.cica.es/guadalinex/repositorio muflon guada

# Mirror Oficial de Debian
# Sarge
#deb http://ftp.fi.debian.org/debian sarge main contrib non-free
```

```
#deb http://ftp.fi.debian.org/debian-security sarge/updates main contrib non
    -free
#deb http://non-us.debian.org/debian-non-US sarge/non-US main contrib non-
    free
```

Listado 1.1: Debian:/etc/apt/sources.list

Todas las líneas de este fichero están comentadas (están encabezadas por el símbolo "#") salvo dos. Las líneas comentadas encabezadas con `deb-scr` sólo se deben descomentar en el caso de que deseemos bajarnos los paquetes fuente.

Si bien los repositorios de Guadalinex están casi siempre al día, es preferible trabajar con las últimas versiones de los programas. Para conseguirlo debemos descomentar las líneas de los repositorios oficiales de Debian, se trata de descomentar y adecuar las últimas líneas:

```
deb http://ftp.fi.debian.org/debian sarge main contrib non-free
deb http://ftp.fi.debian.org/debian-security sarge/updates main contrib non-
    free
#deb http://non-us.debian.org/debian-non-US sarge/non-US main contrib non-
    free
#deb http://ftp.fi.debian.org/debian-non-US sarge/non-US main contrib non-
    free
```

Fedora

Con Fedora en general no tendremos que modificar nada, y con la configuración por defecto es suficiente para iniciar el curso.

En ambos

Si realizamos algún cambio en este fichero **siempre** deberemos ejecutar el comando:

```
# apt-get update
```

Los paquetes descargados son almacenados en el directorio `/var/cache/apt/archives` por si los necesitamos en algún otro momento o deseamos instalarlos en otro ordenador: ya no tenemos por qué descargarlos de nuevo.

1.1.2. apt-get

Para conocer más sobre el comando `apt` se puede consultar el documento *Apt HOWTO*: `http://www.debian.org/doc/manuals/apt-howto/index.es.html`

A partir de ahora usaremos sólo el comando `apt-get` (siempre que sea posible) y partiremos de la idea de que la lista de paquetes está siempre actualizada, es decir, que se ha ejecutado:[3]

```
# apt-get update
Get:1 http://ayo.freshrpms.net fedora/linux/3/i386 release [1990B]
Fetched 1990B in 1s (1146B/s)
Hit http://ayo.freshrpms.net fedora/linux/3/i386/core pkglist
Hit http://ayo.freshrpms.net fedora/linux/3/i386/core release
Get:1 http://ayo.freshrpms.net fedora/linux/3/i386/updates pk-
glist [287kB]
Hit http://ayo.freshrpms.net fedora/linux/3/i386/updates release
Get:2 http://ayo.freshrpms.net fedora/linux/3/i386/freshrpms pk-
glist [160kB]
Hit http://ayo.freshrpms.net fedora/linux/3/i386/freshrpms release
```

[3]En Debian los repositorios de paquetes son otros y no se corresponden con las líneas que se listan.

```
Fetched 447kB in 21s (21,2kB/s)
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
```

Para instalar un paquete escribiremos:

```
# apt-get install paquete
```

y en el caso de que surja algún problema de dependencias

```
# apt-get -f install
```

Si se daña un paquete instalado o deseamos reinstalar una nueva versión disponible del mismo, deberemos añadir la opción `--reinstall`

```
# apt-get --reinstall install paquete
```

1.1.3. Desinstalando paquetes

Para eliminar un paquete del sistema escribiremos:
`#apt-get remove nombre_paquete`
o equivalentemente
`#apt-get install nombre_paquete-`
Como ya hemos comentado, los paquetes se bajan a `/var/cache/apt/archives`, si deseamos volver a instalarlos ya los tendremos a mano. Ejecutando `apt-get` como en el ejemplo eliminaremos los paquetes, pero no así sus archivos de configuración, si es que existían. Para una eliminación completa del paquete deberíamos ejecutar:

```
# apt-get --purge remove paquete
```

1.1.4. Eliminando archivos de paquete no utilizados

Los paquetes que se instalan en nuestro sistema se bajan previamente a un repositorio de paquetes desde el que son instalados automáticamente por APT. Con el paso del tiempo, este proceso hace que el repositorio empiece a crecer y vaya ocupando mucho espacio en nuestro disco duro.

Para borrar los paquetes después de haber actualizado por completo nuestro sistema podemos ejecutar:

```
# apt-get clean
```

De esta forma se elimina la totalidad de paquetes de la caché.

Pero si no tenemos problemas de espacio, mejor optar por:

```
# apt-get autoclean
```

De este modo, sólo se eliminan de `/var/cache/apt/archives` los paquetes "inútiles", es decir, los que ya no sirven porque existe una nueva versión de los mismos.

➥ Para practicar

Si está usando un servidor proxy, primero se debe dar valor a la variable de entorno `http_proxy`. Si trabajamos en modo consola (se puede definir en modo gráfico desde Gnome o KDE) y deseamos definir la variable de entorno `http_proxy`, podemos usar:

```
export http_proxy="http://ip_proxy:puerto"
```

Por ejemplo

```
export http_proxy="http://192.168.0.1:3128"
```

Para ver que todo está bien:

```
lynx http://www.iesmurgi.org
```

1. Actualicemos el sistema, para eso, con conexión a internet ejecutamos[4]

   ```
   #apt-get update
   #apt-get upgrade
   ```

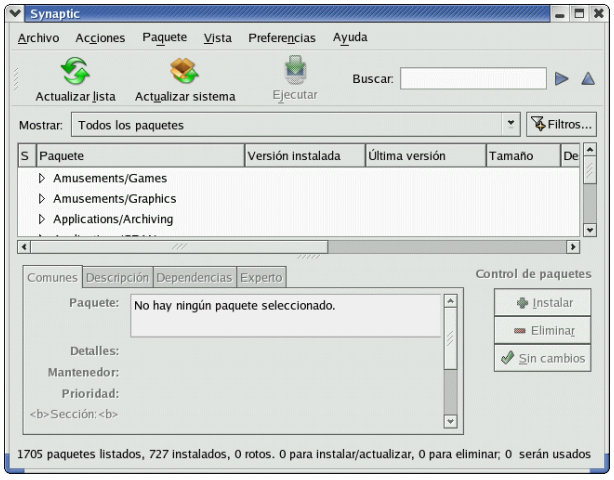
2. Veamos un par de pinceladas más sobre su uso con Fedora ya que si se viene del mundo Debian su manejo no debe presentar ningún problema.

   a) Instalación de **synaptic**: se trata de una herramienta gráfica para gestionar los paquetes instalados en nuestra máquina, disponible tanto para Guadalinex (se instala por defecto) como para Fedora. Para instalarla en Fedora sólo hay que escribir:

   ```
   # apt-get install synaptic
   Leyendo listas de paquetes... Done
   Construyendo árbol de dependencias... Done
   Se instalarán los paquetes NUEVOS siguientes:
     synaptic
   0 upgraded, 1 newly installed, 0 removed and 86 not upgraded.
   Need to get 1497kB of archives.
   After unpacking 5321kB of additional disk space will be used.
   ```

   Una vez instalado, para ejecutarlo desde un xterm, escribimos

   ```
   # synaptic &
   ```

   El manejo de este *front-end* gráfico para mantener el sistema de paquetes es inmediato.

   b) Como ejercicio: instalar el programa[5] `mc` usando el comando `apt-get`. ■

[4] ¡Cuidado: si no tenemos claro qué hacemos, mejor dejarlo como está!
Podemos usar el parámetro `dist-upgrade` en lugar de `upgrade`, la diferencia entre ambos es que con `upgrade` se actualizará el sistema pero no se instalará un paquete nuevo, ni se eliminará uno ya instalado, ni se actualizará un paquete que presente conflictos con otro ya instalado. Sin embargo, si usamos `dist-upgrade` realizamos una actualización completa, es decir, una vez determinado el mejor conjunto de paquetes para actualizar el sistema lo máximo posible, se instalan, actualizan y eliminan todos los que sean necesarios.

[5] En Fedora 3 no se instala por defecto

Capítulo 2

Redes de Ordenadores

> El concepto de trabajo en redes es probablemente tan antiguo como lo es el de las telecomunicaciones. Imagínese por un momento, gente viviendo en la Edad de Piedra, en donde los individuos usen tambores para transmitirse mensajes. Supóngase que un hombre de las cavernas A quiere invitar a otro hombre B a una partida de choques de piedra. Lamentablemente viven tan distantes, que a B le sería imposible escuchar el tambor de A cuando éste lo llame. ¿Qué puede hacer A para remediar esto? Él podría 1) ir caminando al sitio de B, 2) conseguir un tambor más grande, ó 3) pedirle a C, quien vive a mitad de camino, que reenvíe el mensaje. La tercera elección es denominada Trabajo en Redes. (*Guía de Administración de Redes Segunda Edición*, OLAF KIRCH)

La cita anterior nos expresa claramente que la necesidad de comunicación y, más aún, de la comunicación a distancia, ha existido entre los seres humanos desde la noche de los tiempos y ante esta necesidad, se han ido planteado diferentes alternativas "tecnológicas".

Más moderna es la necesidad de comunicar ordenadores, que en un principio fueron islas no conectadas entre sí. Rápidamente, se hizo necesario desarrollar sistemas que permitieran la comunicación entre diferentes ordenadores y la correcta transferencia de datos entre ellos, surgiendo de esta forma el concepto de "redes de ordenadores" y de "trabajo en red"[1].

La parte que viene a continuación puede "pecar" de teórica, pero preferimos presentarla por motivos de completitud y referencia. No es necesario que comprendáis completamente todos los conceptos y si alguno se cansa, puede pasar al capítulo 4.

2.1. Modelo de Referencia OSI

A través de una red se pueden ejecutar procesos en otro ordenador, acceder a sus ficheros, enviar mensajes, compartir programas, etc. Esta comunicación de datos se realiza mediante el envío de unidades de información, lógicamente agrupadas, denominadas *paquetes de datos* .

Los paquetes de datos incluyen la información que intercambian las aplicaciones, junto con otros elementos necesarios para hacer que la comunicación sea factible y confiable en la relación con los dispositivos de destino, como por ejemplo, las direcciones de origen y destino. La dirección de origen de un paquete especifica la identidad del ordenador que envía el paquete. La dirección de destino especifica la identidad del ordenador que recibe el paquete. Esta identificación es necesaria, de la misma forma que lo es la dirección del destinatario en una carta postal.

Para que los paquetes de datos puedan viajar desde el origen hasta su destino a través de una red, es importante que todos los dispositivos de la red hablen el mismo lenguaje, lo que denominamos protocolo. Un **protocolo** es una descripción formal de un conjunto de normas y convenciones

[1] Hasta el punto de que hoy día, con una conexión a la red, es como podemos obtener el máximo aprovechamiento de un ordenador. Sin ella, un ordenador es como un naúfrago en una isla.

que determinan el formato y la transmisión de los datos entre los diferentes dispositivos de una red. Podemos entenderlo como la gramática que rige una lengua, aunque los ordenadores son menos "inteligentes" que nosotros y se bloquearían con nuestras faltas de ortografía. Por ello, esas reglas son más estrictas en la comunicación entre ordenadores.

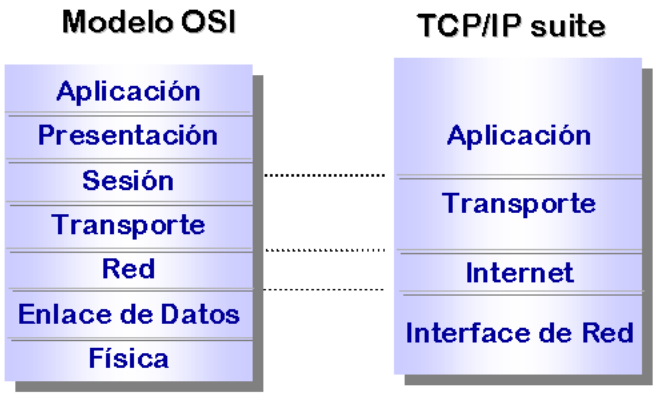
A mediados de los años 70, diversos fabricantes desarrollaron sus propios sistemas de redes locales y protocolos. En 1980, la empresa Xerox, en cooperación con DEC e Intel, desarrolló las especificaciones del primer sistema de red, denominado EtherNet. En 1982 aparecen los ordenadores personales, y en 1986, IBM introdujo la red TokenRing.

El principal inconveniente de los primeros sistemas de comunicación en red fue que cada uno de ellos era propiedad de una empresa, siendo desarrollados con hardware y software propietarios, con elementos protegidos y cerrados, que usaban protocolos y arquitecturas diferentes. Como consecuencia de ello, la comunicación entre ordenadores pertenecientes a distintas redes era muy difícil, por no decir imposible.

Cuando las empresas intentaron comunicar sus redes, cada una con su implementación particular, se dieron cuenta de que necesitaban salir de los sistemas de redes propietarios, optando por una arquitectura de red con un modelo común que hiciera posible interconectar distintas redes sin problemas.

Para solucionar este problema, la Organización Internacional para la Normalización (ISO[2]) reconoció que era necesario crear un modelo común que pudiera ayudar a desarrollar redes que pudieran comunicarse y trabajar conjuntamente[3]. Como consecuencia de ello, elaboraron el modelo de referencia OSI en 1984.

El modelo OSI (*Open Systems Interconnection*) define la forma en que se comunican los sistemas abiertos[4] de telecomunicaciones. El modelo de referencia está compuesto de 7 capas y la forma en que deben comunicarse entre ellas. Estas capas se presentan normalmente como una pila, que se conoce como la Pila de Protocolos OSI (*OSI Protocol Stack*).



Los niveles o capas del modelo OSI son desde el inferior hasta el superior: **Nivel Físico, Nivel de Enlace de Datos, Nivel de Red, Nivel de Transporte, Nivel de Sesión, Nivel de Presentación y Nivel de Aplicación**.

Aunque nuestro objetivo final sea el estudio de las redes TCP/IP, base de la Internet actual, por sus bondades didácticas, comentaremos antes el modelo OSI. Conoceremos el funcionamiento de los modelos de capas y lo trasladaremos posteriormente al estudio de TCP/IP.

↬ El concepto que subyace en el Modelo de Referencia OSI, es similar a cuando escribimos una carta[5]. Veamos la figura que viene a continuacion, que nos presenta dos ordenadores (equipo

[2] *International Standards Organization*

[3] denominado *interoperabilidad*.

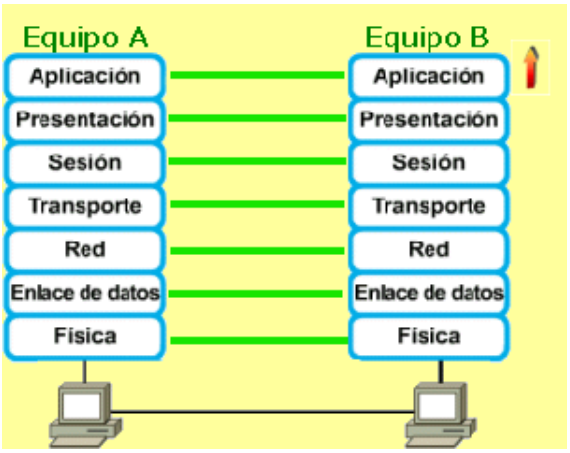
[4] Los SISTEMAS ABIERTOS son aquellos que se basan en especificaciones estándares, definidas por organismos internacionales independientes.

[5] La analogía solamente pretende que comprendamos cómo funciona la comunicación entre las capas del modelo OSI.

A y equipo B) e imaginémonos que usted (usuario del equipo A en Almería) desea inscribirse en los cursos que organiza la Sociedad Andaluza de Educación Matemática Thales y el CICA. Para ello, debe mandar una carta de solicitud a la Secretaría de Thales que se encuentra en Cádiz. Primero, vamos a descender por la pila de protocolos del Equipo A de la izquierda, desde la capa de Aplicación hasta la capa Física. Mediante el cable que une ambos equipos, pasamos al equipo B de la derecha y ascendemos por la pila de protocolos de este equipo, desde su capa Física hasta llegar a la capa de Aplicación.

Realizando una analogía entre ambos modelos, podemos imaginar que en la capa de Aplicación del Equipo A, decidimos el mensaje que queremos enviar: la petición de un curso de Thales. En la capa de Presentación lo plasmamos sobre el papel. Siguiendo el símil, en la capa de Sesión metemos el papel en un sobre con la dirección del destinatario[6]. Lo bajamos al buzón de Correos más cercano, que constituiría la capa de Transporte. Dentro de Correos, no sabemos qué ruta seguirá la carta[7], constituyendo una auténtica capa de Red: la Red de Correos. La capa de Enlace sería la Oficina de Correos que dirige la carta hacia el tren, que llegará "físicamente" a la localidad de destino.

Ya ha llegado nuestra carta a la estación de ferrocarril de Cádiz, con lo que ha llegado al nivel Físico del Equipo B. Se envía a la Oficina de Correos correspondiente a la zona de destino, pasando por los niveles de Enlace y de Red. El cartero, mirando la dirección de entrega, lo deposita en el buzón de la Secretaría de Thales, cumpliendo con el nivel de Transporte. Allí, el personal de la Secretaría recoge el sobre y lo abre, finalizando la capa de Sesión. La solicitud es leída como correspondería al nivel de Presentación y en la capa de Aplicación se procede a inscribir al alumno en el curso solicitado.



Podemos observar que la comunicación final "virtual" se ha producido entre el alumno y la Secretaría de Thales, solicitando nuestra inscripción en los cursos. Sin embargo, la comunicación "real" ha ido viajando desde nosotros, hasta el buzón de correos, los distintos medios de transporte, la oficina de correos de destino, el cartero y la Secretaría de Thales.

Volvamos al Modelo de Referencia OSI. En éste, el protocolo de cada capa sólo se interesa por la información que le corresponde a su capa, y no por la información que necesitan o procesan las demás capas.

↬ Por ejemplo: El e-mail es un protocolo de aplicación que se comunica sólo con otras aplicaciones que hablan el mismo protocolo (SMTP, POP, IMAP). Por lo tanto, la aplicación de e-mail no se interesa de si la capa física es una red ethernet, una línea ADSL o un módem.

[6] Ponemos también el remite, por si nos han de enviar una carta de confirmación de la inscripción.

[7] Puede que en algunos casos, para ir de Almería a Cádiz, pase por Barcelona, pero a nosotros nos dará igual, excepto en el tiempo que tarde.

La información se pasa a las capas de abajo hasta que la información llega a la red. En el nodo remoto, la información es entonces pasada a las capas superiores hasta que llega a la aplicación correspondiente.

Pasamos a detallar las capas del modelo OSI, desde las capas del nivel superior a las inferiores[8]. Aunque las capas TCP/IP, como veremos más adelante, no son exactamente iguales, tomaremos como ejemplo el funcionamiento de un cliente y un servidor web.

2.1.1. Capa de Aplicación

La capa de aplicación permite al usuario acceder al servicio final deseado. Proporciona las interfaces de usuario y soporte a los servicios como correo electrónico, transferencia de archivos o bases de datos.

↬ En el caso de la comunicación entre un cliente y un servidor web, sería el protocolo HTTP que hace que el cliente solicite páginas al servidor y éste se las sirva.

2.1.2. Capa de Presentación

Este nivel elimina los problemas que puedan surgir al comunicar distintas arquitecturas, pues cada arquitectura estructura los datos de una forma específica[9], que no tienen por qué ser compatibles. Se traducen los datos a un formato común, independiente de la arquitectura de máquina.

En esta capa se define el formato de los datos que se van a intercambiar entre las aplicaciones y se ofrece a los programas de aplicación un conjunto de servicios de transformación de datos. En caso de ser necesario, también se encarga de la compresión y del cifrado

↬ Por ejemplo, si el cliente web se ejecuta en una máquina de arquitectura Intel y el servidor en un PowerPC, esta capa utilizaría un formato común de intercambio de la información, que luego cada arquitectura adapta al suyo en concreto.

2.1.3. Capa de Sesión

Se encarga de organizar y sincronizar el diálogo entre los dos extremos implicados. Ofrece mecanismos para gestionar el diálogo entre los sistemas como: quién debe emitir en cada instante, agrupamiento de datos en unidades lógicas, recuperación si se produce algún problema en la comunicación.

↬ Dentro de una sesión, el cliente y el servidor web se intercambian múltiples peticiones. Por ejemplo, dentro de una página web se pueden incorporar múltiples imágenes que el cliente tiene que solicitar al servidor. También, siguiendo los hiperenlaces, pedimos distintos contenidos del servidor. Esta conexión estable entre el cliente y el servidor se mantiene gracias a la capa de sesión. Si se produce un corte momentáneo en la línea de comunicaciones entre nuestro cliente web y el servidor web, esta capa se encarga de mantener la comunicación hasta que se restablezca, en el caso de que no sea demasiado tiempo, o la da por perdida definitivamente.

2.1.4. Capa de Transporte

Es la responsable del envío desde el origen al destino (es decir, de extremo a extremo) del mensaje completo.

La capa de red[10] supervisa el envío extremo a extremo de los paquetes de forma individual, pero no reconoce ninguna relación entre esos paquetes, tratando a cada uno de forma independiente. Sin embargo, la capa de transporte asegura que el mensaje completo llegue desde el origen al destino, intacto y en el orden correcto, supervisando el control de flujo y control de errores desde un extremo a otro de la comunicación. La capa de transporte asegura un servicio fiable.

[8] No os asustéis si hay términos que no entendéis o no habéis escuchado en la vida. Ah, y no es necesario aprendérselo de memoria. Lo importante es que captéis el modelo.

[9] Por ejemplo, en unas arquitecturas el octeto de información más significativo se sitúa primero y en otras, es justo al contrario.

[10] que se encuentra bajo la de transporte.

Si el fichero que el servidor web tiene que transmitirle al cliente es de un tamaño grande, tiene que partirse en trozos más pequeños durante el tránsito. La capa de transporte se ocupa de partirlo en trozos en el origen y de recomponer los trozos en el destino.

↬ Es como si un convoy de 200 vehículos debe atravesar un río y solamente existen barcas con capacidad para 50 vehículos. Deberán partirse en "trozos" de 50 vehículos para caber en las barcas y en la otra orilla, volver a recomponer el convoy.

2.1.5. Capa de Red

Es la responsable del envío desde la estación origen a la estación destino de los paquetes, es decir, se asegura de que cada paquete llegue desde su punto inicial hasta su punto final.

↬ Siguiendo el ejemplo anterior, sería el barquero, cuyo objetivo es pasar cada barca a la otra orilla, pero no tiene responsabilidad sobre el reagrupamiento del convoy una vez en la otra orilla.

Si dos sistemas están conectados en el mismo enlace, no existe la necesidad de la capa de red (por ejemplo, dentro de una misma LAN). Sin embargo, si dos sistemas están en diferentes redes físicas, será necesaria una capa de red para dirigir (o "enrutar") la entrega desde la red física de origen a la red física de destino del paquete.

Entre las responsabilidades de la capa de red se incluyen:

- *Direccionamiento lógico*: El direccionamiento físico implementado en la capa de enlace de datos[11] manipula el problema del direccionamiento localmente. Pero si un paquete pasa de la frontera de la red, se necesita otro sistema de direccionamiento para ayudar a distinguir los sistemas origen y destino. La capa de red añade un encabezamiento al paquete que llega de la capa superior, que entre otras cosas, incluye la dirección lógica del origen y del destino.

- *Enrutamiento*: Cuando redes independientes son conectadas para crear una inter-red (e.g. una red de redes, como Internet), los dispositivos (llamados routers o gateways) enrutan o encaminan los paquetes a su destino final.

↬ Si nuestro cliente web se encuentra en la red interna de nuestro centro y el servidor al que nos conectamos está en el Instituto Tecnológico de Massachusetts (MIT), la capa de red sabra cómo llegar hasta allí, pasando por los múltiples caminos de Internet.

2.1.6. Capa de Enlace de Datos

La capa de enlaces de datos ensambla los bits de la capa física en grupos de tramas (protocolos de red) y asegura su correcto envío.

También es la encargada de la verificación y corrección de errores de la capa física. En caso de que ocurra un error en los bits, se encarga de avisar al transmisor de que efectúe una retransmisión, y por lo tanto, la capa de enlace se encarga también del control de flujo físico de los datos.

La capa de enlace de datos se divide en dos subcapas:

1. LLC (*Logical Link Control*): define cómo se transfieren los datos sobre el cable y provee servicios de enlace de datos a las capas superiores.

2. MAC (*Medium Access Control*): define quién puede usar la red cuando múltiples dispositivos están intentando acceder simultáneamente (e.g. token passing, Ethernet CSMA/CD,..).

↬ Por ejemplo, comunica nuestro sistema linux a la red del centro y controla cuándo podemos emitir en un medio que es compartido entre todos los ordenadores del aula.

[11] Que se encuentra bajo la de red.

2.1.7. Capa Física

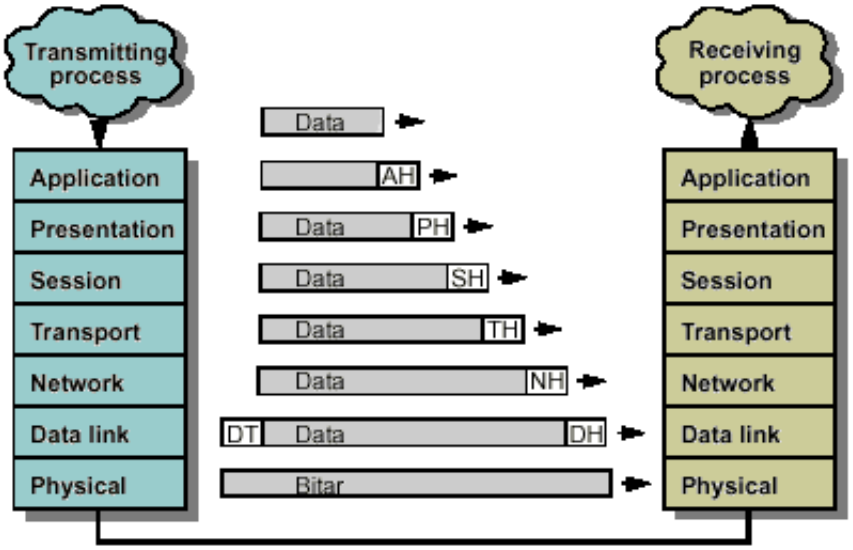
Se ocupa de la transmisión de bits a través de un canal de comunicación físico.

Regula aspectos de la comunicación tales como el tipo de señal (analógica, digital,..), el esquema de codificación, sincronización de los bits, tipo de modulación, tipo de enlace (punto a punto, punto-multipunto), el modo de comunicación (dúplex, half-dúplex o símplex), tasa de bits (número de bits por segundo), topología empleada, y, en general, todas las cuestiones eléctricas, mecánicas, señalización y de procedimiento en la interfaz física (cables, conectores,...) entre los dispositivos que se comunican.

Lo dicho, cables y señales eléctricas, ópticas o de cualquier otro tipo[12].

2.2. Comunicación entre capas

Tras ver las distintas capas del Modelo OSI[13], corresponde estudiar la forma en la que se lleva a cabo una comunicación en el Modelo de capas.



Supongamos que el proceso emisor (en la parte izquierda de la figura) tiene información que enviar, para ello entregará los datos[14] a la capa de Aplicación. Esta capa añade a la información una cabecera (AH[15]) que permite procesar el protocolo que tenga definido. El conjunto formado por los datos originales y la cabecera de aplicación es entregado a la capa de presentación.

Esta capa transforma el bloque recibido en función del servicio pedido, y añade también su nueva cabecera (PH[16]). Este nuevo conjunto de datos es entregado a la capa inmediatamente inferior: la capa de sesión.

Cada nivel de la torre OSI añade una cabecera a los datos a transmitir, a excepción del nivel 1 que no añade nada, y del nivel 2 que además añade una cola (DT[17]). Dichas cabeceras son datos de control para el nivel equivalente[18] en el otro extremo de la comunicación.

Es importante destacar que el conjunto de datos que pasa de la capa N a la capa (N -1) puede ser fragmentado en bloques más pequeños. En este caso, cada bloque llevará su propia cabecera y

[12] Señales cerebrales dentro de poco tiempo.
[13] Conocido coloquialmente como el modelo OSI de la ISO.
[14] En la figura, Data. Perdonad si incluimos la figura original en inglés, pero parecía adecuada y no hubiera salido tan bonita.
[15] Proviene de *Application Header*
[16] *Presentation Header*
[17] De *Data Link Trailer* (Cola de Enlace de Datos).
[18] El que se encuentra a la misma altura.

además, la capa que realiza la fragmentación deberá ser la encargada (en la máquina receptora) de recomponer los bloques hasta formar de nuevo el conjunto original de datos y entregarlos a la capa superior, así hasta llegar al proceso receptor.

↬ Aunque la idea puede parecer rebuscada, es un proceso parecido a la comunicación entre personas. Inicialmente tenemos una idea que deseamos comunicar a nuestro interlocutor. Esta idea es entregada a la zona del cerebro encargada del lenguaje, que generará los impulsos nerviosos necesarios para hacer vibrar nuestras cuerdas vocales y emitir unidades pequeñas (fonemas), produciéndose un sonido que será captado por el oído de nuestro interlocutor. Los impulsos nerviosos generados por el oído de nuestro interlocutor serán enviados a su cerebro, que reconstruye los fonemas, los agrupará en palabras, y de ellas se extraerá el significado de la información. Todo este proceso se puede abstraer y resumir como que la información, el mensaje, que deseábamos comunicar ha llegado a la otra persona.

2.2.1. Tipos de Servicios

Según las capacidades que aportan, para el modelo OSI existen varias clasificaciones de los servicios que puede proporcionar una red. Una clasificación permite dividirlos en *servicios orientados a la conexión* y en *servicios sin conexión*[19], y otra, los divide en *servicios confiables* y *servicios no confiables*.

1. *Servicios orientados a la conexión*: En ellos la conexión es como un tubo a través del cual se envía la información de forma continuada, por lo que los mensajes llegan en el orden en que fueron enviados y sin errores. Proporcionan un servicio confiable de comunicación de datos. Cada paquete se procesa en su secuencia correcta: después del anterior y antes del posterior. No pueden faltar paquetes, ni procesarse uno sin haber procesado los anteriores.

   ↬ Una analogía es el sistema telefónico. Si la comunicación se corta, hay que restablecerla de nuevo.

2. *Servicios no orientados a la conexión*: En los que cada mensaje lleva la dirección completa de su destino. La información no se garantiza que llegue de forma continua. La ruta que sigue cada mensaje es independiente. El servicio no es entonces confiable, pues no se garantiza el orden de llegada de los paquetes, ni se controla su flujo, porque para estos servicios no es necesario. Por ello, los paquetes deben llevar sus direcciones completas de destino.

   ↬ Una analogía sería el caso del sistema de correo convencional. Puede que una carta que habíamos enviado hace cinco días, llegue antes que una que enviamos hace dos semanas al mismo destinatario. Podemos procesar un paquete (leer la carta), aún cuando el anterior no haya llegado aún, o incluso puede que nunca lo haga.

Otra clasificación de los servicios es la que distingue entre confiables y no confiables:

- *Servicios confiables*: son aquellos en los que la transmisión de datos está controlada en cada momento, pudiéndose determinar el correcto envío y recepción de todos los datos transmitidos. Para ello la máquina receptora envía mensajes de acuse de recibo de las tramas recibidas a la máquina emisora.

  ↬ Es como si todas las cartas que mandáramos en correo postal lo hiciéramos con acuse de recibo. Es más lento y más caro, pero más seguro.

- *Servicios no confiables*: en éstos no existe un control de los datos transmitidos, por lo que no se puede garantizar que se hayan recibido todos los datos.

  ↬ Una analogía sería el caso de un naúfrago en una isla. Manda muchos mensajes dentro de botellas y se conformaría con que uno al menos sea encontrado por alguien.

[19] o no orientados a la conexión.

Capítulo 3

Conjunto de Protocolos TCP/IP.

> En vez de ser un ideal académico, el modelo de referencia TCP/IP lo elaboraron los fabricantes y los programadores que, por fin, llegaron a un acuerdo sobre las comunicaciones en Internet. Además, este modelo es más sencillo que el modelo ISO, porque desarrolla TCP/IP desde el punto de vista del programador, partiendo de que el mundo es real y práctico. (*Firewall Linux*, Robert L. Zieger)

TCP/IP ha llegado a convertirse en un estándar de facto para la comunicación de redes de ordenadores. De hecho, es la familia de protocolos[1] empleada por Internet.

TCP equivale a las siglas de *Transmission Control Protocol* e IP corresponde a las siglas de *Internet Protocol*. Estos protocolos se crearon y normalizaron mucho antes de que se definiera el Modelo de Referencia OSI de la ISO. Ya a finales de los 80, muchas empresas, administraciones y organismos usaban TCP/IP, cuando todavía OSI no estaba totalmente desarrollada. Aun así, el modelo OSI es una buena arquitectura de organización de protocolos, y es muy didáctico. Esa ha sido la razón por la que lo hemos visto anteriormente.

No existe un modelo oficial de protocolos TCP/IP, al contrario que en OSI. Los protocolos se han ido definiendo de una forma un tanto anárquica, y a posteriori han sido englobados en capas. La operatividad (que el sistema funcionara) primaba sobre el academicismo.

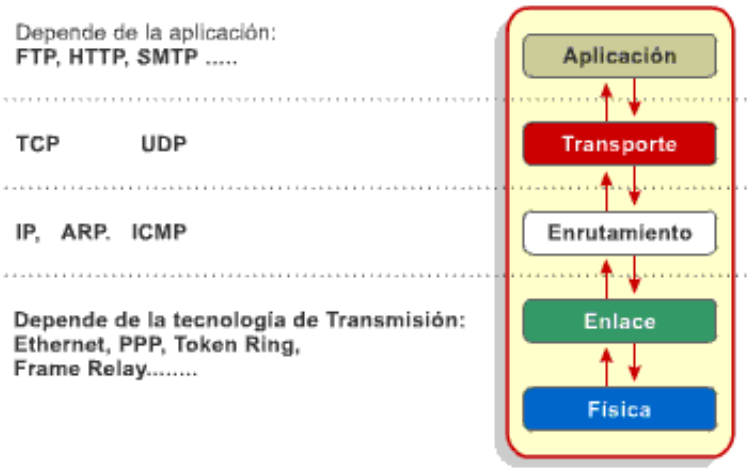
3.1. Niveles de la Arquitectura TCP/IP

En la imagen se pueden apreciar los niveles de la arquitectura TCP/IP junto a ejemplos de protocolos pertenecientes a cada una de las capas. La arquitectura TCP/IP podemos encontrarla, según las fuentes que consultemos, dividida en cuatro o en cinco niveles. Éstos son: Nivel Físico, Nivel de Enlace[2], Nivel de Red[3], Nivel de Transporte y Nivel de Aplicación.

[1] De todos los protocolos que la componen, dos de ellos, el protocolo TCP y el protocolo IP, dan nombre a toda la familia.

[2] Algunos autores unifican los niveles de Enlace y Físico en uno solo.

[3] o de Enrutamiento



La forma en que trabajan las capas de TCP/IP es similar a como hemos visto en el modelo OSI.

3.1.1. Nivel de Aplicación.

Proporciona una comunicación entre procesos o aplicaciones que pueden estar en sistemas distintos[4]. Además de las aplicaciones, este nivel se ocupa de las posibles necesidades de presentación y de sesión.

Los protocolos de aplicación más utilizados son: TELNET (terminal remoto), FTP (transferencia de ficheros), HTTP (el protocolo entre clientes y servidores web) o SMTP (correo electrónico).

3.1.2. Nivel de Transporte.

Proporciona transferencia de datos de extremo a extremo[5], asegurando que los datos llegan en el mismo orden en que han sido enviados, y sin errores. Esta capa puede incluir mecanismos de seguridad.

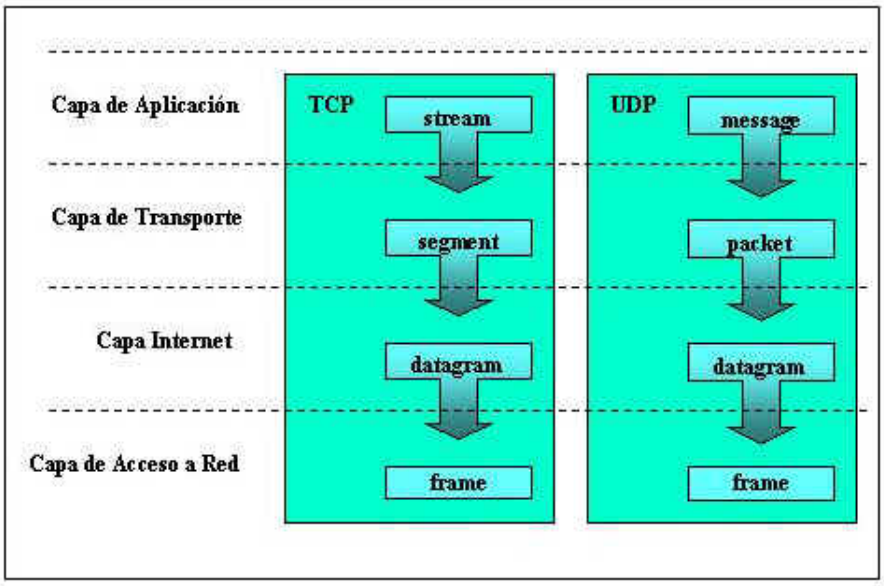
Consta de dos servicios diferenciados. Un servicio consiste en el envío y recepción de datos orientado a conexión (TCP) y el otro (UDP) consiste en el envío y recepción de datos no orientados a conexión.

↬ Para ver mejor la diferencia entre estos dos servicios, podemos poner dos ejemplos. En el primer caso, escuchamos una canción a través de Internet. Si en un momento determinado falta un paquete por congestión en la red, es preferible descartarlo y pasar al siguiente para seguir escuchando la música, a que paremos para recuperar el paquete perdido y no seguir escuchando nada. El servicio adecuado en este caso sería el no orientado a conexión (UDP). En el otro ejemplo, estamos descargando un fichero, por ejemplo una imagen del CD de instalación de Linux, desde un servidor de Internet. En este caso, si se pierde un paquete, debemos volver a pedirlo y esperar a que llegue, porque en caso contrario, todo lo recibido antes y después no nos serviría de nada.

El protocolo TCP (*Transmission Control Protocol*) de la capa de transporte es un servicio orientado a conexión. La unidad de datos que envía o recibe el protocolo TCP es conocida con el nombre de *segmento* TCP.

El protocolo UDP (*User Datagram Protocol*) de la capa de transporte es un servicio no orientado a conexión. La unidad de datos que envía o recibe el protocolo UDP es conocida con el nombre de *paquete* UDP. En la siguiente figura vemos los nombres que reciben las unidades de datos de las distintas capas, según pertenezcan al servicio TCP o al UDP.

[4]Para comunicar procesos en un único ordenador podemos también utilizar TCP/IP y, de hecho, es muy común.
[5]De ordenador origen a ordenador destino, aunque entre medio pase por otros sistemas como routers o gateways.



3.1.3. Nivel de Red.

Se encarga de conectar equipos que están en redes diferentes. Permite que los datos atraviesen distintas redes interconectadas desde un sistema origen hasta un sistema destino.

La capa de red es la responsable de proveer los siguientes servicios a la capa de transporte:

- Establecer el sistema de direccionamiento lógico de la red.
- Enrutamiento de paquetes.

Durante el proceso de enrutamiento se hace uso de un servicio no orientado a conexión para el envío y recepción de paquetes.

Si un paquete que va a ser enrutado excede la máxima unidad de transferencia *Maximum Transfer Unit* (MTU) de un medio físico[6], esta capa fragmenta el paquete con el fin de adaptarse al tamaño máximo de cada medio y el paquete es ensamblado en el sistema destino. Recordemos el ejemplo del convoy que debe fragmentarse para cruzar un río. Por ejemplo, en Ethernet este tamaño es de 1.500 bytes o de 4.450 bytes en un enlace de tecnología ATM.

Esta capa está compuesta por los protocolos: IP, ARP e ICMP principalmente.

- El protocolo IP (*Internet Protocol*) ofrece el servicio de direccionamiento lógico de la red TCP-IP y el de enrutamiento de paquetes.
- El protocolo ARP (*Address Resolution Protocol*) ofrece el servicio de resolución de direcciones IP con su respectiva dirección física.
- El protocolo ICMP (*Internet Control Message Protocol*) ofrece el servicio de informe de errores que pueden ocurrir durante el enrutamiento de paquetes.

La unidad de datos que envía o recibe el protocolo IP se conoce con el nombre de *datagrama* IP.

3.1.4. Nivel de Enlace o de Acceso a la red.

Es el nivel responsable del intercambio de datos entre dos sistemas conectados a una misma red[7]. Controla la interfaz entre un sistema final[8] y una red.

[6] Que puede ser una Ethernet, línea de módem...

[7] Por ejemplo, en una Red de Área Local con tecnología Ethernet.

[8] La terminología para referirse en este ámbito a un ordenador conectado a una red es variada: sistema, host, estación, nodo...

3.1.5. Nivel Físico.

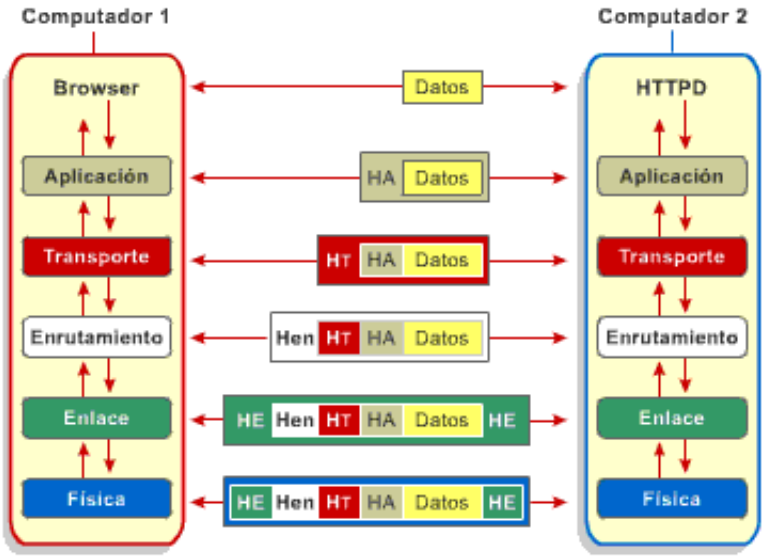
Define las características del medio, su naturaleza, el tipo de señales, la velocidad de transmisión, la codificación, etc.

La unidad de envío o recepción de datos de la capa física se conoce con el nombre de *trama* (frame). Entrarían aquí el cableado, la electrónica de red y parte de la tarjeta de red.

3.2. Añadiendo cabeceras y colas

En la siguiente figura, podemos observar que el envío y recepción de datos entre dos aplicaciones es un proceso de intercambio de datos entre capas del mismo nivel, basado en un modelo cliente-servidor.

La aplicación del computador 1 es una aplicación cliente (Browser o Navegador) que utiliza el protocolo "HTTP" de la capa de aplicación. La aplicación del computador 2 es un servidor de páginas web (proceso httpd, que puede ser Apache), que se intercambian los datos, en un mismo lenguaje. Los niveles correspondientes en cada una de las máquinas se comunican haciendo uso de la información que incorporan en las cabeceras y colas correspondientes.

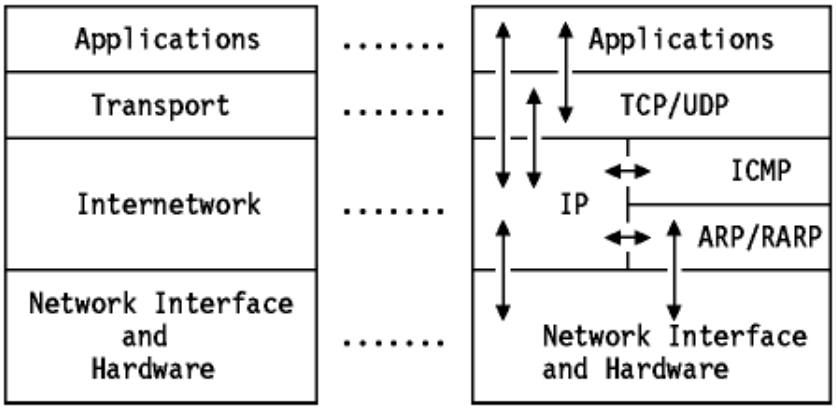


Capítulo 4

TCP/IP: desde los pulsos hasta los datos

> Crear y operar cortafuegos que protejan el perímetro de la red no es física cuántica. De todas formas, hacerlo exige un mejor conocimiento de las redes TCP/IP que el necesario para la mayoría del resto de las funciones de la administración de sistemas y redes. (*Firewalls*, Bill MacCarty)

Pasemos a introducir algunos protocolos importantes de las diferentes capas de TCP/IP, empezando desde el nivel inferior y continuando hasta los de nivel superior. En el nivel inferior, el físico, el protocolo más común en una red de área local es el Ethernet. En el nivel de red, la estrella es el protocolo IP junto a sus ayudantes[1], ARP, RARP e ICMP. En el nivel de transporte, se encuentran TCP y UDP, que dan servicio a los protocolos de aplicación como pueden ser HTTP, FTP o SMTP.



4.1. Nivel Físico

4.1.1. Ethernet e IEEE 802.3

Es el medio más común que nos encontramos en una Red de Área Local (LAN). Conviven dos implementaciones muy parecidas, la Ethernet y la norma de IEEE 802.3, que está basada en la anterior.

[1] IP es el protocolo que transfiere los datos hacia las aplicaciones y ARP, RARP e ICMP le sirven para el control de errores, búsqueda de direcciones...

Ethernet es una especificación LAN de "banda base" inventada por Bob Metcalfe (fundador de 3com) y David Boggs en 1973, mientras trabajaban en Xerox PARC (*Palo Alto Research Center*) que operaba a 10Mbps utilizando un protocolo de acceso múltiple al medio conocido como CSMA/CD (*Carrier Sense Multiple Access/Collision Detect*) sobre un cable coaxial. Ethernet fue creado en Xerox en los años 70, pero el término es usualmente referido para todas las LAN CSMA/CD.

La especificación IEEE 802.3 fue desarrollada en 1980 basada sobre la tecnología original Ethernet. La versión 2.0 de Ethernet fue desarrollada conjuntamente por DEC (*Digital Equipment Corporation*), Intel, y Xerox y es compatible con el estándar IEEE 802.3. El estándar IEEE 802.3 provee una gran variedad de opciones de cableado. Tanto Ethernet como IEEE 802.3 se implementan normalmente en la tarjeta de red.

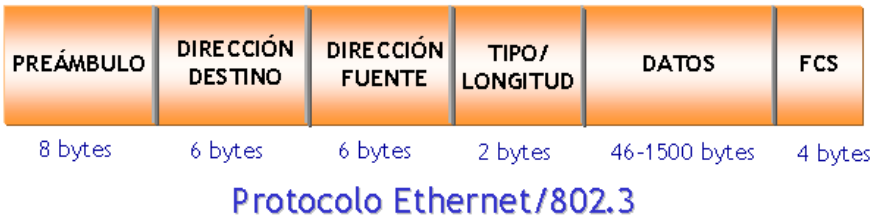
Direcciones MAC

Todos los interfaces (tarjetas) compatibles Ethernet/802.3 poseen una Dirección MAC o **Dirección Hardware** única en el mundo, de 48 bits (o lo que es lo mismo, 6 bytes) de longitud. Cada fabricante de equipos ethernet tiene asignado un rango de direcciones, y es responsabilidad de éste asignar una dirección distinta a cada tarjeta de red. Las direcciones MAC están almacenadas en una pequeña memoria que poseen las tarjetas de red.

Las direcciones MAC se representan en hexadecimal con el siguiente formato: XX:XX:XX:XX:XX:XX[2]. Por ejemplo, 00-40-05-7F-CE-85, sería una dirección MAC válida.

Conviene decir que las direcciones MAC solamente sirven para comunicarse en un mismo medio físico, como por ejemplo, una Red de Área Local (LAN). Cuando hay que comunicarse con un dispositivo que no pertenece a esa LAN, entran en juego el nivel de red y las direcciones lógicas[3].

Estructura de la trama Ethernet



La estructura de la trama Ethernet/802.3 es como sigue:

Preámbulo: (64 bits) El paquete comienza con una secuencia de unos y ceros alternados (de 56 bits en 802.3 ó de 62 bits en Ethernet), que se completa hasta 64 bits en ambos casos. El preámbulo recibido en la red no es pasado por la tarjeta de red hasta el sistema.

Dirección de Destino: (6 bytes) La dirección de destino (DD) es de 48 bits (6 bytes) de tamaño[4], de la cual se transmite primero el bit menos significativo. La DD es utilizada por la tarjeta del sistema receptor, para determinar si el paquete entrante es para él. Si el sistema recepetor detecta una correspondencia entre su dirección MAC y la dirección que viene en el campo DD, recogerá el paquete. Los otros sistemas, a los que no se dirige la trama, ignorarán el resto del paquete.

Las direcciones de destino pueden ser:

1. **Individual** (física): El campo DD contiene una dirección única e individual asignada a un nodo en la red. Es decir, una dirección MAC de una tarjeta de la red.

2. **Broadcast** (difusión): El campo DD está formado todo por unos. Es una dirección especial y todos los dispositivos MAC de la red deberán recibir el mensaje de broadcast.

[2]El separador pueden ser los dos puntos (:) o un guión (-)
[3]Por ejemplo, las direcciones IP.
[4]Es una dirección MAC.

Dirección Fuente: (6 bytes) La dirección fuente (DF) es de 48 bits (6 bytes) de tamaño, transmitiéndose primero el bit menos significativo. El campo DF lo provee la MAC de la tarjeta emisora, la cual inserta su propia dirección física en este campo al transmitirse la trama, indicando que fue la estación origen. Los formatos de direcciones tipo broadcast son ilegales en el campo DF.

Longitud/Tipo: (2 bytes) El campo Longitud (en 802.3) o Tipo (en Ethernet) de 2 bytes va tras el campo DF. La elección de escoger Longitud o Tipo es dependiente de si la trama es 802.3 o Ethernet.

Datos: (46 - 1500 bytes) Este campo contiene los datos (la información útil que es transferida) cuyo tamaño varía de 56 a 1.500 bytes.

FCS (*Frame Check Sequence*): (4 bytes) Contiene el valor del algoritmo CRC (*Cyclic Redundancy Check*) de 32 bits de la trama completa. El CRC es calculado por la estación emisora sobre los campos DD, DF, Longitud/Tipo y Datos y es anexado en los últimos 4 bytes de la trama. El mismo algoritmo CRC es utilizado por la estación receptora para calcular el valor CRC en la trama recibida. El valor calculado por el receptor es comparado con el valor que fue puesto en el campo FCS por la estación emisora, obteniendo un mecanismo de detección de errores en caso de datos corruptos.

Protocolo CSMA/CD.

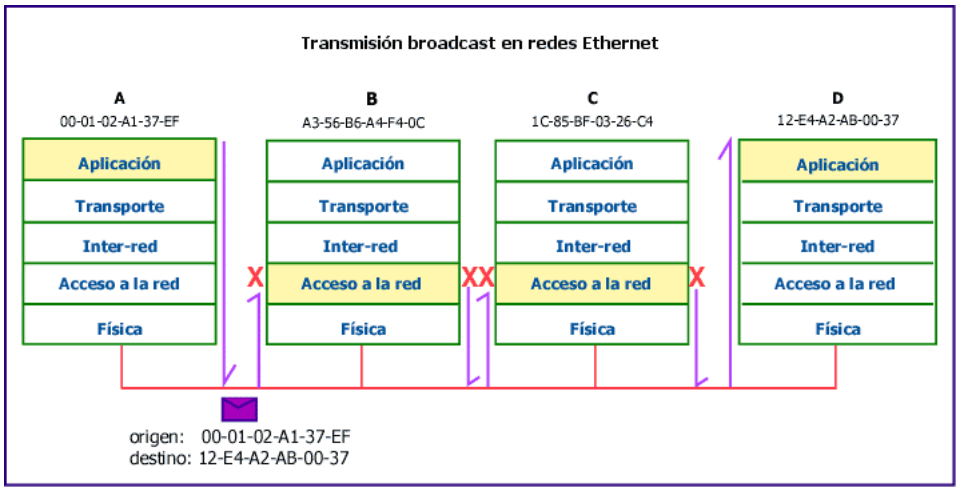
Expliquemos cómo funciona el protocolo de acceso al medio[5] CSMA/CD. Son las siglas de *Carrier Sense, Multiple Access with Collision Detect*[6].

El problema que pretende resolver es cómo acceden a un medio compartido (el cable de red) varios ordenadores. Otro protocolo de acceso al medio es el Paso de Testigo[7], en donde solamente la estación que disponga del token (testigo) puede transmitir en ese momento. Como en una carrera de relevos.

↬ El protocolo CSMA/CD funciona de la siguiente forma. Cuando un ordenador necesita transmitir información, la tarjeta de red (que es donde se implementa el protocolo, como dijimos antes) escucha en el cable, de forma parecida a como los indios del Oeste Americano escuchaban si venía un tren por la vía, pegando la oreja al raíl. Si está pasando información, el medio está ocupado y tiene que esperarse durante un tiempo aleatorio. Si no pasa información en ese momento, puede transmitir. Hasta aquí la parte de Acceso Múltiple con Detección de Portadora. Poner la oreja y ver que no pasa nadie.

Puede ocurrir que dos estaciones hayan hecho lo mismo: escuchar, ver que no emitía nadie, y emitir ellas, produciéndose lo que se conoce como **colisión**. Para resolver el conflicto, lo que hace cada estación es emitir y quedarse escuchando el momento posterior, por si se ha producido una colisión. Si se ha producido una colisión, cada estación da por no emitida la trama y espera un tiempo aleatorio[8] para volver a intentarlo.

[5] Estos protocolos se encargan de arbitrar quién puede acceder al medio físico en un momento determinado.
[6] En castellano: Acceso Múltiple con Detección de Portadora y Detección de Colisiones.
[7] Token. Utilizado en entornos específicos, con requerimientos muy exigentes de tiempo real, como los de producción industrial. Ejemplos: token ring, token bus.
[8] El tiempo de espera debe ser aleatorio para minimizar el que vuelvan a colisionar indefinidamente.



Como se observa en la figura, cuando la estación A ha emitido la trama, las demás estaciones de la red están escuchando si la comunicación va dirigida a ellas. En este caso, la dirección MAC de destino es la de la estación D, que será la única que reciba la trama. El resto de estaciones (B y C en este caso) descartarán la trama al nivel de la tarjeta de red[9].

4.2. Nivel de Red

4.2.1. Direccionamiento IP

Las direcciones TCP/IP (y aplicable a Internet) pueden ser simbólicas o numéricas. La forma simbólica es más fácil de leer y recordar por las personas, por ejemplo: *mileto.cica.es*. La forma numérica es un número binario sin signo de 32 bits, habitualmente expresado en forma de números decimales separados por puntos. Por ejemplo, *150.214.5.11* es una dirección numérica, que se refiere a la misma máquina que la dirección simbólica anterior. Ocurre lo mismo que con el nombre de una persona (Juan Pérez García) y su número de documento nacional de identidad (23.456.789-X). A las personas nos es más fácil recordar el nombre y sin embargo, las máquinas[10] trabajan mejor con el DNI.

La forma numérica es usada por las máquinas porque es más eficiente y es la que utilizan los protocolos de nivel de red. La función de correspondencia entre los dos tipos de direcciones, la simbólica y la numérica, la realiza el sistema de DNS (*Domain Name System*). A partir de ahora[11], nos referiremos a la forma numérica, cuando hablemos de dirección IP.

Las direcciones IP son números de 32 bits, habitualmente expresados como cuatro valores decimales[12] separados por puntos[13]. El formato binario para la dirección IP 128.2.7.9 es: 10000000 00000010 00000111 00001001[14].

Cada interfaz de una máquina conectada a una red TCP/IP tiene asignada una dirección IP. Por interfaz de red entendemos el dispositivo que nos une a la red, como una tarjeta de red o un módem.

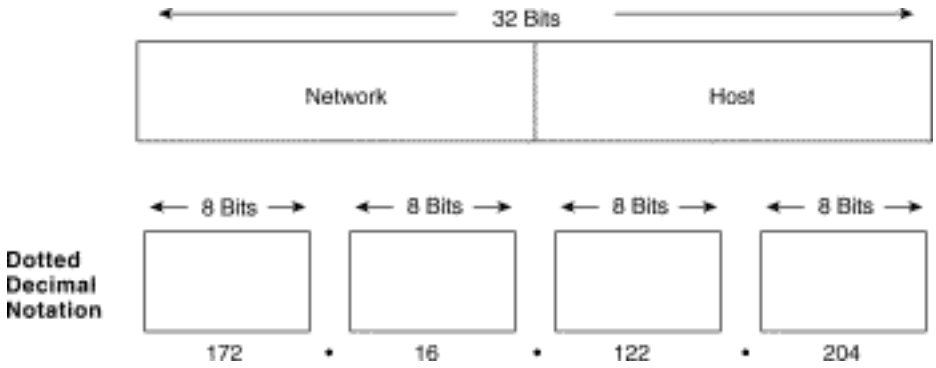
Dos nodos conectados a una misma red no pueden tener la misma dirección IP[15]. Todas las máquinas conectadas a una misma red poseen direciones IP con los primeros bits iguales (bits de

[9] Hay una manera de que la tarjeta funcione en modo promiscuo y escuche las tramas aunque no vayan dirigidas a ella. En este funcionamiento se basan los sniffers o "husmeadores" de la red.
[10] Sobre todo las de la Agencia Tributaria
[11] Hasta que abordemos el estudio del sistema DNS en la siguiente entrega.
[12] Cada número decimal, expresa 8 bits, con un rango del 0 al 255. $2^7, 2^6, 2^5, 2^4, 2^3, 2^2, 2^1, 2^0$
[13] Conocida como notación decimal separada por puntos (dotted decimal notation).
[14] $(2^7 =)128.(2^1 =)2.(2^2 + 2^1 + 2^0 =)7.(2^3 + 2^0 =)9$
[15] Como si en una calle hubiera dos casas con el mismo número.

red), mientras que los bits restantes[16] son los que identifican a cada máquina concreta dentro de esa red.

↬ Como por ejemplo, el prefijo telefónico identifica la provincia y el resto de números identifica el teléfono concreto dentro de esa provincia.

Las máquinas que pertenecen a una misma red IP, pueden comunicarse directamente unas con otras. Para comunicarse con máquinas de una red IP diferente, deben hacerlo mediante mecanismos de interconexión como routers, proxys o gateways.



A la parte de la dirección IP que es común a todas las direcciones que se encuentran en una red IP, se le llama la parte de la red[17] (network). Los bits restantes son llamados la parte de puesto[18] (o de host).

El tamaño de la parte dedicada al puesto depende del tamaño de la red. Entre estas dos partes deben completar los 32 bits. Para satisfacer diferentes necesidades, se han definido varias clases de redes[19], fijando diferentes sitios donde dividir la dirección IP. Las clases de redes se dividen en las siguientes:

Clase A: Comprende redes desde 1.0.0.0 hasta 127.0.0.0. El número de red está contenido en el primer octeto (byte). Esta clase ofrece una parte para el puesto de 24 bits, permitiendo aproximadamente 1,6 millones de puestos por red.

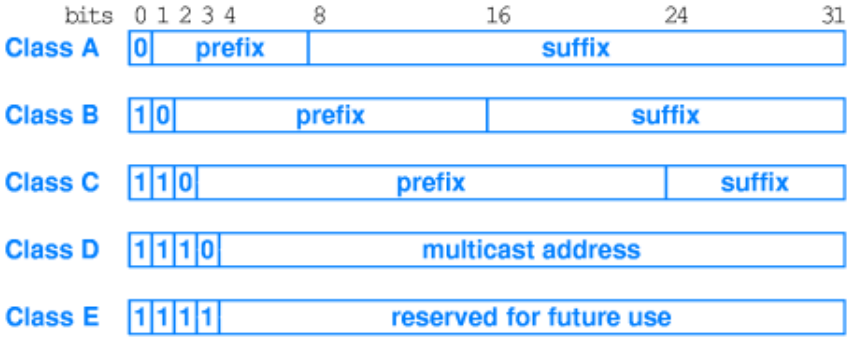
Clase B: Comprende las redes desde 128.0.0.0 hasta 191.255.0.0; el número de red está en los dos primeros octetos. Esta clase permite 16.320 redes con 65.024 puestos cada una.

Clase C: Van desde 192.0.0.0 hasta 223.255.255.0, con el número de red contenido en los tres primeros octetos. Esta clase permite cerca de 2 millones de redes con 254[20] puestos cada una.

Clases D, E, y F: Las direcciones que están en el rango de 224.0.0.0 hasta 254.0.0.0 son experimentales o están reservadas para uso con propósitos especiales. Por ejemplo, IP Multicast, un servicio que permite trasmitir información a muchos puntos en una internet a la vez, tiene direcciones dentro de este rango.

Las redes de clase A se distinguen porque comienzan por 0, las de clase B por 01, las de clase C por 110, las de clase D por 1110 y las de clase E por 1111.

[16] Hasta completar los 32.
[17] Estos bits, se encuentran situados en la parte de más a la izquierda
[18] Los bits situados a la derecha.
[19] Esta clasificación de redes no tiene mucha utilidad actualmente, pero se incluye como reseña histórica y mecanismo de comprensión de los conceptos de *parte de la red* y *parte del puesto*.
[20] Más adelante veremos de dónde sale este número.



Sin embargo, estas clases no deben ya tomarse al pie de la letra, porque lo normal es que se unan o dividan redes según las necesidades de cada organización.

Al número de bits que comparten todas las direcciones de una red se le llama *máscara de red* (netmask), y su papel es determinar qué direcciones pertenecen a la red y cuáles no. Veámoslo con un ejemplo.

	Decimal	Binario
Dirección de puesto	192.168.1.5	11000000.10101000.00000001.00000101
Máscara de red	255.255.255.0	11111111.11111111.11111111.00000000
Parte de red	192.168.1.	11000000.10101000.00000001.
Parte de puesto	.5	.00000101
Dirección de red	192.168.1.0	11000000.10101000.00000001.00000000
Dirección de difusión	192.168.1.255	11000000.10101000.00000001.11111111

Una dirección de puesto IP (192.168.1.5) a la que se aplique una operación "and"[21] de bits con su máscara de red (255.255.255.0[22]), nos dará la dirección de la red a la que pertenece (192.168.1.0). La dirección de red será el menor número de dirección IP dentro del rango de la red porque tiene la parte de puesto toda con ceros.

Una red IP queda definida por la dirección de la red y la máscara de la red. La notación que se usa para referirnos a dicha tupla es DIRECCIÓN DE RED/MÁSCARA DE RED (192.168.1.0/255.255.255.0) o DIRECCIÓN DE RED/NÚMERO DE BITS DE RED (192.168.1.0/24[23]).

En esta red, para la parte de puesto han quedado 8 bits, que nos darán $2^8 = 256$ direcciones posibles. Siempre hay que quitar dos direcciones especiales, que son la dirección de la red (todo a ceros en la parte de host) y la dirección de broadcast (todo a unos en la parte de host). Entonces, quedarán 254[24] direccciones posibles para puestos dentro de la red.

La dirección de difusión (broadcast) es una dirección especial que escuchan todas las máquinas en esa red IP, además de a la suya propia. Esta dirección es a la que se envían los paquetes de datos si se supone que todas las máquinas de la red lo deben recibir.

↬ Por ejemplo, si en un centro se manda una circular para que todos los profesores acudan a una reunión; va dirigida a todos en general y a ninguno en particular.

Ciertos tipos de datos, como la información de encaminamiento y los mensajes de aviso son transmitidos a la dirección de difusión[25] para que cada estación en la red pueda recibirlo simul-

[21] ó "Y lógico"

[22] En este caso, 24 bits para la parte de red y 8 bits para la parte de puesto.

[23] 11111111.11111111.11111111.00000000 tiene 8x3=24 unos

[24] De aquí viene: 256-2=254

[25] Esta es la dirección de difusión de IP. No confundir con la difusión física en Ethernet. Cumplen parecida función, cada una en su nivel correspondiente.

táneamente. Para ello se utiliza la dirección más alta posible en la red, conseguida con la parte de red a la que se añade el resto de bits (de la parte de puesto) todos con valor a uno (1). En el ejemplo anterior sería 192.168.1.255.

Podemos partir las redes en otras más pequeñas[26], es lo que se conoce como *subnetting*. Por ejemplo, tomemos la red de clase C 172.26.1.0/24, en donde estarían incluidas las direcciones desde la 172.26.1.0 hasta la 172.26.1.255, incluyendo la dirección de red y la de difusión. Tendríamos como hosts, desde el 172.26.1.1 hasta el 172.26.1.254. Si hacemos subnetting, utilizando 25 bits para la red en vez de 24, hemos partido esta red en dos. Veamos cuáles son estas redes: la primera sería la red 172.26.1.0/25[27], que iría desde la dirección 172.26.1.0 (dirección de red) hasta la 172.26.1.127 (que seria la dirección de broadcast). Siendo los nodos desde el 172.26.1.1 hasta el 172.26.1.126, en total 126 nodos posibles. La segunda red sería la 172.26.1.128/25, con los datos que aparecen en la tabla.

Red	Broadcast	Primer host	Último host
172.26.1.0/255.255.255.128	172.26.1.127	172.26.1.1	172.26.1.126
172.26.1.128/255.255.255.128	172.26.1.255	172.26.1.129	172.26.1.254

A partir de la dirección IP de destino, un nodo de la red puede determinar si los datos deben ser enviados a través de un router o un gateway hacia el exterior de la red, porque pertenezcan a una red distinta a la suya. Si los bytes correspondientes a la parte de red de la dirección IP de destino, son los mismos que la parte de red de la dirección IP del host origen, los datos no se pasarán al router, porque están en la misma red; si son diferentes, se pasarán a un router, para que los encamine hacia el exterior de la red. En este caso, el router tendrá que determinar el camino de enrutamiento idóneo en base a la dirección de la red de destino de los paquetes y una tabla interna que contiene la información de enrutamiento.

Las direcciones IP son usadas por el protocolo IP para definir únicamente un host en la red. Los datagramas IP (los paquetes de datos elementales intercambiados entre máquinas) se transmiten a través de alguna red física[28] conectada a la interfaz de la máquina y cada uno de ellos contiene la dirección IP de origen y la dirección IP de destino. Para enviar un datagrama a una dirección IP de destino determinada, la dirección de destino debe ser traducida o mapeada a una dirección física. Esto puede requerir transmisiones en la red para encontrar la dirección física de destino (por ejemplo, en una Red de Área Local, el protocolo ARP (*Address Resolution Protocol*), se usa para traducir las direcciones IP a direcciones físicas MAC).

Direcciones de red reservadas.

Desde el punto de vista de su accesibilidad, podemos clasificar las direcciones IP en:

- Direcciones IP públicas: aquellas que son visibles por todos los puestos conectados a Internet. Para que una máquina sea visible desde Internet, debe tener asignada obligatoriamente una dirección IP pública, y no puede haber dos puestos con la misma dirección IP pública.
- Direcciones IP privadas: aquellas que son visibles únicamente por los puestos de su propia red privada. Los puestos con direcciones IP privadas no son visibles desde Internet, por lo que si quieren salir a ésta deben hacerlo a través de un router[29] o un proxy (o intermediario). Las direcciones IP privadas se utilizan en redes de empresas, organismos, centros para interconectar los puestos de trabajo. Se definen en el RFC[30] 1918, que ha convertido en obsoleto al RFC 1597.

[26] O agruparlas, para formar otras más grandes.
[27] La máscara sería 11111111.11111111.11111111.10000000, o 255.255.255.128
[28] Recordemos que una red IP es una red lógica
[29] U otro dispositivo con capacidad de realizar NAT, que básicamente es una traducción de direcciones. La dirección IP privada es traducida a una IP pública en este dispositivo para que pueda salir a Internet.
[30] Los RFC (*Request For Comments*) son documentos técnicos que definen los estándares de Internet.

Si estamos construyendo una red privada y no tenemos intención de conectar nunca esa red a Internet, entonces podríamos elegir las direcciones que queramos, pues no vamos a colisionar con nadie, sin embargo, es buena práctica utilizar direcciones privadas por si en el futuro nos conectáramos. Estas direcciones son:

Dirección	Bits de red	Máscara de red	Número de redes
10.0.0.0	8	10.255.255.255	1 de clase A
172.16.0.0	12	172.31.255.255	16 de clase B
192.168.0.0	16	192.168.255.255	256 de clase C

Estas direcciones no se corresponden con las de ninguna máquina en Internet y no se encaminarán a través de los "routers" de Internet. Podremos utilizarlas de forma interna, con la ventaja de que si conectamos mediante un proveedor a Internet, nuestras direcciones no coincidirán con las de ninguna máquina de Internet.

En este caso, puede que haya multitud de equipos en redes distintas con estas mismas direcciones IP, pero como estas direcciones no se "rutean" no hay por qué coordinar su uso. En el caso de que se conecten a Internet, se habilitan mecanismos como la traducción de direccciones (NAT) o el uso de intermediarios (proxys) para que las direcciones que salgan a Internet sean direcciones válidas.

También la dirección 127.0.0.1, denominada de bucle local (*loopback*), es una dirección especial, que como veremos, utiliza la propia máquina para acceder a sus procesos locales.

4.2.2. Protocolo ARP

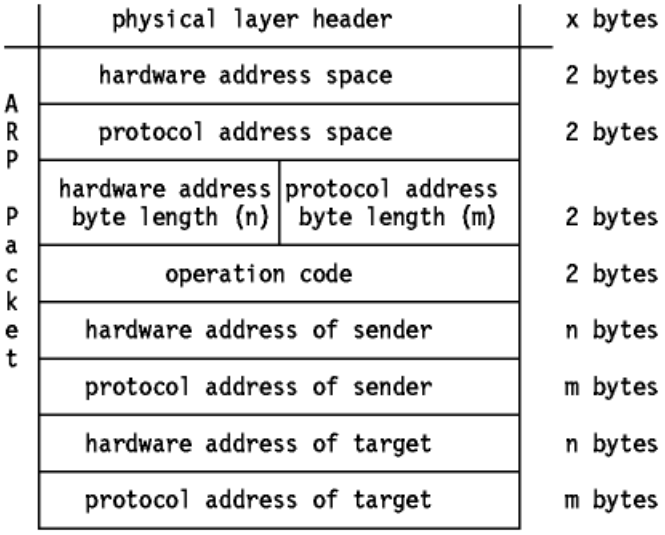
En una red física, los hosts individuales se conocen en la red a través de su dirección física[31]. Los protocolos de más alto nivel[32] direccionan a los hosts de destino con una dirección lógica[33]. Cuando se quiere enviar un mensaje a una dirección IP de destino que se encuentra en nuestra red, 172.26.1.5, el manejador de dispositivo físico no sabe a qué dirección MAC enviarla.

Para resolver este problema, se suministra un protocolo (el ARP) que traducirá la dirección IP a la dirección física del host de destino. No es un protocolo que transporte datos, sino que tiene un propósito específico: **asociar direcciones lógicas con direcciones físicas**.

Existe un protocolo, el RARP (Reverse ARP) cuyo propósito es el complementario: sabiendo una dirección física, obtener la dirección lógica que le corresponde. Se utiliza cuando tenemos un dispositivo (una impresora, por ejemplo) que debe arrancar e integrarse en la red con una dirección lógica, y lo único que conoce es su dirección física. En este caso debe existir un servidor (con el protocolo DHCP o bootp) que se encargue de gestionar estas asignaciones.

Comentemos cómo funciona el protoolo ARP. Utiliza una tabla (llamada caché ARP[34]) para realizar esta traducción. Cuando la dirección física no se encuentra en la caché ARP, se envía un broadcast a la red[35], con un formato especial llamado petición ARP. Si una de las máquinas en la red reconoce su propia dirección IP en la petición, devolverá una respuesta ARP al host que la solicitó. La respuesta contendrá la dirección física del hardware, así como información de encaminamiento (si el paquete ha atravesado puentes[36] durante su trayecto). Tanto esta dirección como la ruta se almacenan en la tabla caché ARP del host solicitante. Todos los posteriores datagramas enviados a esta dirección IP se podrán asociar a la dirección física correspondiente.

[31] Como la dirección MAC.
[32] Del nivel de red hacia arriba.
[33] en este caso, la dirección IP
[34] Que contiene correspondencias entre direcciones lógicas y direcciones físicas.
[35] Correcto, lo recibirán todas las máquinas de esa red.
[36] Los puentes (bridges) conectan varias redes físicas que pertenecen a la misma red lógica. Trabajan a nivel físico y de enlace.



Este es el formato del paquete ARP, donde tras la cabecera de la capa física vienen:

- *Hardware address space*: Especifica el tipo de hardware; ejemplos son Ethernet o Packet Radio.
- *Protocol address space*: Especifica el tipo de protocolo, el mismo que en el campo de tipo EtherType en la cabecera de IEEE 802.
- *Hardware address length*: Especifica la longitud (en bytes) de la dirección hardware del paquete. Para Ethernet e IEEE 802.3 será de 6 bytes.
- *Protocol address length*: Especifica la longitud (en bytes) de la dirección lógica. Para IP será de 4 bytes.
- *Operation code*: Especifica el tipo de operación ARP.
- *Source/target hardware address*: Contiene las direcciones físicas hardware. En IEEE 802.3, son direcciones de 48 bits.
- *Source/target protocol address*: Contiene las direcciones lógicas. En TCP/IP son direcciones IP de 32 bits.

Para el paquete de solicitud, la dirección hardware de destino es el único campo indefinido del paquete, que es el que pretende obtener.

El host solicitante recibirá la respuesta ARP en caso de que la máquina buscada se encuentre en la red, y seguirá el proceso ya comentado para tratarla. Como resultado, la tripleta `<tipo de protocolo, dirección de protocolo, dirección hardware>` para el host en cuestión se añadirá a la caché ARP. La próxima vez que un protocolo de nivel superior quiera enviar un paquete a ese host, el módulo de ARP ya tendrá la dirección hardware, a la que se enviará el paquete.

↬ **Ejemplo:** Mediante el comando `arp` podemos mostrar la caché arp en un sistema linux.

```
[root@linux entrega05-1]# arp -e
Address HWtype HWaddress Flags Mask Iface
172.26.0.1 ether 00:01:38:11:D6:03 C eth0
```

4.2.3. Protocolo IP

IP es el protocolo que oculta la red física subyacente creando una vista de red virtual[37]. Es un protocolo de entrega de paquetes no fiable y no orientado a conexión, y se puede decir que aplica la ley del mínimo esfuerzo.

No aporta fiabilidad, control de flujo o recuperación de errores. Los paquetes (datagramas) que envía IP se pueden perder, desordenarse, o incluso duplicarse, e IP no manejará estas situaciones. Proporcionar estos servicios depende de protocolos de capas superiores.

IP asume pocas cosas de las capas inferiores, sólo que los datagramas "probablemente" serán transportados a la estación de destino.

La versión actual es la IP versión 4 (IPv4). Desde mediados de los años 90 se escucha que será sustituida por la nueva generación IPv6, pero diversos factores han hecho que no se haya producido dicha sustitución.

1.- IPv4 ha ido desarrollando mecanismos para resolver sus deficiencias, como el agotamiento de direcciones IP contrarrestado con el uso de direccionamiento privado, o la aparición de protocolos como SSL para añadirle la seguridad de la que carecía.

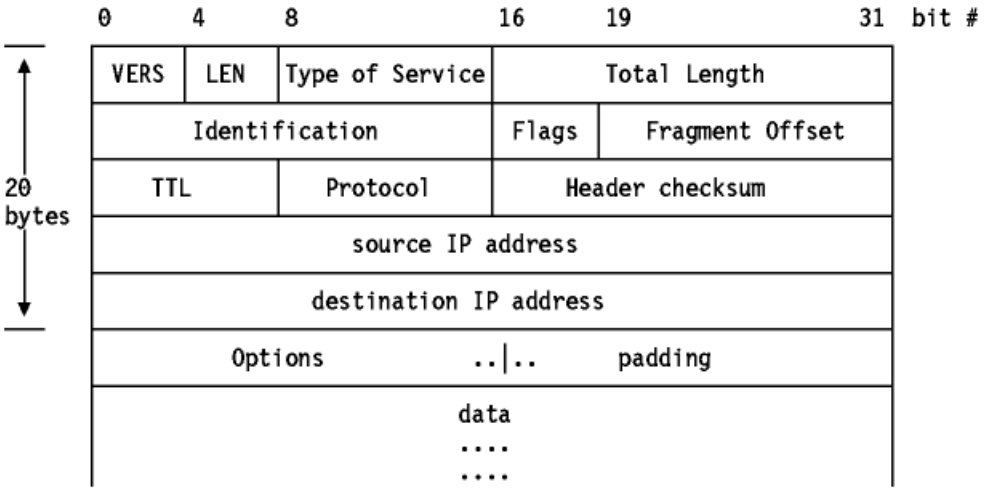
2.- IPv6 ha tardado en desarrollarse y probarse más de lo esperado, además de no proveer mecanismos suaves de evolución. Por ejemplo, al pasar de redes Ethernet a 10Mbps a redes de 100Mbps ó de 1000Mbps, el cambio en la mayoría de los sitios casi ha sido imperceptible.

3.- Las empresas y organismos han sentido pereza ante el cambio: si lo que tengo me funciona, porqué voy a cambiarlo, unido a la mayor complejidad de IPv6 que necesitará una formación y entrenamiento.

Hablaremos por tanto, todavía del "anciano" IPv4 y en el apéndice 8, avanzaremos conceptos sobre IPv6.

Estructura del datagrama IP El datagrama IP es la unidad de transferencia en la pila IP. Tiene una cabecera con información para IP, y su carga de datos para los protocolos superiores.

La cabecera del datagrama IP es de un mínimo de 20 bytes de longitud:



En la figura se muestra la estructura del datagrama IP, donde:

- *VERS*: La versión del protocolo IP. La versión actual es la 4. La 5 es experimental y la 6 es la nueva generación IPv6[38].

[37] o red lógica.

[38] Véase 8 en la página 123

- *LEN*: La longitud de la cabecera IP contada en cantidades de 32 bits. No incluye el campo de datos.
- *Type of Service*: El tipo de servicio es una indicación de la calidad del servicio solicitado para este datagrama IP.
- *Total Length*: La longitud total del datagrama, cabecera y datos, especificada en bytes.
- *Identification*: Un número único que asigna el emisor para ayudar a reensamblar un datagrama fragmentado. Los fragmentos de un datagrama tendrán el mismo número de identificación.
- *Flags*: Varios flags de control. Donde:

  0 está reservado, debe ser cero

  DF: No fragmentar (*Don't Fragment*): con 0 se permite la fragmentación, con 1 no.

  MF: Más fragmentos (*More fragments*): 0 significa que se trata del último fragmento del datagrama, 1 que no es el último.

- *Fragment Offset*: Usado con datagramas fragmentados, para ayudar al reensamblado de todo el datagrama. El valor es el número de partes de 64 bits (no se cuentan los bytes de la cabecera) contenidas en fragmentos anteriores. En el primer (o único) fragmento el valor es siempre cero.
- *Time to Live*: Especifica el tiempo (en saltos de router) que se le permite viajar a este datagrama. Cada "router" por el que pase este datagrama ha de sustraer uno de este campo. Cuando el valor alcanza cero, se asume que este datagrama ha estado viajando sin llegar a su destino por alguna razón y se desecha. El valor inicial lo deberá fijar el protocolo de alto nivel que crea el datagrama. Es una forma de eliminar los paquetes zombies vagando eternamente por la red.
- *Protocol Number*: Indica el protocolo de alto nivel al que IP deberá entregar los datos del datagrama. Algunos valores son:

  0 Reservado

  1 ICMP (*Internet Control Message Protocol*)

  2 IGMP (*Internet Group Management Protocol*)

  3 GGP (*Gateway-to-Gateway Protocol*)

  4 IP (IP encapsulation)

  5 Flujo (*Stream*)

  6 TCP (*Transmission Control*)

  8 EGP (*Exterior Gateway Protocol*)

  17 UDP (*User Datagram*)

  89 OSPF (*Open Shortest Path First*).

- *Header Checksum*: Es el checksum de la cabecera. Si su comprobación no es válida, el datagrama se desecha, ya que al menos un bit de la cabecera está corrupto, y el datagrama podría haber llegado al destino equivocado. No tenemos la seguridad de que lo que esté mal pueda ser la dirección de destino.

- *Source IP Address*: La dirección IP de 32 bits del host emisor.
- *Destination IP Address*: La dirección IP de 32 bits del host receptor.
- *Options*: No requiere que toda implementación de IP sea capaz de generar opciones en los datagramas que crea, pero sí que sea capaz de procesar datagramas que contengan opciones. El campo "Options" (opciones) tiene longitud variable. Puede haber cero o más opciones.

➥ **Ejemplos:**

1. El **comando ping** se utiliza para comprobar la conectividad a nivel IP entre nuestro sistema y un sistema destino. Por ejemplo, desde la máquina mileto.cica.es hasta el servidor web del Instituto Tecnológico de Massachussets (MIT).

```
[root@mileto root]# ping www.mit.edu
PING DANDELION-PATCH.mit.edu (18.181.0.31) from 150.214.5.11 : 56(84) bytes of data.
64 bytes from DANDELION-PATCH.MIT.EDU (18.181.0.31): icmp_seq=1 ttl=243 time=126 ms
64 bytes from DANDELION-PATCH.MIT.EDU (18.181.0.31): icmp_seq=2 ttl=243 time=127 ms
64 bytes from DANDELION-PATCH.MIT.EDU (18.181.0.31): icmp_seq=3 ttl=243 time=126 ms
64 bytes from DANDELION-PATCH.MIT.EDU (18.181.0.31): icmp_seq=4 ttl=243 time=126 ms
64 bytes from DANDELION-PATCH.MIT.EDU (18.181.0.31): icmp_seq=5 ttl=243 time=128 ms
64 bytes from DANDELION-PATCH.MIT.EDU (18.181.0.31): icmp_seq=6 ttl=243 time=126 ms
--- DANDELION-PATCH.mit.edu ping statistics ---
6 packets transmitted, 6 received, 0% loss, time 5053ms
rtt min/avg/max/mdev = 126.358/126.871/128.440/0.799 ms
```

2. El **comando traceroute** nos indica los routers o pasarelas por las que tiene que atravesar un paquete para llegar desde nuestro host hasta el destino que especifiquemos. Para ello, utiliza una pequeña argucia utilizando el valor del campo TTL del paquete IP. Envía primero un paquete al sistema destino con un valor de 1 en el campo TTL. El router al que llega descuenta 1 y al llegar el valor TTL a 0, devuelve un paquete de error ICMP. El siguiente paquete posee un valor de 2 en el TTL, con lo cual llegará a 0 al pasar por el segundo router, que será el que envíe el paquete de error. Así se descubre la ruta que siguen nuestros paquetes hasta el sistema destino.

   Por ejemplo, desde mileto.cica.es hasta www.mit.edu pasa por los siguientes routers:

```
[root@mileto root]# traceroute www.mit.edu
traceroute to DANDELION-PATCH.mit.edu (18.181.0.31), 30 hops max, 38 byte packets
1 150.214.5.28 (150.214.5.28) 0.622 ms 0.466 ms 0.460 ms
2 rt1 (150.214.3.4) 0.460 ms 0.606 ms 0.460 ms
3 GE0-1-0.EB-Sevilla0.red.rediris.es (130.206.194.1) 0.601 ms 0.763 ms 0.460 ms
4 AND.SO4-1-0.EB-IRIS2.red.rediris.es (130.206.240.17) 8.037 ms 8.036 ms 8.104 ms
5 SO0-0-0.EB-IRIS4.red.rediris.es (130.206.240.2) 8.106 ms 8.066 ms 8.134 ms
6 rediris.es1.es.geant.net (62.40.103.61) 8.332 ms 8.339 ms 8.312 ms
7 es.it1.it.geant.net (62.40.96.186) 30.908 ms 31.126 ms 30.848 ms
8 it.de2.de.geant.net (62.40.96.61) 40.035 ms 40.076 ms 40.018 ms
9 abilene-gw.de2.de.geant.net (62.40.103.254) 133.684 ms 133.718 ms 133.687 ms
10 nycmng-washng.abilene.ucaid.edu (198.32.8.84) 120.839 ms 121.198 ms 120.859 ms
11 ATM10-420-OC12-GIGAPOPNE.nox.org (192.5.89.9) 125.835 ms 125.970 ms 125.919 ms
12 192.5.89.90 (192.5.89.90) 125.996 ms 126.120 ms 126.059 ms
13 NW12-RTR-2-BACKBONE.MIT.EDU (18.168.0.21) 126.043 ms 126.590 ms 126.435 ms
14 DANDELION-PATCH.MIT.EDU (18.181.0.31) 126.536 ms * 130.035 ms.
```

4.3. Nivel de transporte: TCP

4.3.1. Puertos y Sockets

Puertos

Lejos quedan aquellos tiempos del MS-DOS (por ejemplo) en los que en un ordenador se podía hacer una sola cosa a la vez. Afortunadamente, hoy podemos estar escuchando música con el ordenador, a la vez que navegamos por Internet y realizamos complejos cálculos. De esta forma, cuando una comunicación llega a nuestro ordenador, además debe saber a qué proceso va dirigida.

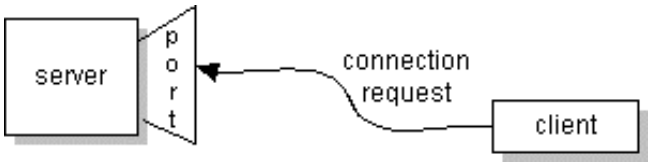
Un puerto es un número de 16 bits (de 1 a 65.535), empleado para identificar a qué protocolo del nivel superior o programa de aplicación se deben entregar los mensajes recibidos. Podemos asimilarlo a un número de atraque donde puede situarse un barco dentro de un puerto[39] .

Hay números de puerto que por convención se asignan a determinadas aplicaciones y/o protocolos[40], como el 80 para el http, el 23 para telnet o el 25 para smtp.

Sockets

En castellano la palabra *socket* significa enchufe, y esa es precisamente la función que realiza: un enchufe que conecta dos procesos. Un socket está formado por una dirección IP y un puerto. Un ejemplo, en el formato *dirección IP:puerto*, sería 172.26.0.2:80.

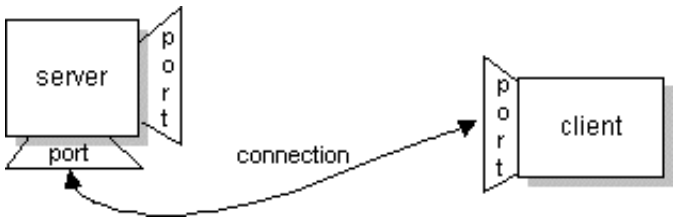
Cuando un proceso servidor se inicia, abre un socket en la máquina y se pone a esperar a que alguien se conecte.



Del mismo modo, cuando un proceso cliente quiere conectarse, abre otro socket en su máquina. La conexión se establece cuando ambos sockets se conectan, formando la tupla

DirecciónOrigen:PuertoOrigen⟶DirecciónDestino:PuertoDestino

Indicamos con la flecha la dirección del cliente al servidor, o más concretamente, desde quién ha solicitado la apertura de la conexión hasta quién la ha recibido.

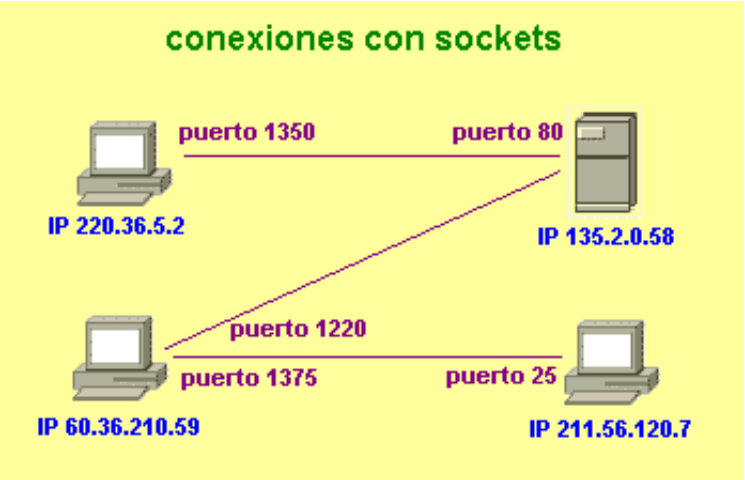


En la figura siguiente hay tres conexiones. Vamos a identificarlas, empezando por las que están situadas más arriba. Suponemos para el ejemplo que las máquinas cliente se sitúan a la izquierda y las máquinas servidoras a la derecha.

La primera conexión sería 220.36.5.2:1350⟶135.2.0.58:80. Probablemente sea una conexión a un servidor web[41], ya que el puerto 80 normalmente es utilizado para ello. El servidor web primero abre el socket 135.2.0.58:80, al que se conecta el cliente desde el socket 220.36.5.2:1350. Con este mecanismo, cada conexión tiene una identificación única. Si el cliente abriera una nueva conexión

[39] de los de verdad, con barcos.
[40] En el fichero `/etc/services` podemos encontrar un gran número de ellos
[41] Pero no tenemos la total seguridad, los puertos son convenciones que pueden respetarse o no.

con el mismo servidor web, su identificación sería 220.36.5.2:1351⟶135.2.0.58:80. Observamos que el puerto del cliente ha cambiado (de 1350 a 1351) y las dos conexiones tienen identificaciones distintas. Cuando un cliente abre un nuevo socket, se le asigna un número de puerto más alto y que no esté ocupado. Si llegara al más alto, empezaría de nuevo por el más bajo que esté libre.



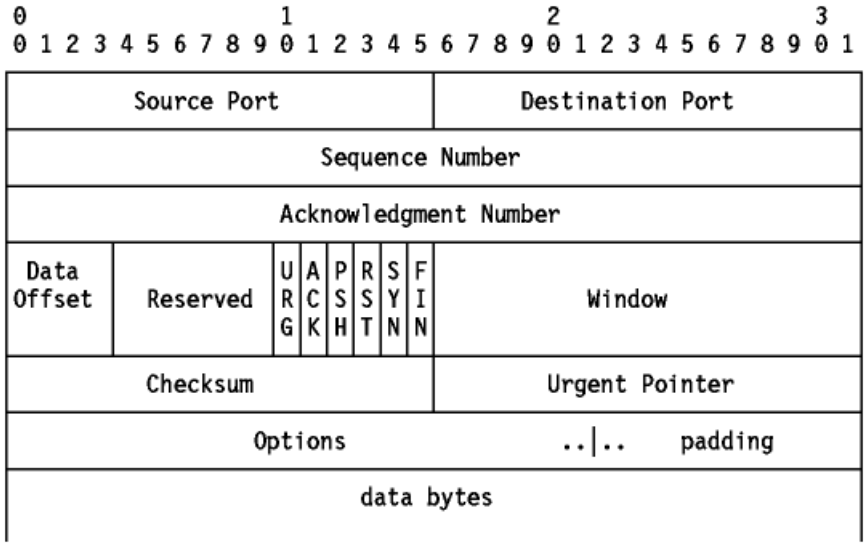
La siguiente conexión de la figura sería: 60.36.210.59:1220⟶135.2.0.58:80, correspondiente a otra máquina cliente conectándose al mismo servidor web de antes. Como vemos, el servidor web y los clientes pueden identificar perfectamente cada una de las conexiones.

La última conexión, definida por 60.36.210.59:1375⟶211.56.120.7:25, podría ser a un servidor de correo SMTP (porque ese es el número de puerto al que se conecta).

4.3.2. Protocolo TCP

El principal propósito de TCP es proporcionar una conexión lógica fiable entre dos procesos. Asume que los protocolos de niveles inferiores (como IP) no garantizan la fiabilidad, por lo que debe ocuparse de garantizarla.

Formato de segmento TCP:



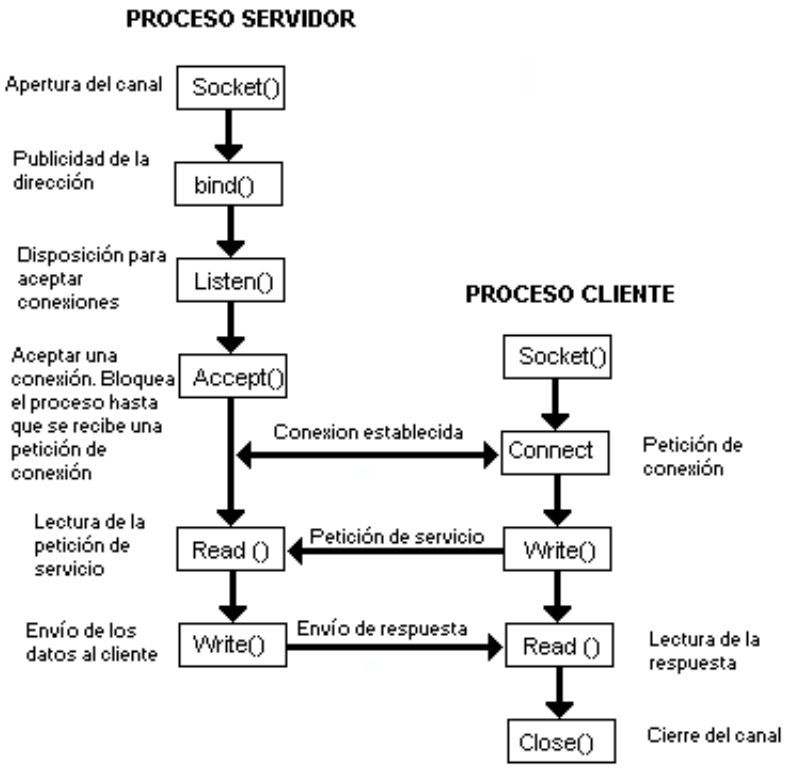
Esta es la estructura de un segmento TCP, donde:

- *Source Port*: Es el número de puerto de 16 bits del emisor, que el receptor usa para responder.
- *Destination Port*: El número de puerto de 16 bits del receptor.
- *Sequence Number*: El número de secuencia del primer byte de datos del segmento. Si el byte de control SYN está a 1, el número de secuencia es el inicial (n) y el primer byte de datos será el n+1.
- *Acknowledgment Number*: Si el bit de control ACK está a 1, este campo contiene el valor del siguiente número de secuencia que se espera recibir.
- *Data Offset*: El número de palabras de 32 bits de la cabecera TCP. Indica dónde empiezan los datos.
- *Reserved*: Seis bits reservados para su uso futuro; deben ser cero.
- *URG*: Indica que el campo *urgent pointer* es significativo en el segmento.
- *ACK*: Indica que el campo de reconocimiento (acuse de recibo) es significativo en el segmento.
- *PSH*: Función *Push*.
- *RST*: Resetea la conexión.
- *SYN*: Sincroniza los números de secuencia.
- *FIN*: No hay más datos del emisor.
- *Window*: Usado en segmentos ACK (de acuse de recibo). Especifica el número de bytes de datos que el receptor está dispuesto a aceptar hasta que le llegue el siguiente acuse de recibo.
- *Checksum:* Campo de 16 bits que permite verificar tanto la cabecera como los datos TCP.
- *Urgent Pointer*: Apunta al primer octeto de datos que sigue a los datos importantes. Sólo es significativo cuando el bit de control URG está a uno.
- *Options:* Sólo para el caso de opciones en los datagramas IP.
- *Padding Bytes*: Todos a cero para rellenar la cabecera TCP a una longitud total que sea un múltiplo de 32 bits.

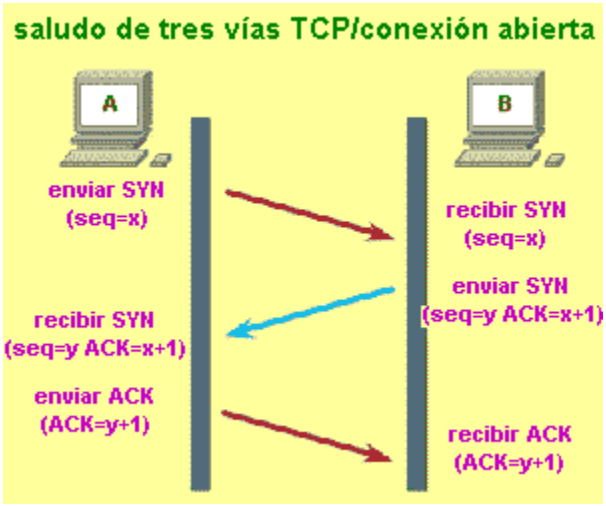
Como vemos, las direcciones IP de origen y destino no son necesarias, eso se ha quedado para el protocolo IP.

Estableciendo una conexión TCP

Antes de que se pueda transferir cualquier dato, se ha de establecer una conexión entre los dos procesos. Uno de los procesos (normalmente el servidor) lanza una llamada para abrir un socket pasivo. El servidor permanece en espera hasta que otro proceso intenta comunicarse con él a través de una solicitud de conexión. El proceso cliente lanza una llamada de apertura de socket y conexión al servidor.



En la red, en el momento de establecimiento de la conexión, se intercambian tres segmentos TCP:



Este proceso completo se conoce como *three-way handshake*, o acuerdo en tres fases. Notar que los segmentos TCP intercambiados incluyen los números de secuencia iniciales de ambas partes, para ser usados en posteriores transferencias. El cliente envía un paquete para iniciar la conexión con el bit SYN activo. Esta característica la utilizarán los cortafuegos, que veremos en una sección posterior, para denegar el establecimiento de conexiones TCP. Sin el envío de este bit SYN no se podrá establecer una nueva conexión. El servidor, en el paquete de vuelta realiza un acuse de recibo (ACK) del paquete anterior y envía a su vez un SYN al cliente. El tercer paquete consiste

en el envío del acuse de recibo por parte del cliente. A partir de ahí la conexión TCP se encuentra activa para que cliente y servidor se transfieran datos.

El cierre de la conexión se hace de forma implícita enviando un segmento TCP con el bit FIN activo. Como la conexión es *full duplex*[42], el segmento FIN sólo cierra la conexión en un sentido del canal. El otro proceso enviará los datos restantes si quedaran, seguidos de un segmento TCP en el que el bit FIN está activo. La conexión se borra (es decir, la información de estado en ambos extremos) una vez que el canal se ha cerrado en ambos sentidos.

➥ **Ejemplo:** Para ver los sockets y conexiones de nuestro sistema, utilizamos la **orden *netstat*.** En el comando siguiente, le indicamos que nos muestre todos los sockets (opción `-a`, los conectados y los que están a la escucha), tanto del protocolo tcp (opción `t`) como del protocolo udp (opción `u`).

```
[root@mileto root]# netstat -atu
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address Foreign Address State
tcp 0 0 *:7937 *:* LISTEN
tcp 0 0 *:http *:* LISTEN
tcp 0 0 *:ssh *:* LISTEN
tcp 0 0 *:smtp *:* LISTEN
tcp 0 0 *:https *:* LISTEN
tcp 0 0 mileto.cica.es:ssh 106.Red-81-44-35.p:1394 ESTABLISHED
tcp 0 0 mileto.cica.es:19094 dolmene.cica.es:7945 ESTABLISHED
udp 0 0 localhost.localdo:32769 localhost.localdo:32769 ESTABLISHED
udp 0 0 *:7938 *:*
udp 0 0 *:863 *:*
```

Comentemos toda la información interesante que nos presenta. En las primeras cinco líneas, vemos que hay cinco procesos servidores esperando conexiones en los puertos de http, https, smtp, ssh y 7937. Los que reciben un nombre, por ejemplo http (que es el 80) es porque han encontrado una línea en el fichero `/etc/services` y nos muestra ese nombre. El puerto 7937 no ha tenido esa suerte y no ha sido "bautizado". Habría que mirar qué proceso es el que se encuentra ahí escuchando para quedarnos tranquilos:

```
[root@mileto root]# fuser -n tcp 7937
7937/tcp: 879
```

Con el comando `fuser` podemos identificar procesos que utilizan ficheros o sockets (al fin y al cabo todo en Unix/Linux son ficheros, hasta los sockets). La opción `-n tcp` le indica que busque en el espacio de tcp[43]. En este caso nos ha dicho que el socket tcp que escucha en el puerto 7937 le corresponde al proceso 879. Pues vamos a mirar qué proceso es ese.

```
[root@mileto root]# ps aux|grep 879
root 879 0.0 0.1 3008 996 ? S Jan18 0:00 /usr/sbin/nsrexec
```

¡Ah!, resulta que es el proceso que hace las copias de seguridad de la máquina (el Legato Networker). Podemos estar tranquilos por dos razones, porque no es un proceso que un intruso nos ha colocado en nuestra máquina[44] y además, vuestras prácticas[45] no se perderán aunque el disco duro se rompa. Las recuperaremos de la copia de seguridad ;-).

[42] En ambos sentidos simultáneamente.
[43] Otras opciones serían *udp* o *file* para ficheros.
[44] Aunque para eso deberíamos asegurarnos de que el programa es realmente quien dice ser y al terminar el curso seguro que estaréis en condiciones de saberlo.
[45] Cuando las coloquéis.

4.4. Nivel de Aplicación

Como ejemplo de nivel de aplicación, veremos el protocolo HTTP, utilizado entre los navegadores y los servidores web.

4.4.1. Protocolo HTTP

El Protocolo de Transferencia de HiperTexto (*Hypertext Transfer Protocol*) es un protocolo cliente-servidor que articula los intercambios de información entre los clientes Web y los servidores HTTP. La especificación completa del protocolo HTTP 1.0 está recogida en el RFC 1945. Fue propuesto por Tim Berners-Lee, atendiendo a las necesidades de un sistema global de distribución de información como el *World Wide Web*.

Desde el punto de vista de las comunicaciones, está soportado sobre los servicios de conexión TCP e IP, y funciona de la misma forma que el resto de los servicios comunes de los entornos UNIX: un proceso servidor[46] escucha en un puerto de comunicaciones TCP (por defecto, el 80), y espera las solicitudes de conexión de los clientes Web. Una vez que se establece la conexión, el protocolo TCP se encarga de mantener la comunicación y garantizar un intercambio de datos libre de errores.

HTTP se basa en sencillas operaciones de solicitud/respuesta. Un cliente establece una conexión con un servidor y envía un mensaje con los datos de la solicitud. El servidor responde con un mensaje similar, que contiene el estado de la operación y su resultado. Todas las operaciones pueden adjuntar un objeto o recurso sobre el que actúan; cada objeto Web (documento HTML, fichero multimedia) es conocido por su URL[47].

Cada vez que un cliente realiza una petición a un servidor, se ejecutan los siguientes pasos:

1. Un usuario accede a una URL, seleccionando un enlace de un documento HTML o introduciéndola directamente en el campo Dirección (*Location*) del cliente Web.

2. El cliente Web decodifica la URL, separando sus diferentes partes. Así identifica el protocolo de acceso (http), la dirección DNS o IP del servidor (`mileto.cica.es`), el puerto (como en este caso no aparece, toma el valor por defecto, que es 80) y el objeto requerido del servidor (en este caso el recurso raíz /).

3. Se abre una conexión TCP/IP con el servidor, llamando al puerto TCP correspondiente.

4. Se realiza la petición. Para ello, se envía el comando necesario (GET, POST, HEAD), la dirección del objeto requerido (el contenido de la URL que sigue a la dirección del servidor), la versión del protocolo HTTP empleada y un conjunto variable de información, que incluye datos sobre las capacidades del browser o datos opcionales para el servidor.

5. El servidor devuelve la respuesta al cliente. Consiste en un código de estado y el tipo de dato MIME[48] de la información de retorno, seguido de la propia información.

6. Cuando no hay más solicitudes, se cierra la conexión TCP.

➥ Veamos este proceso con un ejemplo:

1. Desde un navegador web, solicitamos la dirección `http://thales.cica.es`. El navegador (mozilla en este caso), se encarga de abrir una conexión tcp con el servidor web de la máquina `thales.cica.es` en el puerto 80. El navegador envía la solicitud siguiente, que cumple con el protocolo HTTP:

   ```
   GET / HTTP/1.1
   ```

[46] En los entornos Unix, el proceso servidor que se ejecuta en segundo plano, esperando conexiones de los clientes se denomina demonio (*daemon*). Profundizaremos en ello en la segunda entrega.

[47] Localizador Universal de Recursos. Por ejemplo: `http://mileto.cica.es`

[48] Identifica el tipo de objeto multimedia (audio, vídeo, texto...)

(es una solicitud GET, del objeto / y cumpliendo la versión HTTP/1.1)

```
Host: thales.cica.es
User-Agent: Mozilla/5.0 (X11; U; Linux i686; es-ES; rv:1.4.1) Gecko/20031030
Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,video/x-
mng,image/png,image/jpeg,image/gif;q=0.2,*/*;q=0.1
Accept-Language: es-es,es;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
```

2. El servidor `thales.cica.es` recibe la petición, la procesa y devuelve lo siguiente:

   ```
   HTTP/1.1 200 OK
   ```

   (El servidor devuelve en el protocolo HTTP/1.1 con el código de retorno 200, sin errores)

   ```
   Date: Sat, 10 Jan 2004 23:19:47 GMT
   Server: Apache/1.3.27 (Unix) (Red-Hat/Linux) mod_ssl/2.8.12 OpenSSL/0.9.6b DAV/1.0.2
   PHP/4.0.6 mod_perl/1.26
   Last-Modified: Fri, 19 Dec 2003 22:12:36 GMT
   ETag: "bd98c-1e9e-3fe377d4"
   Accept-Ranges: bytes
   Content-Length: 7838
   Content-Type: text/html
   Age: 46842
   ```

   (Aquí comienza el contenido HTML, con la cabecera y el cuerpo)

   ```
   <html>
   <head>
   <meta http-equiv="Content-Type"
   content="text/html; charset=iso-8859-1">
   <meta name="Author" content="Antonio Aranda Plata">
   <meta name="GENERATOR" content="Microsoft FrontPage Express 2.0">
   <title>S.A.E.M. THALES</title>
   </head>
   <body background="http://thales.cica.es./thales/BACK.GIF"
   bgcolor="#DDFFFE" text="#000000" link="#004091" vlink="#303031"
   alink="#3366FF" nosave>
   <p align="center"> </p>
   <div align="center"><center>
   <table border="0" cellpadding="0" cellspacing="0" width="100%"...
   ```

4.5. Ver para creer: Ethereal

La mejor forma de comprender todos estos conceptos es viéndolos, y ése va a ser nuestro objetivo, desmenuzar y ver cada uno de los protocolos.

Para ello, utilizaremos Ethereal. Ésta es una herramienta muy útil a la hora de comprobar el tráfico de la red con objeto de solucionar problemas. Contiene una serie de funcionalidades que nos permiten localizar rápidamente los problemas de nuestra red:

- Permite centrarnos en paquetes y protocolos concretos.
- Soporta un gran número de protocolos.

- Proporciona una visión clara del tráfico de nuestra red, proporcionando herramientas que nos permiten distinguir y aislar los paquetes que lo constituyen por distintos criterios.

Además, esta herramienta proporciona varios decodificadores[49] de protocolos que permiten realizar filtrados selectivos por protocolos. En total, Ethereal maneja 335 protocolos de red, entre los que se encuentran TCP, SMB, Telnet, LDAP o SNMP.

Ethereal también proporciona gráficos basados en el tráfico de red así como en el RTT[50]. Estos gráficos son fáciles de obtener y son interactivos, pulsando sobre un punto del gráfico podemos ver el paquete correspondiente en la ventana de análisis.

4.5.1. tcpdump

Realmente, Ethereal es una GUI de `tcpdump`[51], una herramienta de dominio público que imprime las cabeceras de los paquetes que pasan por una interfaz de red. Es posible ejecutar `tcpdump` en modo promiscuo, con lo que capturaremos los paquetes que viajen por la red. Así, podemos acceder a la distinta información que se almacena en los campos que constituyen un paquete. Es una herramienta que entraría dentro de las que denominamos como *sniffers* de red.

Tanto en la captura como en la visualización es posible aplicar filtros por protocolo (TCP, UDP, IP, ARP, RARP, ...), puertos (pueden ser números o los nombres según aparecen en `/etc/services`), direcciones fuente, direcciones destino, direcciones de red y, mediante el uso de operadores, una combinación de éstos.

Puede tambien usarse para comprobar los datos obtenidos en una captura realizada con anterioridad, pudiendo ser la fuente de estos datos otro programa[52]. Del mismo modo, la captura obtenida por `tcpdump` puede exportarse a un formato que puedan leer estos programas.

No entraremos en el detalle de cómo ejecutar esta utilidad ya que nos será más cómodo realizar la captura de paquetes utilizando ethereal, nos limitaremos a describir brevemente el uso básico de la misma, así como el significado de la salida que obtenemos.

Obteniendo datos con tcpdump

Hay ocasiones en que la máquina donde estamos no tiene instalado Ethereal. En este caso es mucho más cómodo realizar la captura en modo texto, volcándolo en un fichero, para su posterior análisis detallado con Ethereal en otra máquina.

Al realizar la captura con `tcpdump` hay que tener en cuenta que el resultado de la captura tendrá los paquetes truncados. Por defecto, `tcpdump` únicamente captura los primeros 68 *bytes* de cada paquete. La ejecución que haremos con `tcpdump` para evitar esto será de la forma:

```
tcpdump -i <interface> -s 1500 -w <fichero>
```

Con el tamaño de 1.500, nos aseguramos la captura de la totalidad del paquete en una red ethernet. La captura la finalizamos pulsando `^C`.

```
root@guadalinex:~# tcpdump -i eth0 -s 1500 -w /tmp/captura.dat
tcpdump: listening on eth0, link-
type EN10MB (Ethernet), capture size 1500 bytes
^C
88 packets captured
88 packets received by filter
0 packets dropped by kernel
```

[49] Ethereal los denomina *dissectors*.

[50] *Round Trip Time*

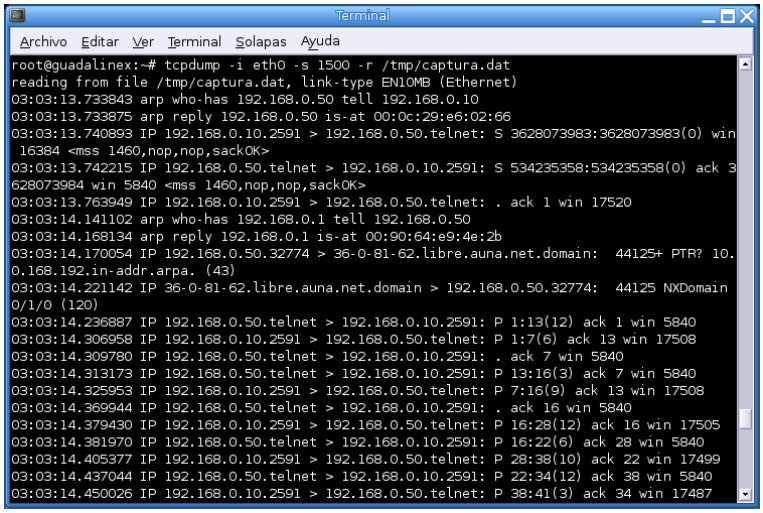
[51] Hay que tener en cuenta que `tcpdump` no viene con la distribución de Ethereal, es un software independiente. Puede obtenerse de `http://www.tcpdump.org`, o con nuestro socorrido `apt-get`.

[52] Además de las capturas realizadas con tcpdump, puede utilizar las realizadas por programas como Network Associates Sniffer y Sniffer Pro, LANanalyzer, Microsoft Network Monitor y otros. Mirar la documentación asociada a esta utilidad para obtener una lista de compatibilidad más amplia.

Interpretando la salida

El formato de la salida que proporcionan los datos capturados por `tcpdump` dependerá del protocolo, aunque todos tienen en común el primer campo que es una marca de tiempo que indica cuándo se realizó la captura.

Figura 4.1: Visualizar captura con `tcpdump`



Veamos ahora cómo se interpretan los paquetes de una captura del protocolo TCP realizado con `tcpdump`. La línea general de un paquete TCP es como sigue:

```
src > dst: flags [dataseq ack window urgent options]
```

En principio `src`, `dst` y `flags` están siempre presentes, dependiendo el resto del tipo de conexión TCP de que se trate. El significado de dichos parámetros es:

- `src`: Dirección y puerto origen. En caso de no especificar el parámetro `-n` se intenta resover el nombre vía DNS y se busca el nombre del puerto en `/etc/services`.
- `dst`: Dirección y puerto destino. En caso de no especificar el parámetro `-n` se intenta resover el nombre vía DNS y se busca el nombre del puerto en `/etc/services`.
- `flags`: Indica los flags de la cabecera TCP. Puede ser un `.` cuyo significado es que no hay flags, o bien una combinación de `S` (SYN), `F` (FIN), `P` (PUSH), `W` (reducción de la ventana de congestión), `E` (ECN eco).
- `dataseq`: El número de secuencia del primer byte de datos en este segmento TCP. El formato es `primero:ultimo(n)`, que significa que desde `primero` a `ultimo` (sin incluir `ultimo`) hay un total de `n` bytes de datos.
- `ack`: El número de asentimiento. Indica el número siguiente de secuencia que se espera recibir.
- `win`: Tamaño de la ventana de recepción.
- `urgent`: Existen datos urgentes.
- `options`: Indica la existencia de opciones. En caso de que haya van entre <y >.

Veamos con datos reales cómo sería una captura sencilla realizada con `tcpdump`:

```
root@guadalinex:~# tcpdump -i eth0 -s 1500
tcpdump: listening on eth0
21:32:16.657523 172.26.0.1.1863 > 172.26.0.40.ssh: S 1473470138:1473470138(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
21:32:16.657762 172.26.0.40.ssh > 172.26.0.1.1863: S 2409899396:2409899396(0) ack 1473470139 win 5840 <mss 1460,nop,nop,sackOK> (DF)
21:32:16.665792 172.26.0.1.1863 > 172.26.0.40.ssh: . ack 1 win 17520 (DF)

3 packets received by filter
0 packets dropped by kernel
```

Esto simula el inicio de una conexión originada por la máquina `172.26.0.1` con destino a `172.26.0.40`, con el servicio `ssh`.

El significado de las líneas anteriores es:

1. Inicio de conexión de `172.26.0.1` por el puerto `1863` con `172.26.0.40` por el puerto `ssh`. El flag `SYN` está activado, el paquete tiene un número de secuencia `1473470138`, no contiene datos y la ventana de recepción es de `16384` *bytes*.

2. Envío de paquete `SYN` de `172.26.0.40` por el puerto `ssh` a `172.26.0.1` por el puerto `1863`. El número de secuencia es `2409899396`, ventana de `5840` y constituye el `ACK` del `SYN` anterior.

3. `ACK` del `SYN` mandado por `172.26.0.40`. No hay *flags*

En una red formada por concentradores (hubs), el tráfico se replica por todas las puertas (bocas). Si estamos en una red conmutada (formada por switches), los paquetes solamente salen por la boca en la que se encuentra la máquina de destino. Es un poco más segura, pues no es tan fácil capturar los paquetes, aunque herramientas hay para ello.

4.5.2. Arrancando ethereal

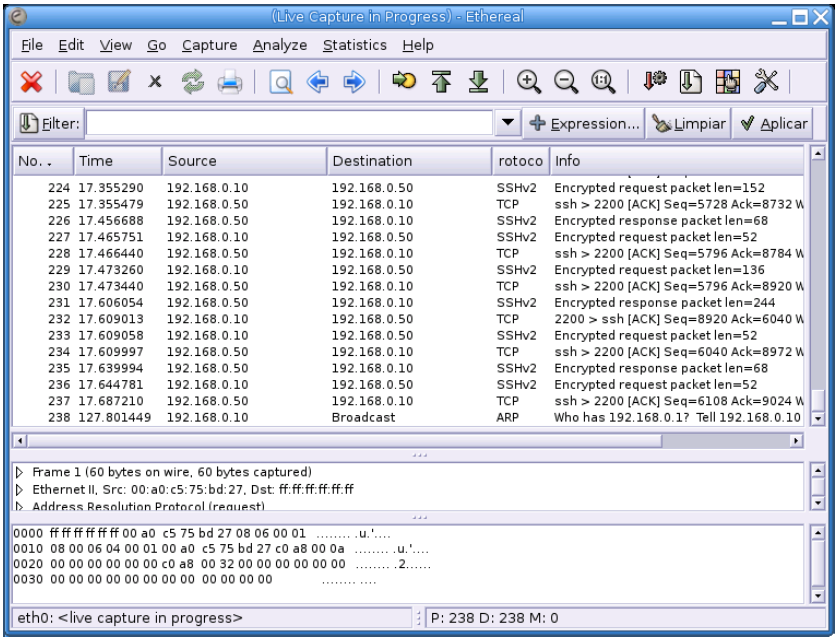
Ethereal está compuesto por tres ventanas o paneles:

1. El panel superior contiene el listado de paquetes. Muestra un resumen de cada paquete capturado. Pulsando sobre los paquetes que aparecen en este panel podemos controlar lo que se muestra en los otros paneles.

2. El panel central muestra una visión en árbol con más detalle del paquete seleccionado en el panel anterior.

3. El panel inferior muestra los datos del paquete. Muestra los datos correspondientes al paquete seleccionado en el panel superior y resalta el campo seleccionado en la visión en árbol.

Figura 4.2: Pantalla de inicio ethereal



Además de estos paneles, hay cuatro elementos en la parte superior/inferior de Ethereal que pueden sernos de utilidad:

Botón Filter. Pulsando este botón se muestra la ventana de filtrado de paquetes, donde pueden contruirse filtros tan complicados como queramos.

Cadena de filtrado. El campo de entrada de texto nos permite editar los filtros. Aquí se muestra también el filtro que se usa en la captura actual.

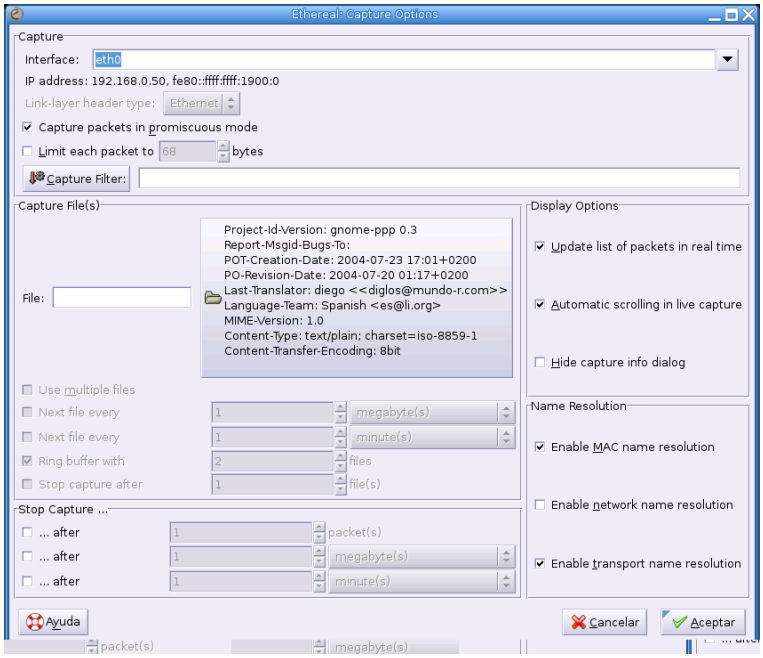
Reset. Limpia el filtro que se está utilizando en la captura actual.

Información general sobre la captura. Muestra información relativa al proceso que realiza en un momento dado Ethereal. Se encuentra en la parte inferior.

4.5.3. Capturando paquetes

Para comenzar una captura de los datos será necesario entrar en el menú **Capture→Start**. Aparecerá un nuevo cuadro de diálogo donde se configurarán algunos aspectos relativos a la captura.

Figura 4.3: Configuración de la captura



- **Interface**. Selecciona el interfaz de red por el que realizaremos la escucha. Sólo podemos realizar capturas por un interfaz de red al mismo tiempo.
- **Limit each packet to**. Indica el tamaño máximo en bytes que van a tener los paquetes que capturemos. En caso de no estar seleccionada la casilla se establece el límite de 65535 bytes por defecto.
- **Capture packets in promiscuous mode**. En caso de no seleccionar esta opción ethereal únicamente capturará los paquetes que circulen entre nuestro equipo y el resto de la red.
- **Filter**. Permite especificar un filtro de captura.
- **File**. Especifica el fichero que guardará la captura. En caso de que no indiquemos ninguno, la realizará en un fichero temporal.
- **Use ring buffer**. En caso de estar seleccionado, indica el número de ficheros que van a utilizarse para guardar las capturas.
- **Rotate capture file every**. Indica el tiempo, en segundos, que transcurre hasta que se realiza la captura en un nuevo fichero.
- **Update list of packets in real time**. Si se selecciona, mostrará las capturas en tiempo real.
- **Automatic scrolling in live capture**. Sólo puede activarse en caso de que la opción anterior lo esté, mostrando así los últimos paquetes que se han capturado realizando un desplazamiento automático de la pantalla con este objeto.
- **Stop capture after**. Establece un punto en el que la captura va a detenerse.
- **Enable resolution**. Indica que las direcciones MAC, las direcciones de red y los números de puerto de la capa de transporte se traduzcan a nombres.

4.5.4. Filtrado durante la captura

Dependiendo del tráfico del segmento de red donde nos encontremos podemos obtener capturas exageradamente grandes. Es recomendable el uso de filtros de forma que únicamente capturemos los datos que necesitemos (con destino/origen en un determinado servidor, los referidos a un protocolo en concreto, ...).

Ethereal usa el lenguaje de filtrado proporcionado con la librería `libpcap`. Pueden encontrarse más detalles al respecto en la página del manual de `tcpdump`.

El formato general que se utiliza para definir los filtros es el siguiente:

```
[not] primitiva [and|or [not] primitiva ...]
```

Entrando en más detalle, las primitivas que pueden utilizarse en la creación de filtros para la visualización de paquetes son:

[src|dst] host <host>

Permite filtrar por una dirección IP o nombre de host, indicando con src y dst si es dirección fuente o destino respectivamente[53]. En el caso que no se indique, se mostrarán los paquetes en los que aparezca la dirección indicada, ya sea fuente o destino.

ether [src|dst] host <host>

Permite filtrar por dirección ethernet.

gateway host <host>

Permite filtrar los paquetes que utilizan host como gateway. Es decir, donde la fuente o el destino ethernet es host pero ni la IP de la fuente ni del destino es host.

[src|dst] net <net>[mask <mask>]

Permite filtrar por número de red, pudiendo especificar la máscara de red.

[tcp|udp] [src|dst] port <port>

Permite filtrar por puertos, ya sean TCP o UDP.

less|greater <length>

Permite filtrar paquetes cuya longitud sea menor/mayor o igual que la longitud especificada respectivamente.

ip|ether proto <protocol>

Permite filtrar en el protocolo específico en la capa Ethernet o la capa IP.

ip|ether broadcast|multicast

Permite filtrar el tráfico broadcast o multicast.

<expr>relop <expr>

Permite crear expresiones de filtrado más complejas que seleccionan bytes o rangos de bytes en paquetes.

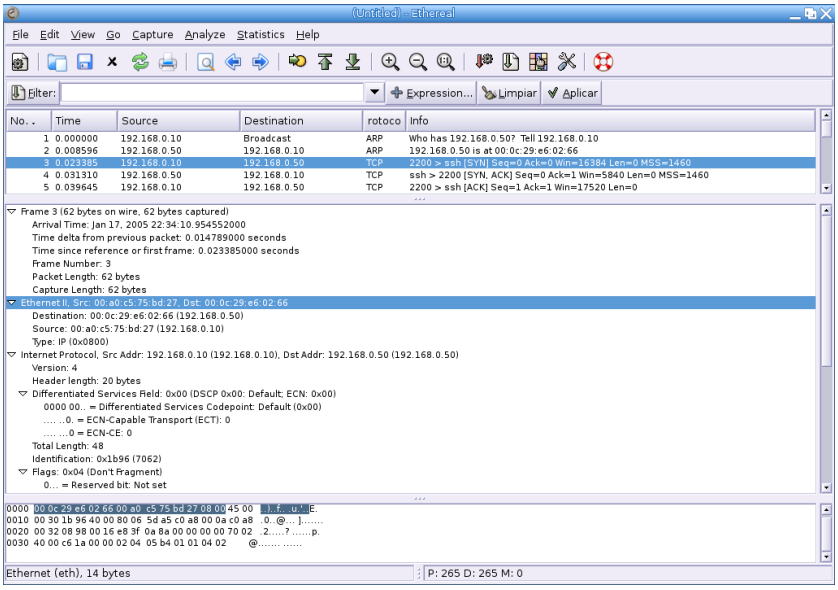
4.5.5. Visualización de los datos capturados

Una vez que hemos finalizado la captura de los paquetes, o estamos visualizando una captura almacenada previamente, tenemos la posibilidad de ver los paquetes con más detalle. Para ello, únicamente tenemos que pulsar sobre un paquete en el panel superior para obtener los detalles del mismo en el panel central.

Podemos expandir cualquier parte del árbol que constituye el paquete pulsando sobre + en la parte izquierda.

[53] Los modificadores src y dst tienen el mismo efecto en las demás combinaciones de filtros.

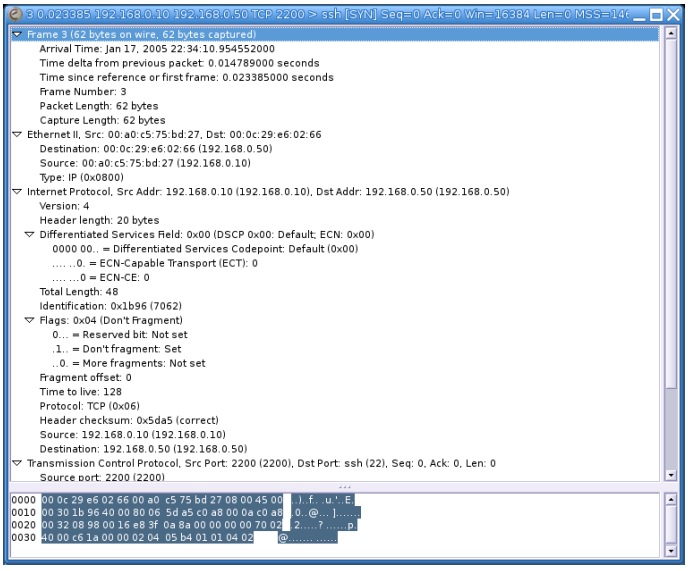
Figura 4.4: Detalles de un paquete capturado



También podemos seleccionar y visualizar paquetes con detalle mientras se está realizando la captura, si hemos seleccionado la opción "**Update list of packets in real time**".

Otra posibilidad para la visualización de paquetes es utilizando una ventana auxiliar, para ello sólo tenemos que pulsar con el botón derecho sobre el paquete previamente seleccionado y aparecerá una nueva ventana con los detalles.

Figura 4.5: Detalles de un paquete capturado en ventana independiente



Hasta aquí la herramienta nos presenta información muy útil del tráfico de la red, pero aún podemos sacarle algo más de jugo. Tal como acabamos de ver, al pulsar sobre cualquier paquete que esté seleccionado con el botón derecho aparece un nuevo menú:

- **Follow TCP Stream**. Permite ver todos los datos que una cadena de paquetes produce entre dos nodos. En una ventana separada muestra todos los segmentos TCP capturados que están en la misma conexión TCP como un solo paquete. Los datos de la secuencia se ordenan y se eliminan los paquetes duplicados, mostrándose en formato ASCII.
- **Decode as**. Permite al usuario forzar a ethereal a decodificar ciertos paquetes como un protocolo en particular.
- **Display Filters**. Permite especificar y manejar filtros.
- **Colorize Display**. Permite colorear los paquetes en el panel de la lista de paquetes de acuerdo a los filtros que elijamos.
- **Print**. Permite imprimir todos los paquetes de una captura.
- **Print Packet**. Permite imprimir el paquete seleccionado.

Además del filtrado de paquetes durante la captura, podemos realizar un filtrado posterior en la visualización de los paquetes capturados o que se están capturando. El lenguaje de filtrado de paquetes en la visualización de los mismos es distinto del utilizado en la captura.

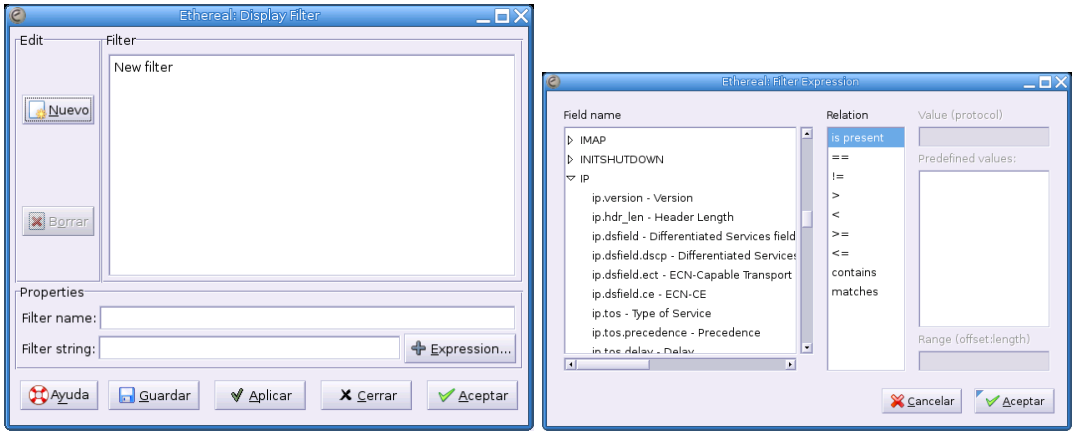
El filtrado de paquetes mientras los visualizamos nos permite concentrarnos en los paquetes en los que estamos interesados. Nos permite seleccionar paquetes según distintos criterios.

- Protocolo
- Presencia de un campo
- Valor de un campo
- Comparación entre campos

4.5.6. Filtrado durante la visualización

Los filtros utilizados durante la visualización de los paquetes tienen más potencia que los utilizados durante la captura. Esto hace que su uso sea algo más complicado. Sin embargo, una vez que nos hayamos peleado con estos filtros, veremos que no es algo difícil y que, normalmente, utilizamos un pequeño subconjunto de etiquetas y expresiones, lo que simplifica su uso. Para acostumbrarnos a su uso es muy útil la ventana **"Filter Expresion"**.

Figura 4.6: Creación de expresiones de filtrado



Para acceder a la ventana de creación de filtros basta pulsar el botón **"Filter"** que aparece en la parte superior.

La ventana de creación de filtros tiene tres listas de campos. La primera de ellas nos muestra los protocolos disponibles y la segunda la relación existente con el valor del tercer campo:

- **Field Name**. Selecciona un campo de protocolo del árbol existente. Cada protocolo tiene una serie de campos por los que se puede filtrar.
- **Relation**. Permite seleccionar una relación de la lista de disponibles. Todas son relaciones binarias que requieren datos adicionales, excepto "is present" que es unaria y que devolverá verdadero/falso dependiendo de si se cumple o no.
- **Value**. En este campo se introduce el valor con el que se quiere comparar. Como ayuda, en el título del campo se indica el tipo de dato que se espera[54].
- **Predefined values**. Algunos protocolos presentarán la opción de elegir entre una serie de valores predeterminados.

Cuando se selecciona un campo de protocolo y una relación binaria, se espera que se proporcione un valor adicional con el que comparar.

Un ejemplo de filtro sería:

```
(ip.addr eq 192.168.0.10 and ip.addr eq 192.168.0.50) and (tcp.port eq 2552 and tcp.port eq 23)
```

Con este filtro se visualizarán/capturarán los paquetes que cumplan que la dirección IP origen o destino sea `192.168.0.10` y `192.168.0.50` y que utilicen los puertos TCP `23` y `2552`.

4.5.7. Capturando una sesión telnet

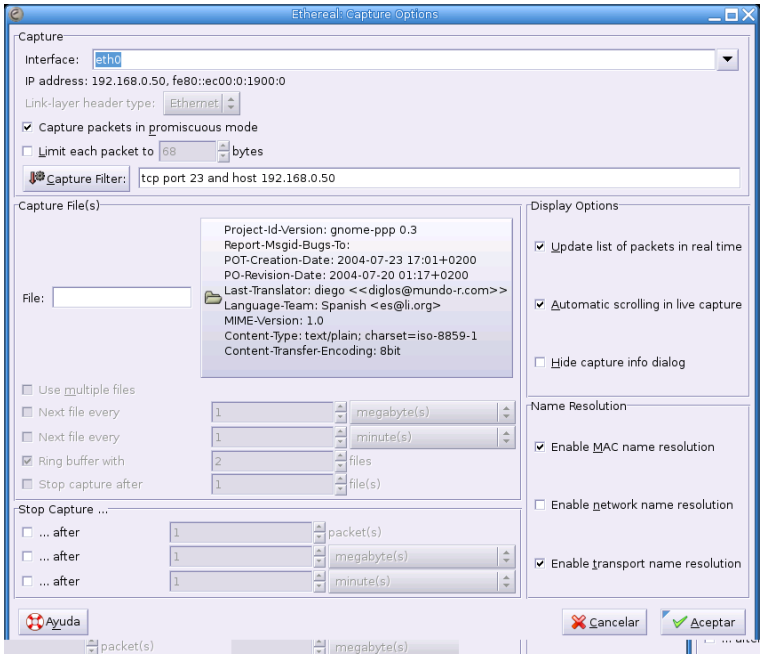
Veamos ahora el resultado de una captura de una sesión telnet que realiza el usuario `legolas`. Mediante la opción "**Follow TCP Streams**" veremos los datos de una sesión de telnet tal y como los vería la capa de aplicación.

Al capturar paquetes en modo promiscuo estaremos capturando todo el tráfico que hay en la red, por lo que utilizaremos los filtros para que capture únicamente los paquetes que circulan hacia/desde el ordenador que tiene el servicio telnet activado con dirección IP `172.26.0.40`. Del mismo modo, al querer capturar únicamente la sesión telnet, indicaremos que el protocolo que queremos capturar es el que utiliza el puerto 23.

```
tcp port 23 and host 172.26.0.40
```

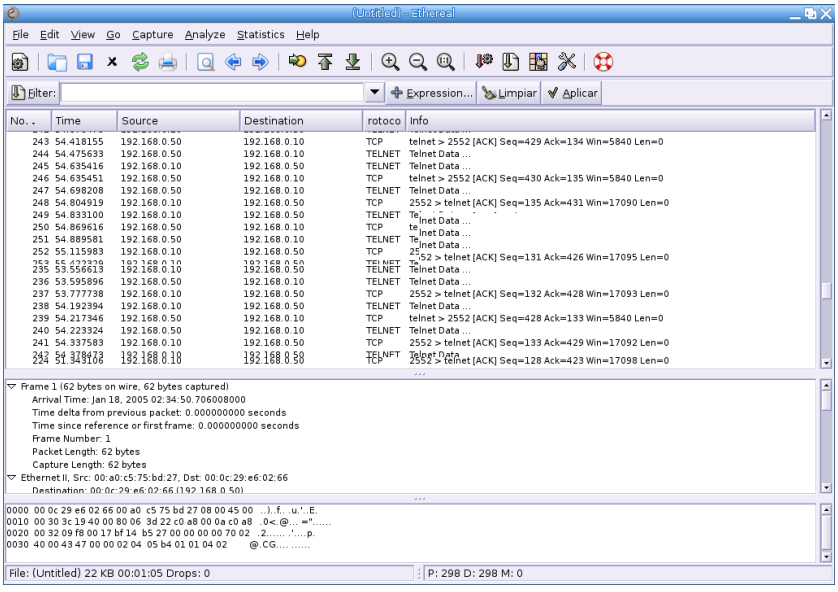
[54] En el caso de seleccionar `ip.addr` se esperará un valor que sea una dirección IPv4.

Figura 4.7: Filtro de captura de sesión de telnet



Después de capturar unos segundos, tendremos una captura de la que podemos sacar completa información de los mensajes que se han transmitido entre el ordenador cliente y el servidor.

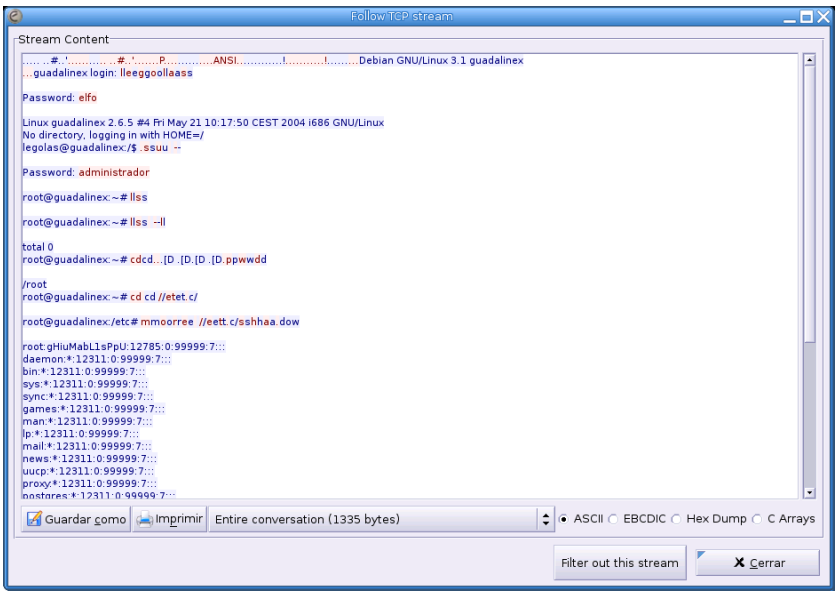
Figura 4.8: Captura de sesión de telnet



Aquí tenemos todos los datos en crudo de la captura, mostrando los paquetes TCP tal y como se transmiten por la red. Sin embargo, es posible hacer que estos datos sean más comprensibles y nos proporcionen datos más jugosos. Para ello utilizaremos el menú **Tools** → **Follow TCP**

Stream.

Figura 4.9: Seguir la trama TCP de la sesión telnet



Como se observa en la imagen anterior, podemos ver exactamente lo que ha estado haciendo el usuario. Hemos obtenido datos sensibles y que vulneran la seguridad del sistema tales como la clave del usuario y la del `root`. Esto último es especialmente grave, ya que nos daría control total sobre el sistema.

Moraleja: no debemos utilizar protocolos no cifrados.

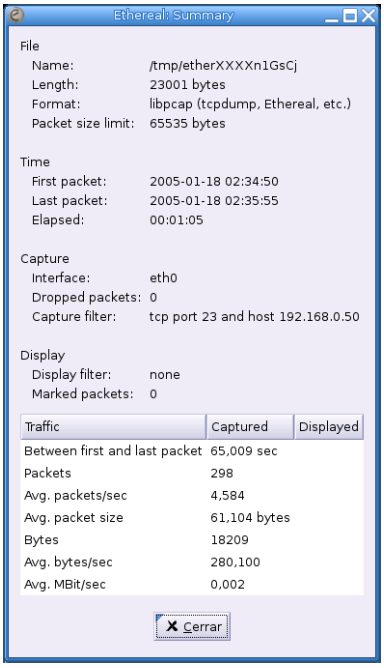
4.5.8. Estadísticas acerca de la captura

Hasta el momento, lo contado sobre Ethereal le proporciona una gran potencia a priori, haciendo su uso muy recomendable en cualquier análisis de los datos que circulan por la red. Sin embargo aún podemos obtener más datos.

Siempre puede ser interesante obtener los datos globales acerca del número de paquetes, tamaño de los mismos, duración de la captura, ... Todos estos datos y algún otro pueden obtenerse con las opciones que presenta el menú **Statistics**.

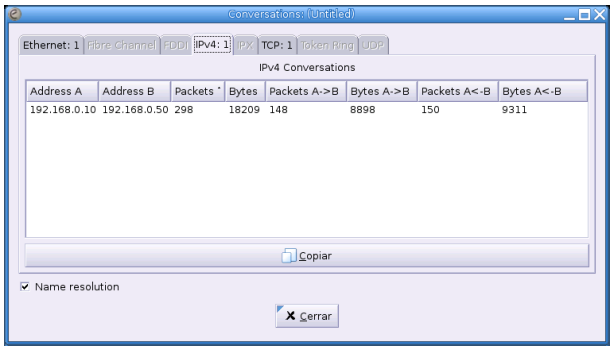
Con la opción **Summary** obtendremos un resumen de la captura realizada.

Figura 4.10: Sumario de la captura



Otra opción que nos puede proporcionar datos interesantes dentro de este menú es **Conversations**. Un ejemplo para entender el concepto de conversación sería el tráfico existente entre dos direcciones IP, este tráfico constituiría una conversación IP.

Figura 4.11: Conversación IPv4



Dentro del menú **Statistics** existen aún más opciones, pero nosotros nos quedaremos aquí, aunque os animamos a investigar sobre el resto de opciones que os proporcionarán datos muy interesantes sobre la captura.

4.6. Monitorizacion de red con EtherApe

El programa `etherape` es una herramienta de monitorizacion gráfica de redes. La principal característica de este software es que es muy intuitivo y sencillo de utilizar. A la hora de monitorizar una red, es un complemento más a otro tipo de programas como el anteriormente visto, `ethereal`.

No tiene la finalidad de un análisis exhaustivo del tráfico de la red como otras herramientas, pero sí ofrece un enorme potencial a la hora de monitorizar en tiempo real el tipo de tráfico existente. Esto último nos va a permitir detectar problemas puntuales de una forma muy rápida y sencilla. Otra funcionalidad importante es obtener estadísticas del tipo de tráfico que tenemos en nuestra red para posibles mejoras de nuestra infraestructura.

Es muy útil a la hora de detectar una máquina que sobrecarga la red, una conexión no deseada o errores de determinados servicios. Y todo ello, de una manera muy sencilla.

4.6.1. Instalación

La instalación de `Etherape` se puede realizar con la propia distribución. Si ya tenemos la distribución sin `etherape` su instalación es similar a la de cualquier otro paquete.

Podemos utilizar la herramienta `apt`:

```
#apt-get install etherape
```

o, en Fedora, el correspondiente paquete rpm:

```
#rpm -Uhv etherape-0.9.1-1.1.fc1.rf.i386.rpm
```

Con esta última opción tendremos que tener en cuenta las dependencias con los paquetes `libpcap`, `gtk+`, `libglade` y la interfaz de escritorio gnome.

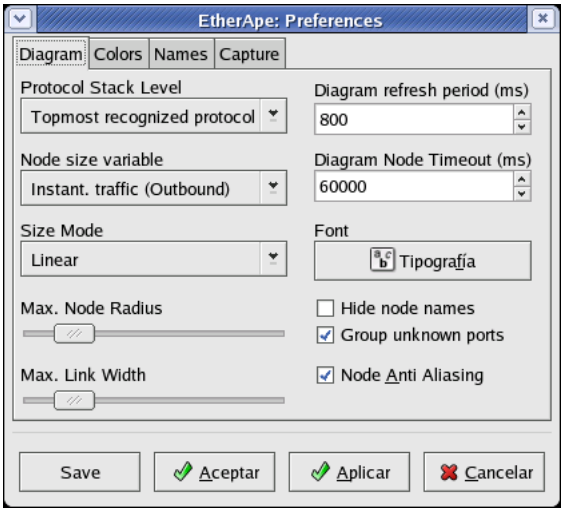
Una vez instalado, nos aparecerá en la opción de Internet del menú principal, listo para utilizarse.

4.6.2. Configuración

Una vez lanzado, ya está listo para monitorizar el tráfico. Según el tráfico que queramos monitorizar, debemos situar la máquina utilizada en un determinado segmento de red.

`Etherape` nos permite en la opción de **preferencias**, variar los parámetros de configuración.

Figura 4.12: Cuadro de preferencias de Etherape



Entre los parámetros que podemos modificar se encuentran:

- Nivel de la pila de protocolos que queremos monitorizar

- Periodo de actualización
- Tráfico y Nodos que se mostrarán en el diagrama y su tamaño
- Aspecto
- Protocolos específicos
- Interfaces y el modo de visualización.

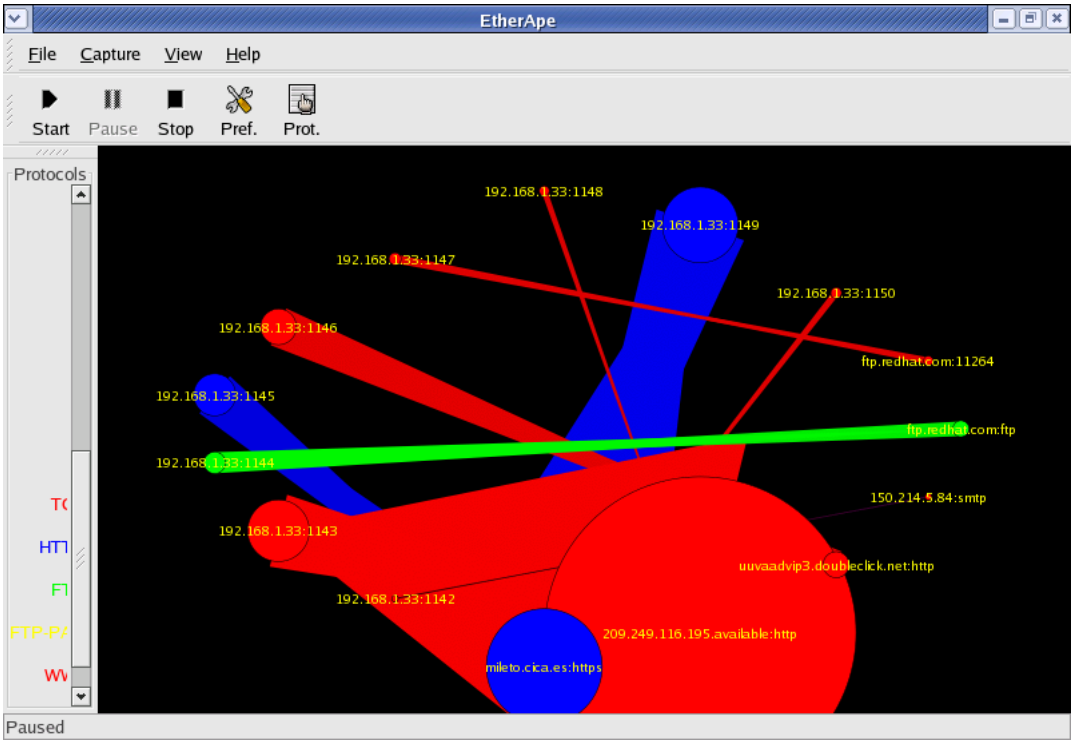
4.6.3. Funcionamiento

El funcionamiento es intuitivo y sencillo. Una vez que comenzamos a monitorizar nos aparecerá cada conexión desde el nodo origen al nodo destino, con un color diferente según cada protocolo y, en función del tráfico, un círculo en torno al nodo destino.

Con los botones de pausa y parada podemos fijar la imagen o resetearla respectivamente. Si dejamos funcionando `Etherape` en un segmento de red con gran cantidad de tráfico, es posible que tengamos que parar de forma periódica o afinar los parámetros para poder ver el tráfico que nos interese.

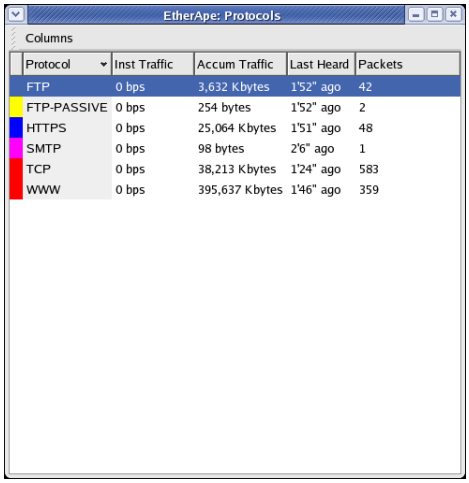
De esta forma, por ejemplo, una máquina que está provocando gran tráfico en la red se podrá detectar de inmediato, incluso saber el destino y el tipo de tráfico (por ejemplo, descarga por ftp a un servidor externo, o de emule).

Figura 4.13: Captura de Etherape



En la opción de **Protocolos**, se nos muestran los diferentes tipos de tráfico junto con las estadísticas totales e instantáneas de cada uno de ellos.

Figura 4.14: Estadísticas de protocolos de Etherape



Capítulo 5

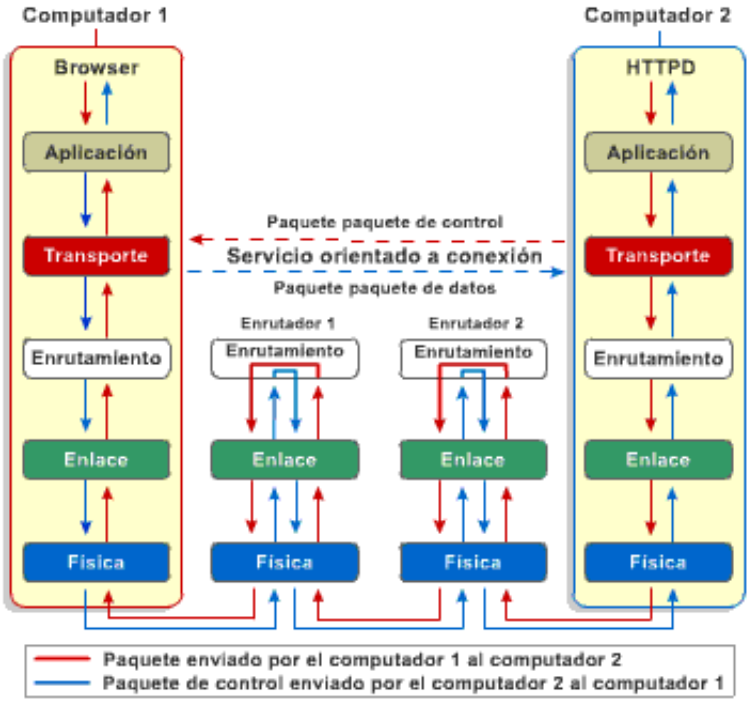
Conectando al mundo exterior

> Siempre me ha fascinado Internet, incluso antes de que existiera.
>
> (*Servidor Apache*, Rich Bowen & Ken Coar, Prentice Hall)

Hasta ahora, hemos visto cómo nuestro sistema se sitúa en nuestra red, nuestro pequeño mundo, sin visibilidad con el exterior. Veamos cómo nos podemos conectar con otras redes y el tan ansiado mundo exterior de Internet.

5.1. Routing o encaminamiento IP

Con el término *Routing* (encaminamiento) se designa al proceso de escoger el camino por el cual van a ser enviados los paquetes IP. El routing sucede cuando el host destino no está en nuestra red IP. Un router es una máquina o un dispositivo que reenvía los paquetes desde una red lógica a otra. A los routers también se les denomina *gateways*[1].



[1]Estrictamente, cuando el sistema que une las diferentes redes trabaja en el nivel de red, se denomina router y cuando trabaja en un nivel superior, se denomina gateway o pasarela.

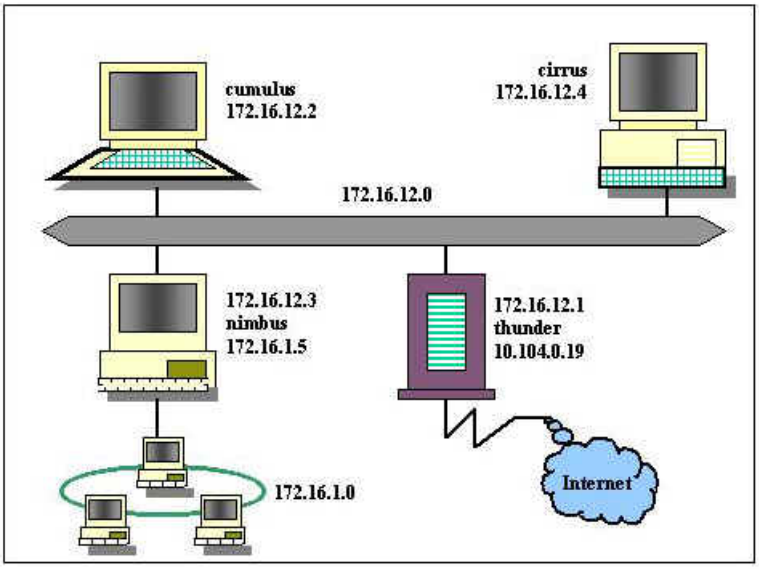
Veamos el proceso que se sigue cuando hay que enviar un paquete de datos en una red TCP/IP.

Para ello, vamos a utilizar la red de ejemplo de la figura siguiente. Veamos los componentes y redes presentes. Hay dos redes internas (que pueden ser la de administración y la de alumnos en un centro educativo): la red 172.16.12.0/24 y la red 172.16.1.0/24.

En la red 172.16.1.0/24[2] hay varias máquinas. Una de ellas es `nimbus`, con dirección IP 172.16.1.5. Otra máquina podría ser la 172.16.1.4, que llamaremos `marte`.

En la red 172.16.12.0/24 se encuentran `cumulus` (172.16.12.2), `cirrus` (172.16.12.4), `nimbus` (172.16.12.3) y `thunder` (172.16.12.1).

Observamos que `nimbus` se encuentra en las dos redes. Dispone de dos interfaces de red y uno de ellos se encuentra en una red y el otro en otra red. También `thunder` pertenece a dos redes, a la 172.16.12.0/24 y posee otro interfaz con la dirección 10.104.0.19, que provee el acceso a Internet.



Veamos los distintos casos que pueden ocurrir.

1. Si la dirección IP de destino está en nuestra misma red lógica, el envío se realiza directamente[3]. Por ejemplo, una comunicación entre `cumulus` (172.16.12.2) y `cirrus` (172.16.12.4).

2. Si la IP de destino se encuentra en otra red, debe tomarse la decisión de por dónde hay que enviarlo. Esta decisión deben tomarla todos los hosts, a los que llega el paquete IP, bien sea nuestro propio host o cualquier router por los que el paquete atraviese. Para tomar la decisión de enrutamiento, la capa IP consulta una tabla de rutas[4] que está almacenada en memoria.

   Mostramos la tabla de rutas de la máquina `nimbus` y veamos qué significa.

```
Red Destino      Máscara        Gateway       Interfaz
172.16.12.0    255.255.255.0   0.0.0.0         eth0
172.16.1.0     255.255.255.0   0.0.0.0         eth1
0.0.0.0        0.0.0.0         172.16.12.1     eth0
```

   Para llegar a la red 172.16.12.0/24[5], lo puede hacer a través de su primer interfaz, que se

[2] Aparece dibujada en forma circular, representando que es una red con topología en anillo.
[3] Solamente hay que encontrar mediante ARP la dirección física.
[4] Una tabla de rutas contiene entradas que relacionan la dirección IP buscada y la interface a utilizar para llegar a ella.
[5] Es la red en la que se encuentra con dirección 172.16.12.3

llama eth0[6]. Para llegar a la red 172.16.1.0/24[7], puede hacerlo directamente a través de su interfaz eth1. Para el resto de redes no especificadas explícitamente, lo que se llama *ruta por defecto*, manda los paquetes al gateway 172.16.12.1 y que éste se encargue de tomar las siguientes decisiones de rutas.

Como ejemplo, veamos lo que sucedería en una comunicación entre `cirrus` y `marte`.

La tabla de rutas de la máquina `cirrus` es la siguiente:

Red Destino	Máscara	Gateway	Interfaz
172.16.12.0	255.255.255.0	0.0.0.0	eth0
172.16.1.0	255.255.255.0	172.16.12.3	eth0
0.0.0.0	0.0.0.0	172.16.12.1	eth0

Para llegar a la red 172.16.12.0/255.255.255.0, en la que se encuentra, lo puede hacer directamente a través de su interfaz eth0[8]. Para llegar a la red 172.16.1.0/24, debe hacerlo a través de la dirección 172.16.12.3, por el interfaz eth0, como recoge la segunda línea de la tabla de rutas. Para el resto de rutas, lo hará a través de la dirección 172.16.12.1, que es su ruta por defecto.

Recordemos ahora que nuestro objetivo es la comunicación entre `cirrus` y `marte`. `Cirrus` (172.16.12.4) tiene que comunicarse con `marte` (172.16.1.4). Como no está en su red, mira la tabla de rutas. Para la red en la que se encuentra `marte` (172.16.1.0/24), comprueba que tiene un camino para llegar, que es enviarlo a `nimbus` (172.16.12.3), al que puede llegar directamente porque está en su misma red.

El paquete llega a `nimbus`, al cual se le presenta nuevamente el problema del encaminamiento. En este caso es más fácil, porque `marte` (172.16.1.4) está en una red a la que `nimbus` pertenece (en este caso, 172.16.1.5) y a la que accede por su interfaz eth1. Así el paquete llega a `marte`. Hemos de notar que deben existir las rutas en el camino inverso para que los paquetes de vuelta[9] entre `marte` y `cirrus` puedan llegar.

3. Si no hay una ruta explícita, IP utiliza el gateway por defecto para enviar el paquete al router. Por ejemplo, si `cirrus` (172.16.12.4) quiere conectarse con el servidor web de `mileto.cica.es` (150.214.5.11), mira sus tablas de rutas y no ve una conexión directa con la red 150.214.5.0 a la que pertenece `mileto.cica.es`. Entonces, lo manda por la ruta por defecto, al gateway `thunder` (172.16.12.1), que lo encaminará hacia Internet. En el gateway, otra vez es consultada su tabla de rutas, para seguir buscando un camino al host remoto. Si no existe un camino explícito, el router reenviará otra vez el paquete a su propio gateway por defecto para continuar la cadena y que sea este siguiente router el encargado de repetir el ciclo.

   Cuando vamos pasando por un router y el paquete se envía al siguiente router, esto se denomina un "salto" y se descuenta una unidad del valor del campo TTL[10] del paquete IP. Finalmente, el proceso acaba si el paquete es entregado en el host destino. Si alguna ruta no se encuentra[11], se envía un mensaje de error, mediante el protocolo ICMP al host origen, diciendo que el destino ha sido inalcanzable.

[6] Es la primera (se empieza por 0) interfaz con protocolo ethernet (eth).
[7] En la que tiene dirección 172.16.1.5
[8] De hecho, sólo tiene una tarjeta de red y es el único sitio por el que puede salir.
[9] Lo normal es que cualquier comunicación sea en ambos sentidos con paquetes de envío y de confirmación por parte del destinatario. Si las rutas en sentido inverso no existen o no están bien configuradas, fallará la comunicación.
[10] Tiempo de Vida (*Time To Live*).
[11] Eso ocurre cuando el valor de TTL llega a cero. Es un mecanismo para evitar que los paquetes vaguen eternamente como almas en pena de router en router sin llegar a ningún destino.

5.1.1. Estático y dinámico

El mecanismo de actualización de las tablas de rutas puede ser estático o dinámico. Esto normalmente sólo afecta a los routers, ya que los hosts solamente suelen tener en sus tablas de rutas la red a la que pertenecen y el gateway por defecto.

Los routers estáticos necesitan que las tablas de rutas sean construidas y actualizadas manualmente. Si una ruta cambia, los routers estáticos no se enteran de este cambio automáticamente. En el ejemplo anterior, las tablas de rutas son pequeñas y podríamos haberlas creado de forma estática.

Sin embargo, cuando el número de redes a las que dan acceso los routers es elevado, actualizar las tablas de forma estática sería una labor muy compleja. Para ello, existen los protocolos dinámicos de routing, que hacen que los routers puedan anunciarse los cambios en sus tablas de rutas de una forma automática.

Por ejemplo, son protocolos de routing el *Routing Information Protocol* (RIP) o el *Open Shortest Parh First* (OSPF). Los protocolos de routing periódicamente intercambian rutas a sus redes conocidas a lo largo de los routers dinámicos. Si una ruta cambia, todos los routers dinámicos son informados de dicho cambio.

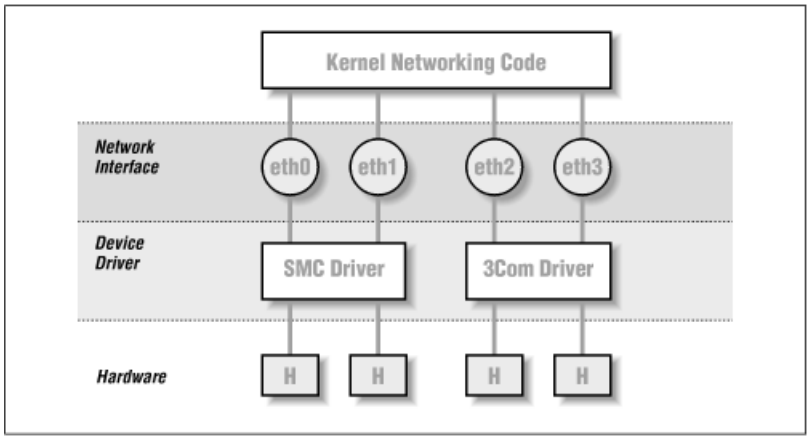
5.2. Vale, pero yo quería un curso de Linux.

Pasemos a ver cómo funciona todo esto de forma práctica en un sistema Linux. Como probablemente ya sabéis, todo en Linux es un fichero, o asimilable a un fichero. Por lo tanto, los interfaces de red, no serán una excepción. La siguiente figura, nos muestra, los elementos tanto hardware como software necesarios para que nuestro sistema linux acceda a la red:

El hardware: puede ser un módem, una tarjeta de red ethernet, tarjeta RDSI...

Los drivers: que son la parte del kernel dependiente del hardware específico. Por ejemplo, para un modelo de tarjeta SMC será diferente al que requiere una tarjeta 3Com.

El interfaz de red dependerá del tipo de red a la que nos conectemos: para una red Ethernet será *eth0*[12], para un módem será *ppp0*, o para el caso del interfaz de loopback, *lo*.



Normalmente, no hay que preocuparse por asuntos de hardware como las direcciones base de Entrada/Salida[13] o las IRQ[14], porque al arrancar el kernel, intenta detectar la tarjeta y los valores que utiliza. Esto se denomina prueba automática[15], que significa que el kernel lee en varias

[12] Si tenemos otra tarjeta de red, sería eth1 y así sucesivamente

[13] En las que el dispositivo físico escribe y lee

[14] Interrupciones que utiliza para comunicarse con el resto del sistema físico, pedir permiso para utilizar recursos comunes.

[15] *Autoprobe*, en inglés

posiciones de memoria y compara los datos que ha encontrado con los que espera de una tarjeta de red en concreto. Solamente en determinados casos[16] habría que especificar valores al kernel para que las encuentre.

En el arranque del equipo[17], podemos observar si ha sido detectado. Por ejemplo:

```
...
Intel(R) PRO/100 Network Driver - version 2.3.18-k1
Copyright (c) 2003 Intel Corporation
PCI: Guessed IRQ 9 for device 00:06.0
PCI: Sharing IRQ 9 with 00:06.1
divert: allocating divert_blk for eth0
e100: selftest OK.
e100: eth0: Intel(R) PRO/100 Network Connection
Hardware receive checksums enabled
cpu cycle saver enabled
...
```

5.2.1. Activando interfaces

Cuando vayamos a conectar nuestra máquina a una red, necesitamos poder ubicarla dentro de la red sin que cause conflictos y se integre con el resto de sistemas que se encuentran en la red. Debemos solicitar a los administradores de la red[18] una serie de valores. Estos valores son: Dirección IP de nuestro sistema, máscara de la red, dirección IP del router[19] de salida y dirección de los servidores DNS. En este ejemplo, utilizaremos 172.26.0.2/255.255.255.0, con el gateway 172.26.0.1 y servidores de DNS 195.235.113.3, 80.58.0.33 y 150.214.4.34.

Hay casos en los que esto, si cabe, puede ser más fácil. Si en nuestra red existe un servidor de DHCP[20], solamente tendremos que decirle a nuestra máquina que coja la configuración de red de forma dinámica mediante este protocolo y estos valores se asignarán automáticamente.

Hay una serie de comandos que se usan con objeto de configurar las interfaces de red e inicializar la tabla de encaminamiento. Esas tareas son ejecutadas generalmente por el script de inicialización de red cada vez que el sistema es arrancado. Las herramientas básicas son `ifconfig`[21], y `route`.

- El comando `ifconfig` se usa para dar acceso al kernel a una interfaz física. Esto incluye la asignación de una dirección IP y la activación de la interfaz. Por activación nos referimos a permitir que el kernel pueda enviar y recibir datagramas IP a través de esa interfaz.
- El comando `route` permite añadir o quitar rutas de la tabla de encaminamiento del kernel.

Con el propósito de que sirva de ejemplo, nos referiremos a la interfaz de *loopback* y a una tarjeta de red ethernet.

Comando ifconfig

El interfaz de bucle local (loopback) es usado para realizar tests, y para simular aplicaciones en red, aun cuando no poseamos hardware de red. Funciona como un circuito cerrado que devuelve cualquier datagrama recibido a la capa de red del host. Siempre hay un dispositivo loopback presente en el kernel, denominado *lo*.

Si nuestra máquina no va a estar conectada a ninguna red, sólo se necesita la dirección de la interfaz de bucle local, que es siempre 127.0.0.1.

[16] Cada vez menos habituales, porque el soporte en el kernel para cada tipo de hardware suele venir ya incorporado, bien en el propio kernel o en módulos que se cargan cuando se necesitan.
[17] También en el fichero `/var/log/messages` o con el comando `dmesg`
[18] Eh, pero si ese soy yo.
[19] O Gateway
[20] Para el mismo propósito serviría bootp, aunque menos utilizado y más antiguo.
[21] Que viene de i*nterface configuration*

Es la primera interfaz en ser activada[22] y el comando que se utiliza es:

```
# ifconfig lo 127.0.0.1
```

que significa, activar la interfaz *lo* con la dirección ip 127.0.0.1

Para ver la configuración de una interfaz, usamos `ifconfig`, pasándole como argumento el nombre de la interfaz:

```
[root@linux entrega04-1]# ifconfig lo
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:2185 errors:0 dropped:0 overruns:0 frame:0
TX packets:2185 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:2125782 (2.0 Mb) TX bytes:2125782 (2.0 Mb)
```

Comentemos la salida obtenida:

- El interfaz *lo* tiene la dirección 127.0.0.1, con máscara de red 255.0.0.0, que se ha asignado automáticamente al ser una dirección de Clase A. Vemos que el estado es UP, es decir, activa.
- El máximo tamaño de unidad de transferencia será de 16.436 bits, quiere decir que si el protocolo superior necesita transmitir paquetes de tamaños superiores, deberá fragmentarlos en trozos más pequeños.
- La métrica es una indicación de la distancia necesaria para llegar a dicha red, como nuestra máquina está directamente en ella es 1.
- Nos dice también el número de paquetes transmitidos (TX) y recibidos (RX) desde que se activó la interfaz, así como el número de errores y colisiones.

Lo siguiente es comprobar que todo funciona como es debido, por ejemplo usando el comando `ping`. Esta orden se usa para verificar que una dirección dada es accesible y para medir el retraso entre el envío de un datagrama y su recepción de vuelta. Este tiempo es conocido como tiempo de ida y vuelta[23].

```
# ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=255 time=0.4 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=255 time=0.4 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=255 time=0.4 ms ^C
--- localhost ping statistics --- 3 packets transmitted, 3 packets received, 0%
packet loss round-trip min/avg/max = 0.4/0.4/0.4 ms
```

Cuando se ejecuta `ping` según se muestra aquí, la emisión de paquetes continúa a menos que sea interrumpida por el usuario. Cuando se pulsa **Ctrl-C**, interrumpimos el comando.

Este ejemplo muestra que los paquetes dirigidos a la máquina 127.0.0.1 están siendo entregados correctamente y la respuesta a `ping` es recibida de forma casi instantánea (0.4 milisegundos). Esto significa que ha establecido con éxito su primera interfaz de red.

La configuración de una interfaz Ethernet es similar a la de la interfaz de bucle local. En nuestro ejemplo:

```
# ifconfig eth0 172.26.0.2 netmask 255.255.255.0
```

Esto asigna a la interfaz *eth0* la dirección IP 172.26.0.2 con la máscara de red 255.255.255.0:

```
[root@linux entrega04-1]# ifconfig eth0
eth0 Link encap:Ethernet HWaddr 00:D0:59:0F:05:1C
```

[22] Lo normal es que ya se encuentre activada en nuestro sistema.

[23] ¿Por qué será que me suena al ping-pong?

```
inet addr:172.26.0.2 Bcast:172.26.0.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:127 errors:0 dropped:0 overruns:0 frame:0
TX packets:138 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:99853 (97.5 Kb) TX bytes:20272 (19.7 Kb)
Interrupt:9 Base address:0x9000 Memory:81200000-81200038
```

En esta salida, se nos muestra la dirección MAC correspondiente a la tarjeta `Ethernet HWaddr 00:D0:59:0F:05:1C`. Se ha fijado la dirección de broadcast automáticamente (que se puede obtener sabiendo la dirección IP y la máscara de red). Se fija la unidad de transferencia de mensajes (MTU) a un máximo de 1.500 bytes. Igualmente, podemos hacer un ping para comprobar que se encuentra operativa.

5.2.2. Estableciendo rutas

Debemos indicar en la tabla de encaminamiento qué redes son accesibles mediante los interfaces lo y eth0. Lo hacemos con los comandos:

```
# route add -net 127.0.0.0 netmask 255..0.0.0 lo
# route add -net 172.16.0.0 netmask 255.255.255.0 eth0
```

Estamos indicando que en nuestra máquina, a la red 172.26.0.0/24[24], se llega a través de la interfaz eth0. Y a la red 127.0.0.0 llegamos a través del interfaz lo. La opción `add` del comando `route` añade rutas, con la opción `del`, las eliminamos y con la opción `-n` nos las mostraría.

Veámoslo con el siguiente comando:

```
[root@linux entrega05-1]# route -n
Kernel IP routing table
Destination     Gateway     Genmask         Flags Metric Ref    Use Iface
172.26.0.0      0.0.0.0     255.255.255.0   U     0      0        0 eth0
127.0.0.0       0.0.0.0     255.0.0.0       U     0      0        0 lo
```

El comando route nos dice que la red 172.26.0.0/255.255.255.0 está activa (UP) y llegamos a ella a través de la interfaz eth0. Igualmente sería para la red de loopback.

Hasta aquí nos hemos ubicado en la red local y podemos comunicarnos con otras máquinas de la red. Nos falta indicar un camino de salida para alcanzar otras redes. Esto se realiza indicando un gateway.

```
#route add default gw 172.26.0.1
```

Veamos cómo se ha modificado la tabla de rutas.

```
[root@linux entrega05-1]# route -n
Kernel IP routing table
Destination     Gateway     Genmask         Flags Metric Ref    Use Iface
172.26.0.0      0.0.0.0     255.255.255.0   U     0      0        0 eth0
127.0.0.0       0.0.0.0     255.0.0.0       U     0      0        0 lo
0.0.0.0         172.26.0.1  0.0.0.0         UG    0      0        0 eth0
```

Se ha añadido una ruta nueva a la red (0.0.0.0), que significa "cualquier red no especificada anteriormente". A ella se llega por medio del gateway 172.26.0.1, que se encuentra accesible por la interfaz eth0. O sea, cualquier paquete que no vaya dirigido a la red 172.26.0.0/24 ni a la red 127.0.0.0/8, se mandará a la dirección 172.26.0.1 para que "se ocupe de ello", que esa se supone que es la función de un router. Es una bonita forma de liberarnos del trabajo[25].

[24] El 24 es más cómodo de teclear que 255.255.255.0

[25] Si esto pudiera aplicarse en más casos ...

5.2.3. Resolución de nombres.

Hasta ahora, de forma premeditada, hemos obviado las direcciones simbólicas (de la forma `thales.cica.es`) y hemos trabajado exclusivamente con direcciones numéricas (172.26.0.2). Veamos cómo funciona en un host la resolución de direcciones simbólicas (nombres) a numéricas.

En una red pequeña, podemos mantener tablas de asignación de nombres a direcciones IP numéricas. Esta información se mantiene en un fichero llamado `/etc/hosts`. Cuando se añaden o se eliminan puestos, o se reasignan direcciones, lo que se hace es actualizar el fichero `/etc/hosts` en todos los puestos. Obviamente, esto solamente funciona en redes muy pequeñas.

```
$more /etc/hosts
127.0.0.1 localhost.localdomain localhost
172.26.0.2 linux linux.midominio.org
```

Para resolver este problema, hacemos uso de los servidores de nombre DNS, a los cuales les preguntamos por una dirección simbólica y nos devuelven la dirección numérica. Esta es una de las labores de los servidores de nombres, que ampliaremos en la segunda entrega, donde hablaremos del sistema BIND (*Berkeley Internet Name Domain*) y el servidor *named*.

El fichero `/etc/nsswitch.conf` es el que guía qué sistema utilizaremos y en qué orden. Por ejemplo, si la entrada de este fichero es la siguiente:

```
hosts: files dns
```

nos está indicando que primero vaya a resolver buscando en el fichero `/etc/hosts` (opción *files*) y luego vaya al sistema de DNS (opción *dns*).

La configuración del sistema de búsqueda para el DNS se encuentra en el fichero `/etc/resolv.conf`.

```
[root@linux entrega05-1]# more /etc/resolv.conf
nameserver 195.235.113.3
nameserver 80.58.0.33
nameserver 150.214.4.34
```

En este ejemplo, se encuentran indicados los servidores de nombres a utilizar en el orden de preferencia, del primero al último.

5.3. Configuración gráfica

5.3.1. Con Fedora

Creación de las interfaces de red.

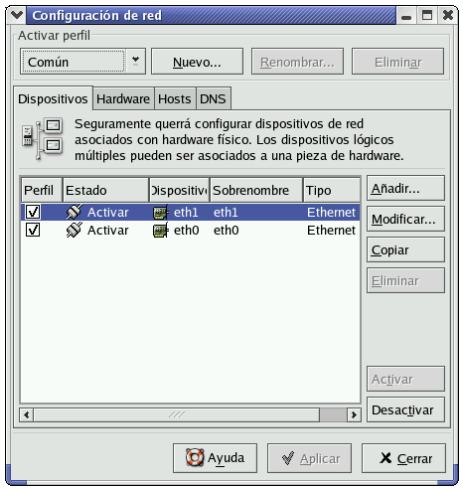
Para un sistema Fedora en modo gráfico, podemos lanzar desde Gnome

 **→Configuración del Sistema→Red**

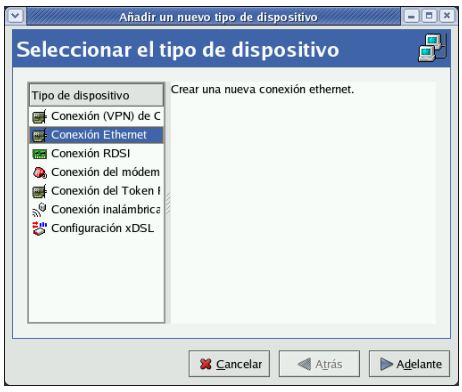
o directamente desde una xterm[26]

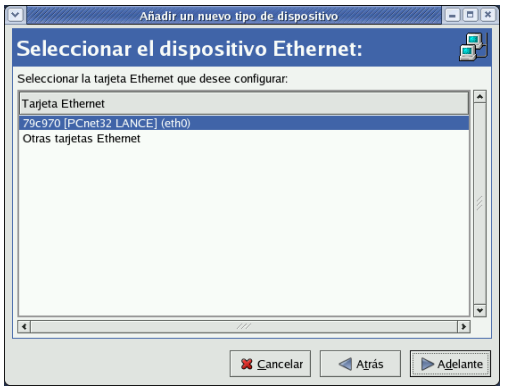
```
# neat &
```

[26] O con `system-config-network`. La captura gráfica no se corresponde con la que aparecería por defecto.



Lo usual es que la tarjeta haya sido detectada y configurada en el arranque/instalación y desde esta ventana podremos editarla. Si no es así, pulsando sobre **Nuevo** podemos configurar nuestra nueva interfaz de red de área local. Seleccionamos **Conexión Ethernet** y debemos elegir la tarjeta correspondiente



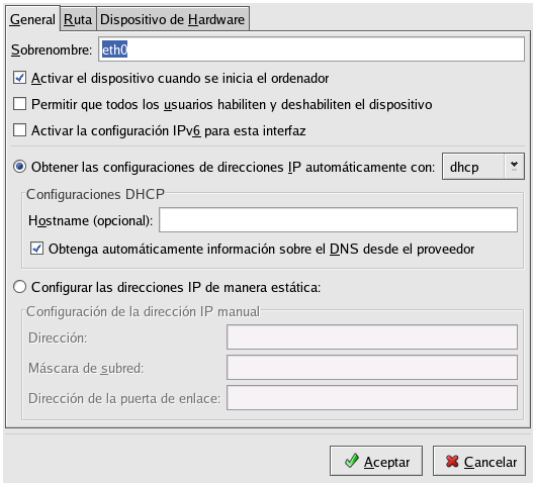


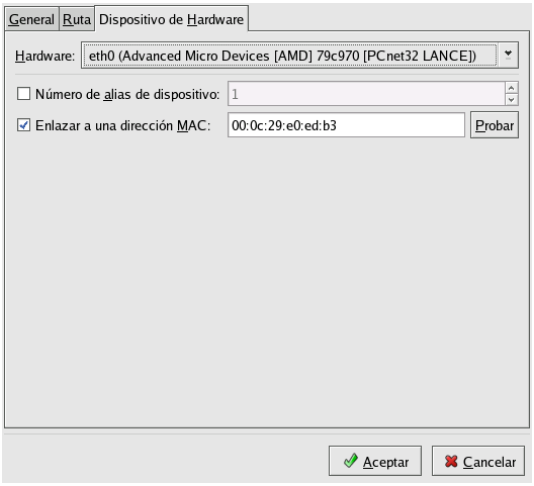
Una vez definidos correctamente los valores hardware de nuestra tarjeta, tendremos que configurar los dispositivos de red asociados con el hardware físico.

Tendremos la posibilidad de permitir que la configuración se obtenga de un servidor de alguno de estos protocolos (DHCP, BOOTP) que se la proporcionará al arrancar, o bien, si marcamos la casilla **Configurar las direcciones IP de manera estática** podremos introducir la dirección IP (172.26.0.2), la máscara de red (255.255.255.0) y la **Dirección de la Puerta de enlace predeterminada** (172.26.0.1).[27]

Si tenemos un router ADSL u otro Linux que hace de pasarela, es el momento de poner aquí su dirección IP para que podamos salir al exterior. Si no, debemos dejar esta casilla en blanco.

Si en la ventana principal de **neat** hacemos doble clic sobre un dispositivo ya instalado en el sistema (o pulsamos sobre el botón **Modificar**) podemos, además de cambiar las opciones anteriores, acceder a más posibilidades de configuración[28]:





[27] Recordemos que el encaminamiento IP es el proceso por el que una máquina decide por dónde dirigir un paquete IP que haya recibido.

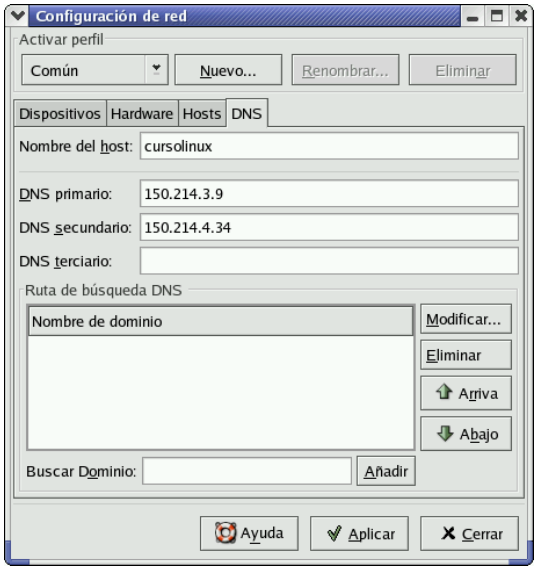
[28] Normalmente seleccionaremos la opción de activar el interfaz en el arranque (**Activar dispositivo cuando arranca el sistema**), pero no permitiremos que cualquier usuario pueda desactivarla en un sistema en el que puede haber usuarios que no son administradores del sistema (**Permitir a todos los usuarios ...**).

Desde la pestaña **Dispositivo de Hardware,** se nos permitirá asociar nuestro dispositivo a una determinada dirección MAC, o asignar un alias a nuestro interfaz de red (permite la posibilidad de que un interfaz de red tenga varias direcciones IP distintas).

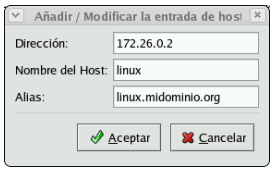
Si tenemos que añadir rutas adicionales[29], seleccionamos la pestaña **Ruta**. Para **Añadir** una nueva ruta, debemos especificar la dirección de la red, su máscara y la puerta de enlace para llegar a ella.[30] Así de fácil hemos configurado nuestra interfaz de red Ethernet, sin tener que utilizar los comandos de bajo nivel `ifconfig` y `route` vistos anteriormente.

Configuración del sistema DNS

Pulsamos en la pestaña **DNS** e introducimos las direcciones IP de nuestros servidores de nombres. Se trata de rellenar los datos necesarios en estos campos. Necesitamos conocer el nombre de nuestro servidor de Internet, que lo escribiremos en el campo **Nombre del host** (no es necesario) y los números DNS de nuestros servidores de nombres. En el caso de la red de ejemplo con la que estamos trabajando, escribiríamos como DNS 195.235.113.3, 80.58.0.33 y 150.214.4.34, que serían los DNS primario, secundario y terciario. Quedaría:



Si deseamos resolver nombres localmente (modificar el fichero `/etc/hosts` en modo gráfico) pulsaremos sobre la pestaña **Hosts**. Podemos modificar las entradas de ese fichero o añadir más pulsando sobre **Nuevo**



[29] No es lo normal. En el 99 % de los casos debería bastarnos con la ruta de nuestra red local y la Puerta de Enlace predeterminada.

[30] Para comprobar que funciona correctamente podemos hacer `ping` a una máquina de fuera de nuestra red y ver si realmente los paquetes pueden salir al exterior.

```
$ping 150.214.4.34
```

Si nos llegan los paquetes de vuelta, estupendo. En caso contrario, debemos asegurarnos de que hemos realizado correctamente la configuración de las rutas, y por si acaso, reiniciando nuestra máquina.

Llegados a este punto, pulsamos **Activamos** el interfaz y cerramos. Para comprobar que realmente resolvemos los nombres, podemos hacer

`$ping thales.cica.es`

Lo primero que hace la máquina será traducir el nombre `thales.cica.es` a su dirección IP que es con la que trabajan las tarjetas de red. Después mandará los paquetes a la dirección indicada, a través del router si no estamos en la misma red.

A modo de resumen

Para un sistema RedHat o Fedora, la configuración que hemos hecho se guardaría en el directorio `/etc/sysconfig/`, que contiene los ficheros que leerá el sistema al arrancar y activar la red.

El fichero `/etc/sysconfig/network` contendrá algo parecido a esto[31]:

```
NETWORKING=yes
HOSTNAME="linux"
FORWARD_IPV4="no"
GATEWAYDEV="eth0"
GATEWAY="172.26.0.1"
```

El fichero `/etc/sysconfig/network-scripts/ifcfg-eth0` contendrá la configuración para la tarjeta de red (`eth0`)[32]:

```
DEVICE=eth0
BOOTPROTO=static
IPADDR=172.26.0.2
NETMASK=255.255.255.0
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
NETWORK=172.26.0.0
BROADCAST=172.26.0.255
PEERDNS=no
GATEWAY=172.26.0.1
```

Si modificamos con un editor alguno de estos ficheros y deseamos releer la configuración ejecutaremos

```
# /etc/rc.d/init.d/network reload
```

5.3.2. Con Guadalinex (Debian)

Creación de las interfaces de red.

Para un sistema Guadalinex podemos configurar la red en modo gráfico lanzando desde Gnome

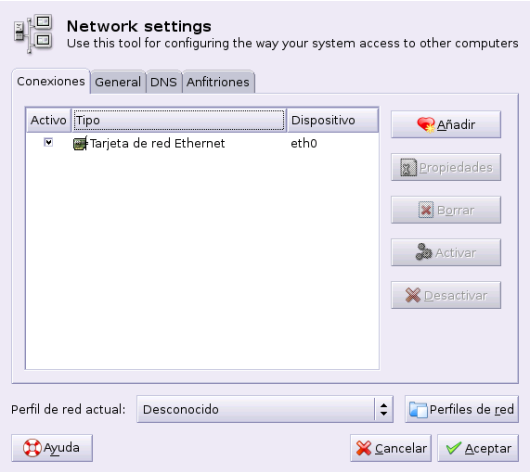
Aplicaciones→Configuración→Sistema→Red

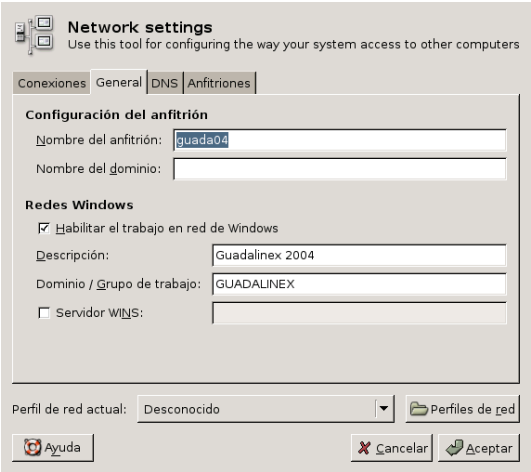
o directamente desde una **xterm**:

[31] Para conocer las opciones de este fichero se puede consultar `/usr/share/doc/initscript-x.x.x/sysconfig.txt`

[32] Además de poder asignar la IP al interfaz de forma estática (`static`), podemos optar por asignar a la variable BOOTPROTO los valores `dchp` o `bootp`.

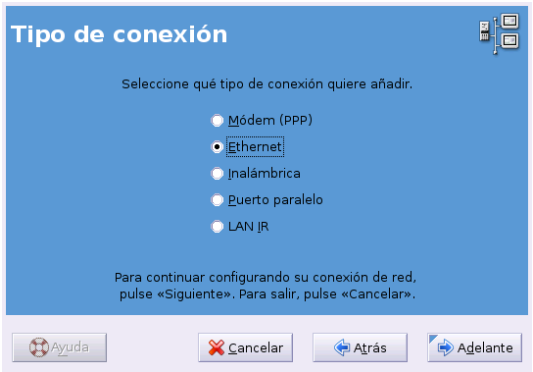
```
# network-admin &
```

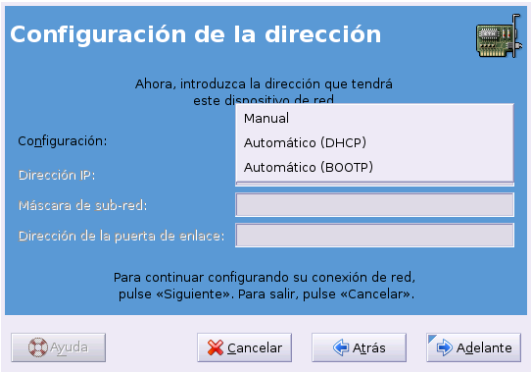




Si pulsamos sobre la pestaña **Conexiones** podremos optar por configurar nuestra red. Lo usual es que la tarjeta haya sido detectada y configurada en el arranque/instalación y desde esta ventana podremos editarla.

Si no es así, pulsando sobre **Añadir** podemos configurar nuestra nueva interfaz de red de área local. Seleccionamos **Conexión Ethernet** y debemos elegir la tarjeta correspondiente



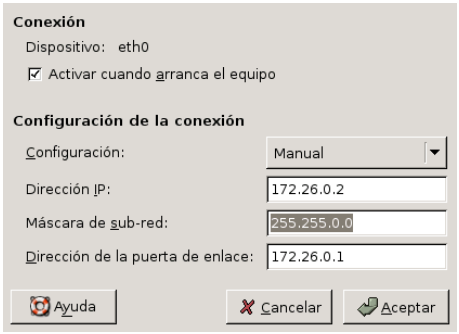


Tendremos la posibilidad de permitir que la configuración se obtenga de un servidor de alguno de estos protocolos (DHCP, BOOTP) que se la proporcionará al arrancar, o bien, si optamos por mantener la **Configuración Manual** podremos introducir la dirección IP (172.26.0.2), la máscara de red (255.255.255.0) y la **Dirección de la Puerta de enlace** (172.26.0.1)[33].

Si tenemos un router ADSL u otro Linux que hace de pasarela, es el momento de poner aquí su dirección IP para que podamos salir al exterior. Si no, debemos dejar esta casilla en blanco.

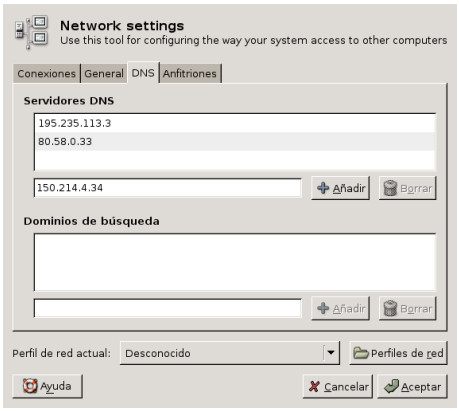
Si en la ventana principal del programa nos situamos sobre un dispositivo ya instalado en el sistema y pulsamos sobre el botón **Propiedades,** podemos cambiar las opciones anteriores o acceder a otras posibilidades de configuración. Deberíamos dejar marcada la opción de activar el interfaz en el arranque (**Activar cuando arranca la computadora**):

[33]El encaminamiento IP es el proceso por el que una máquina decide por dónde dirigir un paquete IP que haya recibido.

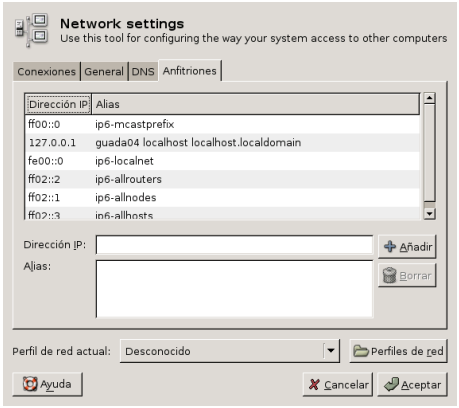


Configuración del sistema DNS

Pulsamos en la pestaña **DNS** e introducimos las IP de nuestros servidores de nombres. Se trata de rellenar los datos necesarios en estos campos, necesitamos conocer el nombre de nuestro servidor de Internet, que lo escribiremos en el campo **Nombre del dominio** (no es necesario) y los números DNS de nuestros servidores de nombres. En el caso de la red de ejemplo con la que estamos trabajando escribiríamos como DNS 195.235.113.3, 80.58.0.33 y 150.214.4.34, que serían los DNS primario, secundario y terciario. Quedaría:



Si deseamos resolver nombres localmente (modificar el fichero `/etc/hosts` en modo gráfico) pulsaremos sobre la pestaña: **Anfitriones**. Podemos modificar las entradas de ese fichero o añadir más pulsando sobre **Añadir**



Llegados a este punto, después de **Aceptar** y volver a la pestaña **General**, nos garantizamos que esté activo el interfaz verificando que esté marcada la casilla **Estado** y cerramos. Para comprobar que realmente resolvemos los nombres, podemos hacer

`$ping thales.cica.es`

Lo primero que hace la máquina será traducir el nombre `thales.cica.es` a su dirección IP que es con la que trabajan las tarjetas de red. Después mandará los paquetes a la dirección indicada, a través del router si no estamos en la misma red.

A modo de resumen

Para un sistema Guadalinex, la configuración que hemos hecho se guardaría en el directorio `/etc/network/`, contiene los ficheros que leerá el sistema al arrancar y activar la red. El contenido del fichero `/etc/network/interfaces` será similar a[34]:

```
auto␣lo
iface␣lo␣inet␣loopback

auto␣eth0
iface␣eth0␣inet␣static
————→name␣Tarjeta␣de␣red␣Ethernet
————→address␣172.26.0.2
————→netmask␣255.255.255.0
————→broadcast␣172.26.0.255
————→network␣172.26.0.0
————→gateway␣172.26.0.1
```

Listado 5.1: /etc/network/interfaces

 Si modificamos con un editor este fichero y deseamos releer la configuración ejecutaremos

`# /etc/init.d/networking restart`

5.4. Conexión a Internet: RTB y ADSL.

Linux e Internet van cogidos de la mano, sin Internet Linux posiblemente estaría "arrumbado" en el cajón de alguna universidad y no sería lo que es hoy. En este apartado vamos a configurar (en modo gráfico[35]) la conexión a Internet de nuestro equipo. Se va a realizar la conexión intentando que sea lo más estándar y guiada posible.

5.4.1. Conexión con módem

Supondremos que nos asignan la dirección de forma dinámica, como ocurre con la mayoría de proveedores de Internet.

Antes de proceder a realizar la conexión a Internet usando un módem necesitamos una serie de datos:

1. Módem:
   a) Tipo de módem, puerto serie[36] al que está conectado.

[34] Para conocer las opciones de este fichero
`$man interfaces`
El fichero `/etc/hostname` contendrá el nombre de la máquina.

[35] Si se desea realizar la conexión a Internet con script se puede consultar el curso de Linux Thales-CICA 2003, disponible en `http://www.iesmurgi.org`, sección de descargas.

[36] Si no lo sabemos y tenemos Windows instalado, podemos usarlo para conocerlo.

b) IRQ y direcciones de E/S.

c) Velocidad del módem.

2. Datos relativos al proveedor (entre paréntesis los que usaremos de ejemplo[37]):

   a) Dominio de acceso (cica.es)

   b) Número de teléfono de acceso (950542000)

   c) Nombre de usuario (codigo_centro@cica)

   d) Contraseña (********)

   e) Método de autentificación (CHAP o PAP)

   f) Dirección IP del servidor de nombres de dominio (DNS: 195.235.113.3, 80.58.0.33 y 150.214.4.34).

Configuración del módem.

Lo primero que tenemos que conocer antes de iniciar el proceso de conexión a internet es saber si nuestro módem funcionará con Linux. Además, puede que necesitemos saber a qué puerto serie está conectado.

En Linux todo son ficheros, y los puertos serie también. Así, cada "fichero" `/dev/ttySx` se corresponde con el puerto de comunicaciones del MS-DOS

Linux	MS-DOS
ttyS0	COM1
ttyS1	COM2
ttyS2	COM3
ttyS3	COM4

El mejor sitio para saber si nuestro módem funciona con Linux:

- `http://freewebhosting.hostdepartment.com/g/gromitkc/winmodem.html`, o en castellano
- `http://freewebhosting.hostdepartment.com/g/gromitkc/winmodem_es.html`

Una página en la que encontrar información si tenemos problemas con el módem:

- `http://wiki.escomposlinux.org/Escomposlinux/EscomposlinuxHardware`

En líneas generales, para los distintos tipos de módem podemos establecer que:

Módem Internos:

Si nuestro módem no es PCI no debería haber ningún problema. Pero la mayoría de ellos son:

Winmódem:

La mayoría de los módem internos PCI no son módem completos y sólo son módem "software". Han aparecido drivers para que algunos modelos de pseudomódem puedan funcionar bajo Linux. Para saber si el nuestro es uno de los que están soportados lo mejor es mirar en las páginas

[37] En general sólo necesitaremos los 4 primeros

- Linux Winmodem Support `http://linmodems.org/`
- Winmodems no son modems `http://freewebhosting.hostdepartment.com/g/gromitkc/pci_list.html`
- Linmodem-HOWTO `http://www.tldp.org/HOWTO/Linmodem-HOWTO.html`

En general, y aunque estén soportados, no son fáciles de configurar y nuestra experiencia es que incluso los soportados dan bastantes problemas.

De Linux Winmodem Support `http://linmodems.org/` podemos bajarnos la utilidad `ScanModem`, con ella podemos testear nuetro WinModem y con la información obtenida intentar configurarlo. Al estar traducida su forma de uso en `http://linmodems.technion.ac.il/linmodems_support_sp.htm` os remitimos a esa Web en el caso de que necesitéis usarla.

Módem Externos:

Al puerto serie: En general no presentan ningún problema, se autodetectan.

USB Muchos son winmodems, aunque están mejor soportados que sus "hermanos" internos o PCI. Para saber si nuestro modelo está soportado, podemos revisar `http://freewebhosting.hostdepartment.com/g/gromitkc/usblist.html`. Si nuestro módem es de este tipo y al ejecutar (como root)

```
#modprobe cdc-adm
#dmesg
```

obtenemos de salida algo similar a:

```
KERNEL: usb.c: ttyACM0: USB ACM device
```

es que nuestro núcleo lo detecta y podremos trabajar con él como si de un módem serie se tratase.[38]. ¿Y si no sale nada?, casi que mejor pensar en otro modelo.

Conexión con Fedora

La conexión a Internet en modo gráfico es un "juego de niños". Sólo hay que ejecutar (como root)

→Herramientas del sistema→Asistente de configuración de Internet[39]

[38] En este caso hay que comprobar si nuestro dispositivo es `/dev/usb/ttyACM0` o `/dev/ttyACM0`

[39] Equivale a ejecutar desde un terminal el comando:

```
#internet-druid
```

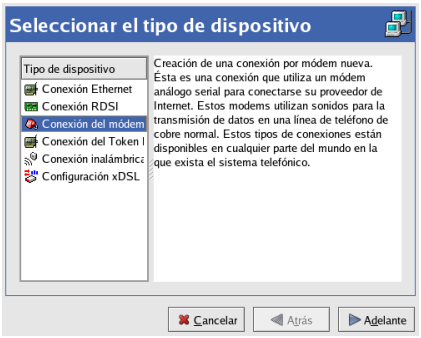
o bien

```
#system-config-network-druid
```

o el ya visto

```
#neat
```

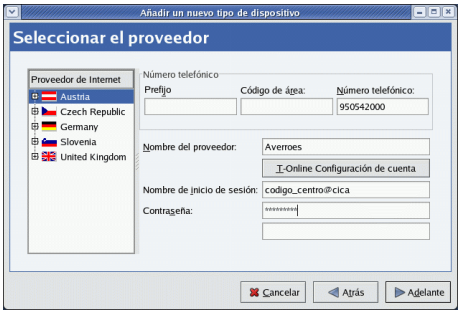
¡Todos los caminos conducen a Roma!



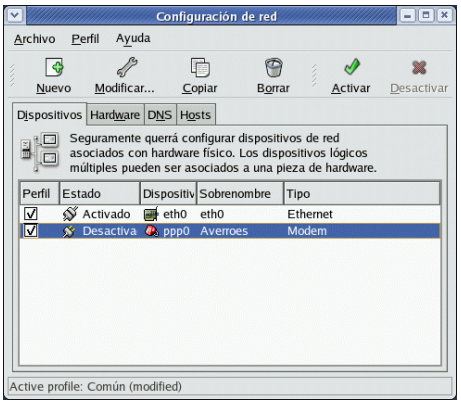
En la ventana que aparece seleccionamos **Conexión del módem** y, al pulsar sobre el botón **Adelante,** comenzará el proceso de autodetección del módem[40].

Puede ocurrir que no detecte el módem, en este caso aparecerá un mensaje de advertencia. Si nos ocurre esto, podemos usar el programa `minicom` para, una vez seguros de que funciona bien y del puerto en el que está conectado, optar por introducirlo de forma manual.

El paso siguiente es configurar la conexión de acceso a partir de los datos de nuestro servidor. En esta pantalla introduciremos el número de teléfono del nodo local al que llamar (por ejemplo 950542000), el nombre que le vamos a dar a esta conexión, el nombre de usuario de nuestra conexión a Internet y la contraseña de acceso. Después se nos preguntará si obtenemos la IP de forma dinámica (es lo habitual), en general dejaremos las opciones por defecto y **Adelante**

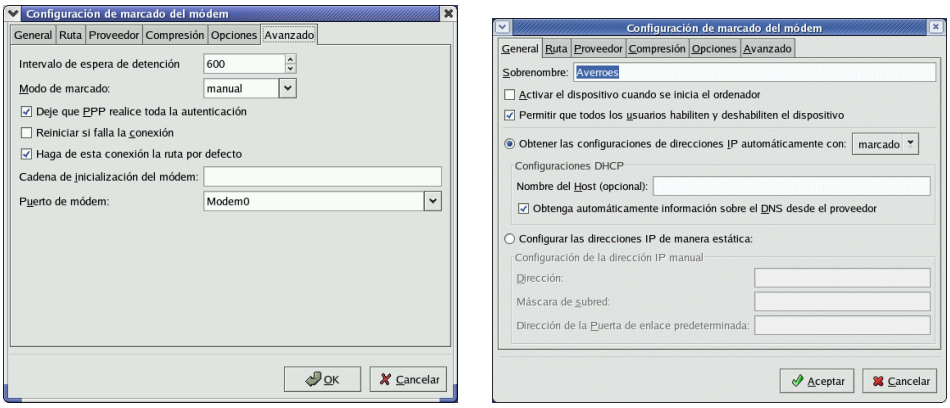


Si continuamos, aparecerá una ventana solicitándonos confirmación sobre los datos introducidos y, por último, la que indica que ya hemos terminado el proceso



[40] El programa `wvdial` es el encargado de "chequear" los puertos.

Mediante el botón **Nuevo** podemos añadir cuantas cuentas de acceso a Internet tengamos a nuestra disposición. El resto son de uso común y sólo comentaremos un poco las posibilidades que se nos brindan al ejecutar **Modificar**



De esta ventana lo interesante es que permite que activemos la posibilidad de que todos los usuarios puedan conectar a internet (opción por defecto) y que debemos dejar activa la casilla de obtener automáticamente información sobre el DNS desde el proveedor[41].

Desde la pestaña correspondiente a **Proveedor** podemos modificar los datos relativos a nombre de usuario, contraseña, etc. Por último, en la ventana a la que se accede en **Avanzado** hay que dejar activas las opciones que aparecen marcadas y pulsar **OK**

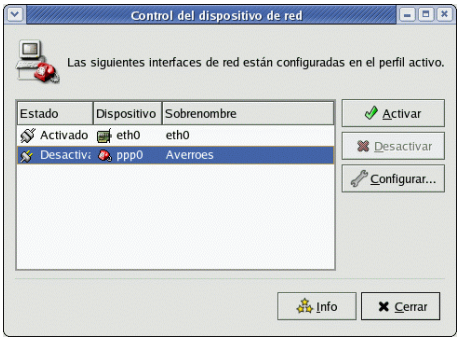
Ahora vamos a testear que nuestra conexión funciona como debe. En la ventana final de configuración hay que pulsar sobre **Activar** (se nos pedirá confirmación de guardar los cambios) y se inicia el proceso de activación del dispositivo de red *ppp0*. Si todo ha ido bien, en el campo **Estado** aparecerá la palabra **Activar**.

Listo, ya podemos comprobar con Mozilla ( ) o el navegador que más nos guste que podemos navegar por la red. Para desconectar de internet sólo tenemos que **Desactivar** el dispositivo.

Cuando queramos conectarnos usaremos el programa `system-control-network` o con menús:

→Herramientas del sistema→Control de dispositivos de red

Desde esta ventana podemos Activar/Desactivar la conexión a internet así como cualquier otro dispositivo de red de nuestra máquina.



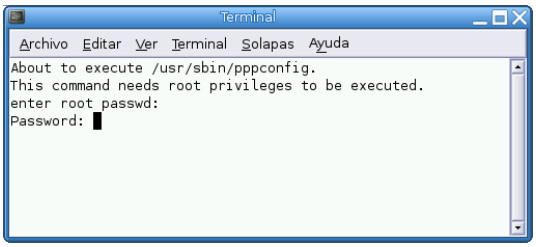
Conexión con Guadalinex

La conexión a Internet es sencilla usando el programa `pppconfig`[42]. Podemos acceder a la aplicación de dos modos diferentes:

[41] Si optamos por hacerlo manualmente: véase 5.3.1 en la página 81

[42] Es mejor que el resto de utilidades gráficas de Guadalinex, es igual de fácil trabajar con ella pero su fiabilidad es mayor.

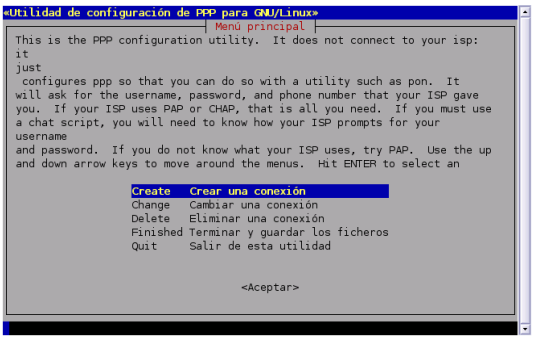
- **Aplicaciones→Menú Debian→Aplicaciones→Sistema→Administración→pppconfig**

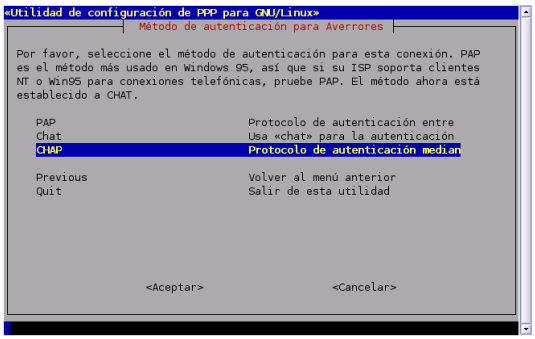


- Desde un terminal de texto **xterm** escribimos (como root):

  `#pppconfig &`

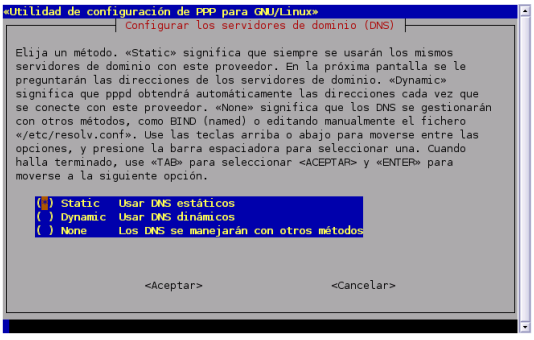
Si optamos por crear una cuenta nueva, la primera pregunta que nos va a hacer es el nombre que le vamos a dar a esta conexión

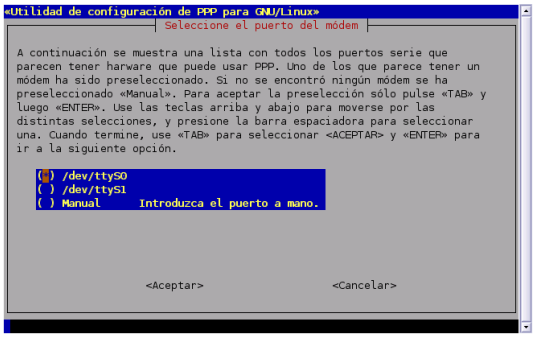




Una vez que "manifestamos" nuestro acuerdo tenemos que introducir las IPs de los servidores de nombres (uno a uno) y el modo de autentificación. Primero deberíamos probar con CHAP y, si no funciona, intentarlo con PAP.

A continuación debemos optar por la forma en que nuestro servidor de acceso nos va a facilitar la IP de los servidores de nombres: de forma estática o dinámica. Vamos a suponer que lo hace de forma estática aunque en la mayoría de los servidores actuales podríamos optar por la segunda opción (dinámica) y después escribiremos el nombre de usuario de nuestra conexión a Internet.





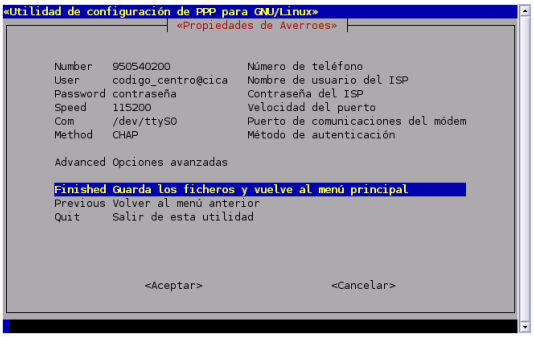
Casi hemos terminado, ahora introducimos la contraseña de acceso. Lo siguiente es optar por seleccionar la velocidad entre el módem y el puerto serie (no es la velocidad del módem; si el ordenador es antiguo quizá haya que poner 57600).

Seleccionamos ahora el tipo de marcado, mejor por "tonos"[43], y pasamos a introducir el número de teléfono del nodo local al que llamar (en el ejemplo 950542000)

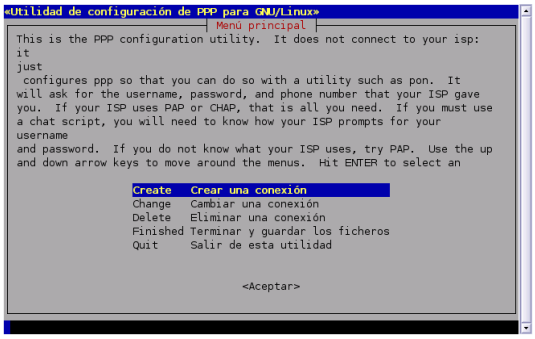
[43] **Marcación decádica por pulsos**

Llega el momento de "la verdad", el de la autodetección del módem. Si lo autodetecta, felicidades: ya es "coser y cantar". Si no es así tendremos que intentar configuarlo de forma manual y, para eso, hay que echar mano de las páginas antes comentadas.

Ya sólo nos falta comprobar que los datos introducidos son correctos, arreglar aquello que esté mal y guardar los datos de esta conexión a Internet. Si todos los datos son correctos seleccionaremos la opción [**Finished**] y [**Aceptar**].



Notar que volvemos a la pantalla inicial de la aplicación. De modo que, si tenemos más de una cuenta de acceso, podemos introducirla ahora:



Cuando terminemos de configurar nuestras cuentas seleccionamos **Quit**→**<Aceptar>** y pasamos a intentar conectar, para eso escribiremos desde un terminal de texto:

```
$pon Averroes
```

Listo, ya podemos comprobar con Mozilla o el navegador que más nos guste que podemos navegar por la red.

Para desconectar de Internet sólo tendremos que ejecutar desde una **xterm**:

```
$poff Averroes
```

Ya hemos configurado la conexión a Internet

Por defecto, en nuestro escritorio tenemos un icono que nos da acceso a un navegador web, Mozilla Firefox, pero hay otros muchos más[44].

De Wikipedia, la enciclopedia libre.

"La marcación decádica por pulsos consiste en el envío por el teléfono de la información numérica, en forma de pulsos, a la central telefónica automática para que ésta le conecte con el teléfono deseado.

Los pulsos los genera el teléfono mediante un dispositivo mecánico denominado disco de marcar, el cual consiste en un disco giratorio provisto de diez agujeros, de aqui lo de decádica, numerados del 0 al 9.

La marcación decádica por pulsos se ha venido utilizado en exclusividad desde los orígenes de la telefonía automática hasta tiempos relativamente recientes.

En la actualidad, aunque las modernas centrales digitales siguen aceptando este tipo de marcación, se utiliza mayoritariamente la marcación por tonos multifrecuencia, mucho más eficiente que la aquí descrita."

[44] Además de los comentados: konqueror, amaya, ... o navegadores en modo texto (lynx, ...)

Pistas para detectar problemas

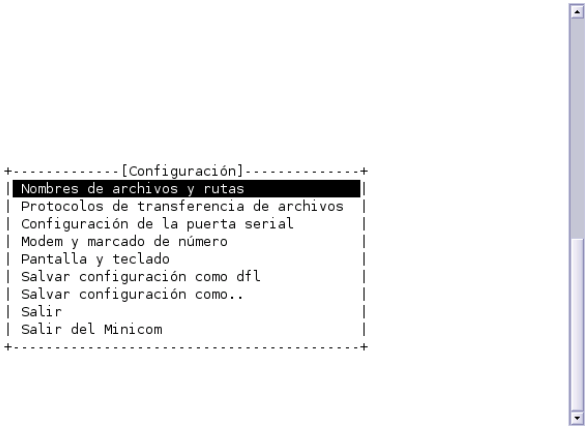
Puertos serie en Linux Disponemos de un comando que nos permite configurar el puerto serie, se trata del comando `setserial`

Para conocer cómo trabajar con él podemos ejecutar (desde un terminal de órdenes):
`$setserial --help`
para obtener una ayuda básica de los parámetros que admite:
`$man setserial`
para obtener la ayuda completa sobre el programa[45].
Para conocer el estado de un puerto serie podemos ejecutar (como root)
`#setserial -a /dev/ttySx`
donde `ttySx` es el correspondiente al puerto de comunicaciones del DOS .
setserial

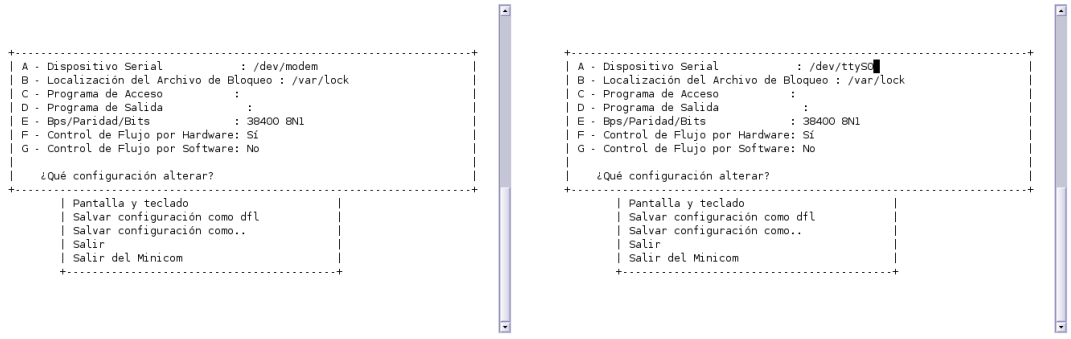
Programa minicom Se trata de una utilidad que, en caso de tener dificultades con la configuración del módem, nos puede ayudar a detectar en dónde puede estar el problema, se trata del programa `minicom`[46]. Con él podemos comprobar si el módem está bien conectado.

Para activarlo tenemos que ejecutar desde un terminal gráfico el comando:

```
$ minicom -s
```



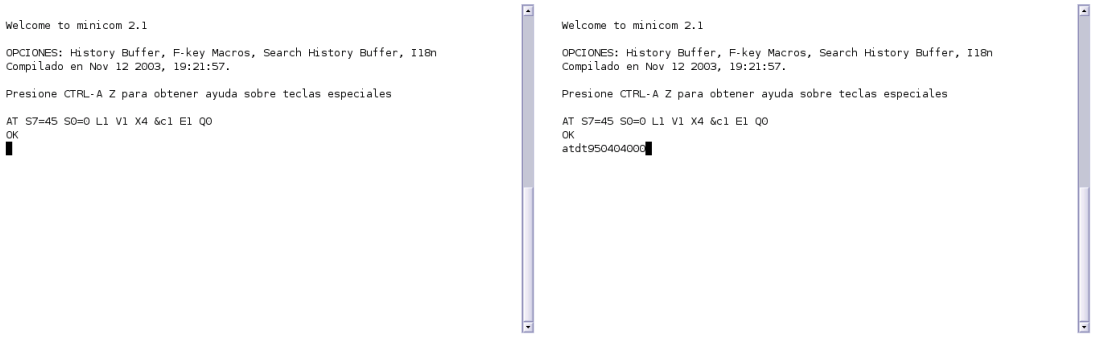
Con la opción `-s` optamos por entrar en el menú de configuración anterior[47]. Si marcamos en **Configuración de la puerta serial**, accederemos a:



Debemos optar por seleccionar el puerto serie al que está conectado nuestro módem, observar que en el ejemplo hemos sustituido `/dev/modem` por `/dev/ttyS0`. Una vez seleccionado el puerto

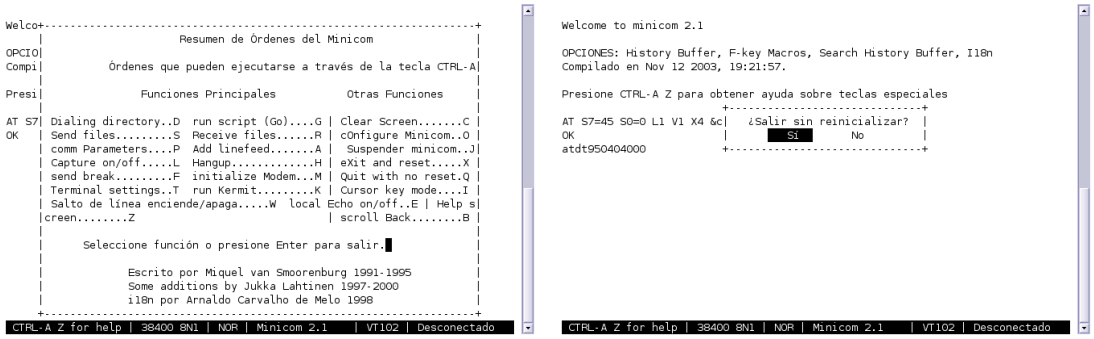
[45] Se sale de la ayuda con `q`.
[46] Programa terminal de comunicaciones
[47] Sólo tendremos que usar esta opción la primera vez que ejecutemos el programa.

adecuado (que no se nos olvide pulsar la tecla [**Intro**]) optaremos por [**Salvar configuración como dfl**] y después [**Salir**]. Si nos aparece una pantalla similar a la que sigue, y siempre que nos aparezca el [**OK**] final, es que todo ha ido bien. Si no es así habrá que reconfigurar la conexión del módem y volver a comprobarlo.



En la segunda captura, estamos comprobando que el teléfono está bien configurado, para eso, llamamos a un número de telefono y comprobamos que da tono de marcado, el comando a usar es `ATDTnúmero_teléfono`.

Para acceder al menú de este programa hay que utilizar la combinación de teclas [**Ctrl**]+[**a**] y después pulsar la letra [**z**]. Por ejemplo, para salir del programa hay que pulsar [**Ctrl**]+[**a**], después [**z**] y por último [**q**].



Si hemos configurado correctamente el programa, para acceder de nuevo a él, ya sólo hemos de escribir:

```
$minicom
```

Conocer la IP asignada Si deseamos comprobar si la conexión se produce con éxito y conocer la dirección IP que se nos ha asignado dinámicamente, podemos escribir desde un terminal la orden[48]:

- Debian:

  `#plog averroes`

- Red Hat (y Debian)

  `#tail -f /var/log/messages`

  Para cancelar el comando `tail` y dejar de visualizar las líneas que van saliendo hay que pulsar [**ctrl**]+[**c**]

[48] `$ /sbin/ifconfig` nos da información similar.

veremos entonces una serie de mensajes que nos muestran cuál es el estado de la conexión, si ésta ha tenido éxito nos tienen que aparecer dos líneas del tipo:

```
local  IP address xxx.xxx.xxx.xxx
remote IP address xxx.xxx.xxx.xxx
```

donde esos números indican las direcciones IP asignadas dinámicamente a nuestra máquina y al servidor.

ping Una forma de saber si realmente hemos conectado bien, es hacer un `ping`. Este comando comprueba que llegamos a la máquina remota que queremos comprobar. Por ejemplo, `$ping 150.214.5.11`, para llegar al servidor de los cursos. El comando nos dirá si llegamos o devuelve error.

```
$ping 150.214.5.11
PING 150.214.5.11 (150.214.5.11) from 195.24.23.44 : 56(84) bytes of data.
64 bytes from 150.214.5.11: icmp_seq=1 ttl=64 time=0.040 ms
64 bytes from 150.214.5.11: icmp_seq=2 ttl=64 time=0.037 ms
64 bytes from 150.214.5.11: icmp_seq=3 ttl=64 time=0.035 ms
64 bytes from 150.214.5.11: icmp_seq=4 ttl=64 time=0.039 ms
--- 150.214.5.11 ping statistics ---
4 packets transmitted, 4 received, 0% loss, time 3006ms
rtt min/avg/max/mdev = 0.035/0.037/0.040/0.007 ms
```

Para terminar pulsamos [**Ctrl**]+[**c**]

Conectamos pero no salimos fuera Si, pese a que conectamos, no podemos visualizar páginas web, revisar:

- El fichero `/etc/resolv.conf`. Puede que pese a tener marcada la opción de que obtengamos la información sobre los DNS desde el proveedor, esto no sea así. En este caso lo mejor es que configuremos esto de forma manual. Podemos conseguirlo de dos formas:
  - Gráfica: véase 5.3.1 en la página 81 y 5.3.2 en la página 84
  - Texto Utilizando un editor de textos escribiremos en el fichero `/etc/resolv.conf`:
    ```
    domain cica.es
    nameserver 195.235.113.3
    nameserver 80.58.0.33
    nameserver 150.214.4.34
    ```
    cambiando los datos del ejemplo anteriores por los de nuestro servidor de acceso.
- Si tenemos una tarjeta de red, revisar la salida del comando:

  `#netstat -ar`

  si sale una línea del tipo:

  `default 192.168.0.254`

  u otra IP local, es que hemos configurado como *Gateway* una máquina local. Debemos eliminarla.

5.4.2. ADSL

Hoy en día suele ser más habitual la conexión a internet a través de una línea ADSL que a través de un módem RTB. Como en el caso anterior, debemos saber primero si nuestro módem ADSL está soportado por Linux o no. Disponemos de dos tipos básicos de módems ADSL:

- USB: son los más difíciles de configurar, serían el equivalente a los Winmodems RTB. No todos funcionan correctamente.
- Módem-router: no presentan ningún problema.

Usando un módem router

El proceso es similar a lo expuesto en 5.3 en la página 78, por tanto, sólo daremos una pinceladas para aclarar esta cuestión. La configuración gráfica de la conexión ADSL la podemos realizar con esas mismas herramientas, usaremos:

Debian

Aplicaciones→Configuración→Sistema→Red

o directamente desde una **xterm**:

```
# network-admin &
```

Fedora

→Configuración del Sistema→Red

o directamente desde una xterm

```
# neat &
```

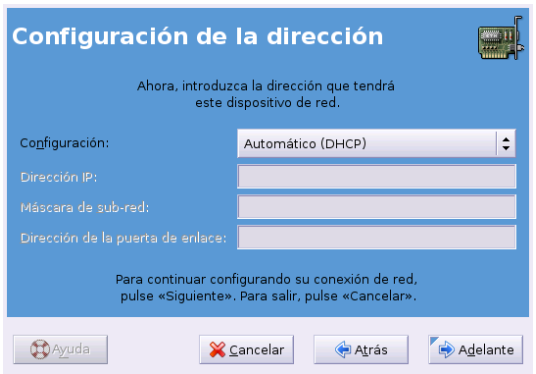
Lo único a tener en cuenta en ambos casos es cómo obtenemos la configuración desde el servidor (DHCP, BOOTP) o manual.

Cuando contratamos una ADSL debemos fijarnos si nuestro ISP[49] (proveedor de acceso a Internet) nos facilita una IP fija o IP dinámica.

En un principio cuando se contrataba una línea ADSL te *"regalaban"* la IP fija, ahora no (salvo ofertas). De todos modos, en cualquier momento, si lo deseamos, podemos solicitar una IP fija a nuestro ISP, previo pago mensual, claro.

Aclarado ésto, configuraremos nuestra conexión ADSL en función de cómo tengamos el módem router configurado:

Monopuesto sólo hay que decirle a la tarjeta de red que obtenga la configuración de direcciones IP automáticamente con DHCP y marcar la opción de obtener automáticamente información sobre el DNS desde el proveedor (capturas de GuadaLinex).



Multipuesto debemos diferenciar dos posibilidades en función de la forma en que esté configurado nuestro router:

[49] Del inglés *Internet Service Provider*.

En modo DHCP es la forma más usual, la conexión se realiza igual que en el caso Monopuesto.

Asignando la IP de forma Manual: en este caso tendremos que poner una dirección IP a nuestro interfaz de red de la misma red que el router (este dato lo tenemos que conocer a través del proveedor de acceso). Así por ejemplo, si nuestro módem router tiene la IP local 172.26.0.1 y como máscara de red 255.255.255.0 sólo le diremos al interfaz de red que use como *Gateway* la IP local del router y las IP de los servidores de nombres. Si optamos por poner de IP a nuestra máquina linux la dirección 172.26.0.2, quedaría (captura para Fedora):

Y para añadir los servidores de nombres (fichero `/etc/resolv.conf`), en la ventana principal de ambos programas, pulsaremos sobre la pestaña DNS e introducimos las IP de nuestros servidores de nombres. Se trata de rellenar los datos necesarios en estos campos. Necesitamos conocer el nombre de nuestro servidor de Internet. En el caso de la red del ejemplo con la que estamos trabajando escribríamos como DNS 195.235.113.3, 80.58.0.33 y 150.214.4.34, que serían los DNS primario, secundario y terciario (véase 5.3.1 en la página 81 y 5.3.2 en la página 84).

Llegados a este punto, y tras activar el interfaz de red, para saber si todo está bien podemos:

- Abrir un navegador Web y comprobar que salimos fuera.
- Hacer un ping a una máquina remota:

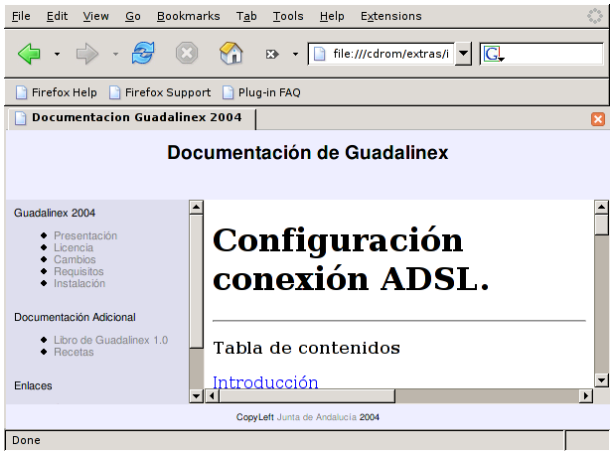
```
$ping mileto.cica.es
PING mileto.cica.es (150.214.5.11) from 80.30.154.77 : 56(84) bytes of data.
64 bytes from mileto.cica.es (150.214.5.11): icmp_seq=1 ttl=53 time=101 ms
64 bytes from mileto.cica.es (150.214.5.11): icmp_seq=2 ttl=53 time=97.3 ms
64 bytes from mileto.cica.es (150.214.5.11): icmp_seq=3 ttl=53 time=113 ms
64 bytes from mileto.cica.es (150.214.5.11): icmp_seq=4 ttl=53 time=93.8 ms
64 bytes from mileto.cica.es (150.214.5.11): icmp_seq=5 ttl=53 time=101 ms
--- mileto.cica.es ping statistics ---
5 packets transmitted, 5 received, 0% loss, time 4033ms
rtt min/avg/max/mdev = 93.849/101.676/113.561/6.657 ms
```

Pistas para conectar con módem USB usando Guadalinex.

Si nuestro módem es USB, tendremos que "confiar" en que algún "alma caritativa" haya resuelto ya el problema ya que sólo algunos modelos (cada vez más) son fácilmente configurables.

La dificultad de trabajar con modelos basados en esta tecnología reside en que tienen que estar soportados por Linux. En primer lugar deberíamos revisar la información contenida en el CD de Guadalinex, accesible desde el escritorio **Conexión a Internet** fichero `LEAME-PRIMERO.txt`. En él, se nos informa de que si nuestro modelo es un Comtrend CT-350, Sagem Fast 800 o un Conexant Accessrunner USB estamos de suerte ya que podremos trabajar con ellos en Linux. Como añadido deberíamos consultar

- La Web `http://www.guadalinex.org/guadapedia/index.php/Receta:_C%C3%B3mo_configurar_una_conexi%C3%B3n_ADSL_%28Guadalinex_2004%29`
- La ayuda contenida en el CD, subdirectorio `extras/info/recetas`,



Podemos obtener más información para nuestro modelo USB en:

- `http://wiki.escomposlinux.org/Escomposlinux/EscomposlinuxHardware`

También pueden ser de ayuda las páginas:

- `http://personal.telefonica.terra.es/web/adslusb/`
- `http://cp4218.sourceforge.net/`

Para los tres modelos de módem USB antes comentados, previo a iniciar la configuración debemos instalar el paquete adecuado.

Capítulo 6

Linux como Router y Cortafuegos

Estaría perdido sin mis cortafuegos (Mark J. Cox, Director adjunto de Red Hat)

6.1. Router Linux

Como ya hemos visto, un sistema Linux puede funcionar haciendo la función de router. Simplemente, se conecta a dos o más redes y sabiendo, a partir de su tabla de rutas, por qué interfaz puede alcanzar cada red, dirige los paquetes entre una y otra red. Éste sería un funcionamiento como router estático[1]. Para ello es necesario que la característica *IP forwarding* (o routing entre interfaces) esté activada en el núcleo de nuestro sistema. Si no está activada, cuando llegue un paquete por un interfaz, no podrá dirigirse hacia otro interfaz, porque no estará permitido el routing entre ellos.

Podemos comprobar si está activo el IP forwarding con el comando:

```
[root@linux entrega04-1]# cat /proc/sys/net/ipv4/ip_forward
0
```

En este caso, la salida `0` indica que no está activado el routing del kernel.

Podemos activarlo mediante el comando[2]:

```
#echo 1 >/proc/sys/net/ipv4/ip_forward
```

Los dos métodos siguientes, harían el cambio permanente. El primero consiste en cambiar el fichero de configuración del kernel (`/etc/sysctl.conf`), poniendo el siguiente valor:

```
net.ipv4.ip_forward = 1
```

En Red Hat o Fedora, también podremos hacerlo en el fichero `/etc/sysconfig/network`, poniendo lo siguiente:

```
FORWARD_IPV4=true
```

Otra opción para que Linux funcione como router, es la de instalar un servidor especializado de routing en nuestro sistema Linux. Por ejemplo, *zebra* (`www.zebra.org`), que incorpora protocolos de routing dinámicos como BGP-4, RIP u OSPF.

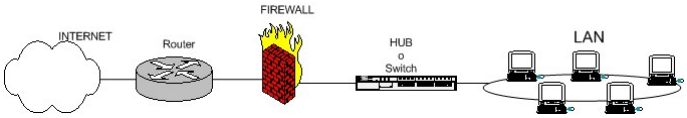
La sola capacidad de routing, nos permite conectar varias redes, pero nos surge otro problema mayor: la seguridad. Esto hace que la mayoría de sistemas Linux (y no Linux) que interconectan redes, incorporen capacidades de seguridad, como los cortafuegos.

[1] Aprovechando incluso un viejo 486, podemos tener un router funcional.

[2] El problema es que con este método, el cambio no sería permanente. Al reiniciar el ordenador lo habremos perdido.

6.2. Cortafuegos Linux

Un cortafuegos (o firewall) es un componente o conjunto de componentes que restringen el acceso entre una red protegida e Internet, o entre varias redes. Incluye tanto componentes hardware como software, configuraciones y definición de políticas de seguridad[3].



Su propósito es doble: proteger los sistemas y servicios internos de los ataques del exterior, y controlar los accesos hacia Internet de nuestros usuarios.

6.2.1. Clasificación de cortafuegos

Según el nivel de la pila de protocolos sobre el que trabajan, podemos clasificar los cortafuegos en:

Cortafuegos de Nivel de red: El control de tráfico a nivel de red consiste en analizar todos los paquetes que llegan a un interfaz de red, y decidir si se les deja pasar o no, en base al contenido de los mismos: protocolo, dirección de origen, dirección de destino, puerto origen y puerto destino fundamentalmente.

↬Su operación se asemeja a la de un guardia de tráfico que en un cruce decide qué coches pueden pasar y cuáles no, dependiendo de su origen y su destino.

Puesto que analizar esta información es muy sencillo, este tipo de cortafuegos suelen ser muy rápidos y transparentes al usuario. Se suelen denominar de filtrado de paquetes (*packet filter*). Una mejora sobre este tipo de cortafuegos, serían los de Inspección de Estado (*Stateful Inspection*) que además, inspeccionan en el interior de los paquetes para comprobar si cumplen las políticas de seguridad.

Cortafuegos de Nivel de aplicación: Se basan en la instalación de intermediarios (proxies), también conocidos como pasarelas (*application gateways*). El cliente, situado en un lado del cortafuegos, habla con un intermediario situado en el propio cortafuegos. Este intermediario lo identifica, registra sus peticiones y, si está permitido, las encamina hacia el verdadero servidor situado al otro lado del cortafuegos. La contestación regresa por el mismo camino, quedando igualmente registrada.

El control se hace interceptando las comunicaciones a nivel de aplicación, modificando el protocolo para incluir medidas adicionales de seguridad. El cortafuegos debe conocer los detalles del protocolo de cada servicio que intercepta, analizar su correcto funcionamiento y añadir los pasos de control precisos a los mismos. Por ejemplo, *squid* es un proxy que debe conocer el protocolo HTTP para recoger las peticiones de los navegadores cliente y redirigirlas al servidor destino. El cortafuegos es, por tanto, mucho más inteligente y posee un control más fino de todo el proceso de comunicación, aunque esto supone una mayor carga de trabajo y penalización en eficiencia. Además, normalmente exigen realizar modificaciones en la aplicación del usuario, como por ejemplo, decirle al navegador que utilice el proxy.

6.2.2. Terminología de cortafuegos

En una arquitectura de sistema cortafuegos, encontramos una serie de términos o componentes como son:

host bastión: (también se denomina *gates*) es un sistema especialmente asegurado, pero que puede recibir ataques por estar accesible desde Internet. Tiene como función ser el punto de

[3]Una política de seguridad nos dice qué es lo que se puede hacer y qué no se puede hacer en una red.

contacto de los usuarios de la red interna de una organización con otro tipo de redes. El host bastión filtra tráfico de entrada y salida, y también oculta la configuración de la red hacia fuera.

El *filtrado* también se conoce como *screening*, y a los dispositivos que lo implementan se les denomina *chokes*.

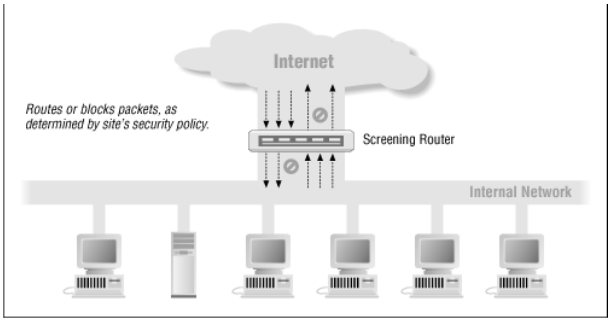
De la combinación de estos elementos, filtrado y host bastión, surgen las siguientes arquitecturas de cortafuegos.

6.2.3. Arquitecturas de cortafuegos

Según los componentes que incluya el cortafuegos y su ubicación, obtenemos las siguientes arquitecturas:

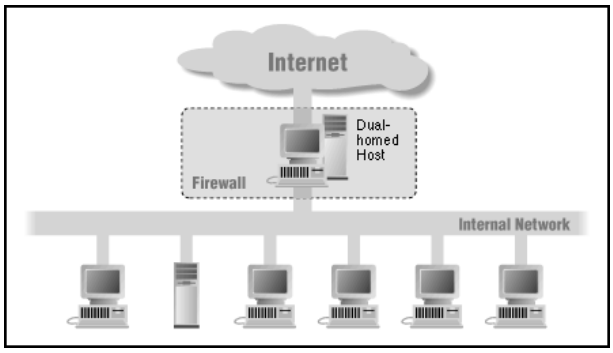
1. Cortafuegos de filtrado de paquetes (*Screening Router*)

   Un firewall sencillo puede consistir en un dispositivo capaz de filtrar paquetes, un choke. Basado en aprovechar la capacidad de algunos routers - denominados screening routers - para hacer un enrutado selectivo, es decir, para bloquear o permitir el tránsito de paquetes mediante listas de control de acceso en función de ciertas características de las tramas, de forma que el router actúe como pasarela de toda la red.



2. *Dual-Homed Host* (Host en dos zonas)

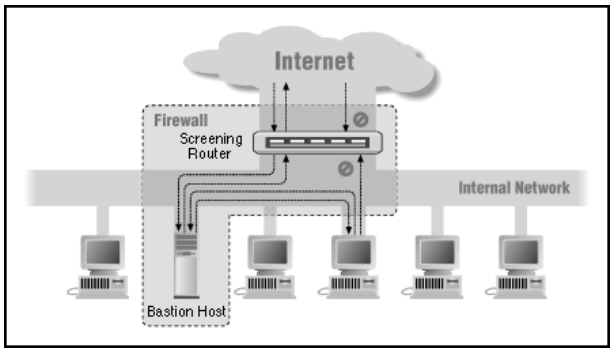
   Este modelo de cortafuegos está formado por máquinas equipadas con dos tarjetas de red y denominadas dual-homed hosts, en las que una de las tarjetas se conecta a la red interna a proteger y la otra a la red externa. En esta configuración el choke y el bastión coinciden en el mismo equipo.



3. *Screened Host*

   Un paso más en términos de seguridad de los cortafuegos es la arquitectura screened host o choke-gate, que combina un router con un host bastión, y donde el principal nivel de seguridad proviene del filtrado de paquetes (es decir, el router es la primera y más importante línea de defensa). En la máquina bastión, único sistema accesible desde el exterior, se ejecutan los

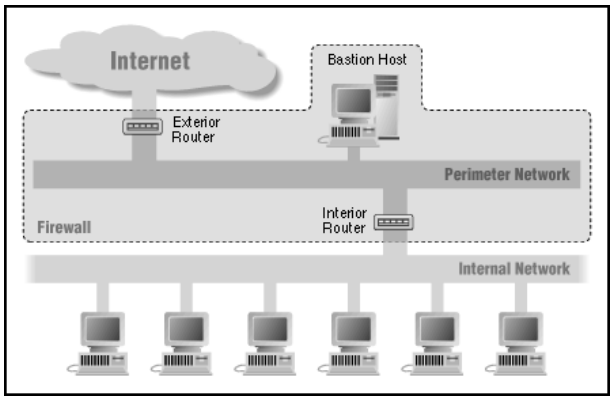
proxies de las aplicaciones, mientras que el choke se encarga de filtrar los paquetes que se puedan considerar peligrosos para la seguridad de la red interna, permitiendo únicamente la comunicación con un reducido número de servicios.



4. *Screened Subnet* (DMZ)

   La arquitectura Screened Subnet, también conocida como red perimetral o Zona Desmilitarizada (*De-Militarized Zone* o DMZ) añade un nivel más de seguridad en las arquitecturas de cortafuegos, situando una subred (la DMZ) entre las redes externa e interna, de forma que se consiguen reducir los efectos de un ataque con éxito al host bastión. En los modelos anteriores, si la seguridad del host bastión se ve comprometida[4], la amenaza se extiende automáticamente al resto de la red. Como la máquina bastión es un objetivo interesante para muchos piratas, la arquitectura DMZ intenta aislarla en una red perimetral de forma que un intruso que accede a esta máquina, no consiga un acceso total a la subred protegida.

   Screened subnet es la arquitectura más segura, pero también la más compleja; se utilizan dos routers, denominados exterior e interior, conectados ambos a la red perimetral como se muestra en la figura. En esta red DMZ, que constituye el sistema cortafuegos, se incluye el host bastión y también se podrán incluir sistemas que requieran un acceso controlado, como baterías de modems, el servidor web o el servidor de correo, que serán los únicos elementos visibles desde fuera de nuestra red.



6.2.4. Firewalls sobre linux

Los cortafuegos pueden estar basados en la combinación de un hardware y un software especializado, constituyendo "cajas" preparadas para realizar esa función, como por ejemplo los Nokia IP, los Cisco PIX o las cajas de StoneSoft. Otros, sin embargo, están construidos sobre un sistema operativo de propósito general, que se blinda[5] y prepara para funcionar como cortafuegos. En este

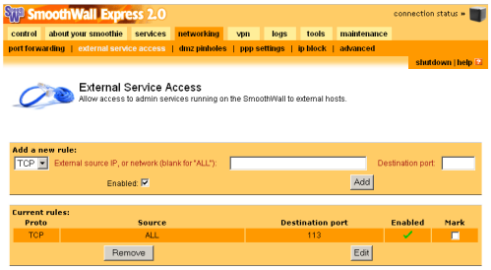
[4] Alguien consigue tomar el control o ejecutar comandos desde él.
[5] Se refuerzan las características de seguridad. *Hardening* en inglés.

caso, podemos tener un sistema operativo Solaris, Windows o Linux sobre los que se ejecuta un software como el Firewall-1 de Check Point.

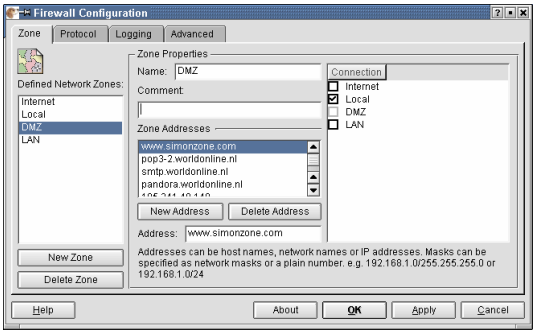
Linux se puede encontrar en las dos opciones. Hay empresas que utilizan linux sobre un hardware específico, dando una caja[6] lista para ser utilizada como cortafuegos, como el iForce de Sun Microsystems. También podemos coger un sistema Linux sobre un PC e instalar un cortafuegos. Lo más normal, en ambos casos, es utilizar las características de cortafuegos que incorpora el propio kernel de Linux: ***iptables***.

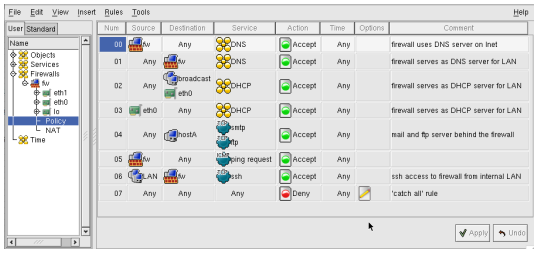
Un sistema cortafuegos puede ofrecer sus servicios a una red que se sitúa detrás de él, denominándose *cortafuegos de la red*, o el cortafuegos protege a la propia máquina en la que se ejecuta, denominándose *cortafuegos personal*.

Para un cortafuegos de red, podemos utilizar distribuciones especializadas, que incorporan facilidades de administración por medio de navegador, integran proxys y utilidades adicionales. Podemos encontrar en este segmento a SmoothWall (`www.smoothwall.org`) o IPCop (`www.ipcop.org`). El proceso a seguir consiste en descargarse la imagen ISO de la distribución, grabar el CD e iniciar la instalación. El proceso es muy guiado y nos va haciendo las preguntas correspondientes para configurar el cortafuegos a nuestro gusto. La apariencia de ambas distribuciones la podéis observar en las siguientes figuras.

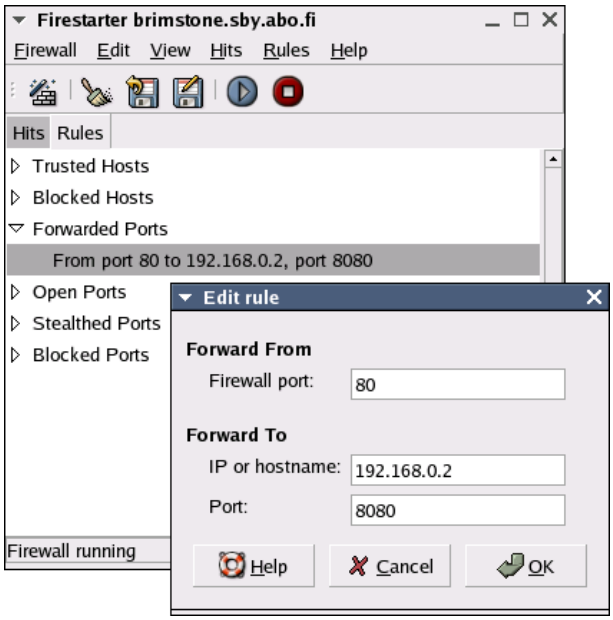


Para un cortafuegos personal, trabajamos con nuestra distribución favorita y se configura iptables para añadir seguridad. Sin embargo, el trabajar directamente con iptables puede ser muy engorroso y existen interfaces gráficos que nos permiten la configuración y el trabajo de forma más fácil. En este segmento encontramos a GuardDog (`http://www.simonzone.com/software/guarddog`), Firestarter (`http://firestarter.sourceforge.net`) o Firewall Builder (`http://www.firewallbuilder.org`), detalles de los cuales se muestran a continuación.

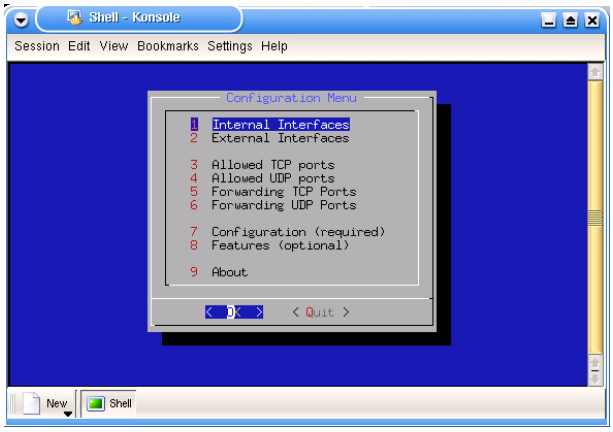




[6]En inglés, al concepto de entregar una caja cerrada y lista para enchufar en la red, se le llama *appliance*.



Incluso tenemos utilidades en terminal alfanumérico como `Firewall-jay` (`http:/firewall-jay.sourceforge.net`), simples pero potentes.



6.2.5. ¿Qué es iptables?

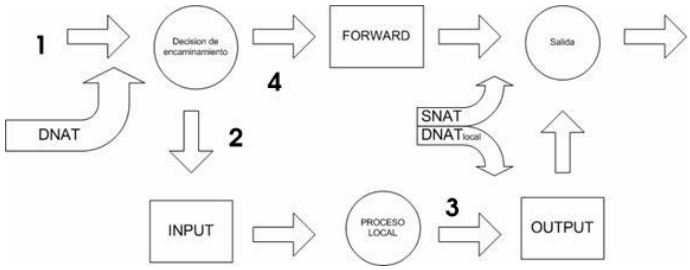
Como hemos comentado anteriormente, la principal herramienta de cortafuegos para Linux a partir de los kernels 2.4[7], es iptables. Iptables reemplaza al anterior *ipchains* de los kernels de la versión 2.2 y a *ipfwadm* de los kernels 2.0. La función de *iptables* es la de establecer, mantener e inspeccionar las reglas de filtrado de paquetes IP en el núcleo de Linux.

Iptables decide qué paquete de información puede pasar, según unos criterios que se almacenan en unas listas. Las listas se componen de reglas con un orden determinado, donde la última regla introducida será la última regla en aplicarse.

Cuando un paquete llega, se mira en qué lista debe aplicarse. En esa lista (iptables las llama tablas) se empieza por la primera regla. Si la regla no es aplicable al paquete, se pasa a la siguiente regla. Cuando una regla es aplicable (*match*) al paquete, se ejecuta la acción que haya sido definida en la regla (descartar el paquete, aceptarlo, enrutarlo, etc).

Veamos el camino que seguiría un paquete en el kernel utilizando iptables:

[7] Y sigue vigente para los actuales kernels 2.6



Cuando iptables recibe el paquete (1), se comprueba si el destino final es nuestra propia máquina o es otra, porque estemos funcionando como router/gateway o cortafuegos. Para los paquetes que van a la propia máquina se aplican las reglas INPUT (2) y para paquetes que van a otras redes o máquinas se aplican las reglas FORWARD (4). Las reglas de OUTPUT (3) se aplican cuando un paquete es enviado desde nuestra máquina al exterior.

INPUT,OUTPUT y FORWARD son los tres tipos de reglas de filtrado (FILTER). Antes de aplicar esas reglas es posible aplicar reglas de NAT[8] y de MANGLE[9].

La estructura de un comando iptables es la siguiente :

```
iptables -t [tabla] -[opciones] [regla] [criterio] -j [acción]
```

Veamos qué significa cada elemento:

-t [tabla] Esta parte del comando especifica cuál es la tabla en la que aplicamos la regla. Existen 3 tipos de tablas: FILTER, NAT y MANGLE, siendo *filter* la tabla por defecto si se omite esta parte del comando.

Filter es la tabla donde se añaden las reglas relacionadas con el filtrado de paquetes.

Nat se refiere a las conexiones que serán modificadas por el firewall, como por ejemplo, enmascarar conexiones, realizar redirecciones de puertos, etc.

Mangle es parecido a Nat, pero tiene la posibilidad de modificar más valores del paquete.

-[opciones] Las opciones básicas del comando son las siguientes :

A para añadir (*Append*) una regla.

L es para listar (*List*) las reglas.

F es para borrar (*Flush*) todas las reglas o en el caso de que INPUT, FORWARD o OUTPUT sean dados como argumento, se borrarán las reglas asociadas sólo a esa clase.

P establece la política (*Policy*) por defecto del firewall. Por defecto es aceptar todas las conexiones.

[regla] Reglas válidas son INPUT, FORWARD y OUTPUT.

[criterio] Aquí es donde se especificarán las características del paquete que casará con esta regla. Algunos ejemplos son:

-s : dirección de origen (source). Puede ser una dirección IP o una red. `-s 192.168.1.0/24`

-d : dirección de destino. `-d 84.56.73.3`

-p : tipo de protocolo (TCP,UDP,ICMP). `-p TCP`

–**sport** : puerto de origen

–**dport:** puerto de destino `--dport 23`

[8] Se usan para hacer redirecciones de puertos o cambios en las IPs de origen y destino

[9] Modifica paquetes, pero es más rica y potente que Nat. Con Mangle podemos modificar cualquier aspecto del paquete (flags, TTL, etc).

-i = in-interface : el interfaz[10] por el que se entra `-i eth0`

-o = –out-interface: el interfaz[11] por el que se sale `-o ppp0`

-j [accion] Aquí establecemos qué es lo que hay que hacer con el paquete. Las posibles acciones son :

ACCEPT: aceptar el paquete.

REJECT o DROP: desechar el paquete. La diferencia entre ellos reside en que DROP descartará el paquete silenciosamente y REJECT emitirá un paquete ICMP Port Unreachable, indicando que está cerrado el puerto.

REDIRECT redirigir el paquete a donde se indique en el criterio del comando.

LOG archiva el paquete para su posterior análisis.

Hay dos maneras de implementar un firewall, según la política por defecto que especifiquemos:

1. Política por defecto ACEPTAR: Se aceptan por defecto todos los paquetes. Sólo se denegará lo que se diga explícitamente. El equivalente sería la política de acceso a un bingo: pueden entrar todas las personas, excepto aquellas cuyo DNI aparezca en la lista de acceso prohibido.
2. Política por defecto DENEGAR: Todo está denegado, y sólo se permitirá pasar por el firewall aquello que se permita explícitamente. El equivalente sería el acceso a la cámara de cajas de seguridad de un banco. El acceso está prohibido a todo el mundo y se habilita una lista de personas autorizadas a entrar. Para un cortafuegos, se recomienda aplicar esta política por defecto.

Ejemplos de iptables

Veamos una regla que acepta conexiones al puerto 80 de nuestro equipo:
`iptables -A INPUT -i eth0 -s 0.0.0.0/0 -p TCP --dport www -j ACCEPT`

Comando `iptables`. Llamada al comando con los argumentos siguientes.

-A *append*, opción para añadir la regla

INPUT aplicable a los paquetes que entran a nuestra máquina

-i eth0: por el interfaz de red eth0

-s 0.0.0.0/0 dirección de origen del paquete. En este caso, cualquier dirección.

-p TCP tipo de protocolo. En este caso TCP.

–**dport** puerto de destino. Se especifica `www`, que mirando en `/etc/services` es el puerto número 80.

-j ACCEPT qué acción se realiza sobre el paquete. Se acepta.

Veamos otro ejemplo:

```
iptables -A INPUT -p tcp -i eth0 -m state --state NEW,ESTABLISHED --dport 22 -j
ACCEPT
iptables -A INPUT -p all -i eth0 -m state --state NEW,INVALID -j DROP
```

La primera regla deja pasar los paquetes con destino a nuestra máquina (INPUT), por protocolo `tcp`, que entren por el interfaz eth0 con destino al puerto 22. Como habéis visto, nos hemos saltado

[10] -i se usa con reglas INPUT y FORWARD
[11] -o se usa con reglas FORWARD y OUTPUT

algo. La opción `-m state` indica que tiene que cargar el módulo de inspección de estado. Es decir, va a escudriñar el interior del paquete, no sólo su filtrado: esto se conoce como inspección de estado. Ahora ya es aplicable la opción `--state NEW, ESTABLISHED`, que indica que pueden pasar los paquetes que abren una nueva conexión[12] (NEW) o pertenecen a una conexión ya establecida (ESTABLISHED).

La segunda regla descarta (`drop`) todas las conexiones entrantes (`INPUT`) de todos los protocolos (`-p all`), que intenten abrir una nueva conexión (`NEW`) o no pertenezcan a una conexión ya establecida (`INVALID`) desde la interfaz eth0.

Firewall Personal con Guadalinex 2004

En Guadalinex 2004 disponemos de un firewall que viene con el sistema. Se llama Firestarter. En realidad es un interface gráfico para iptables, pero que nos será muy útil para configurar nuestro cortafuegos, ya sea como cortafuegos personal[13] o como cortafuegos para proteger una red.

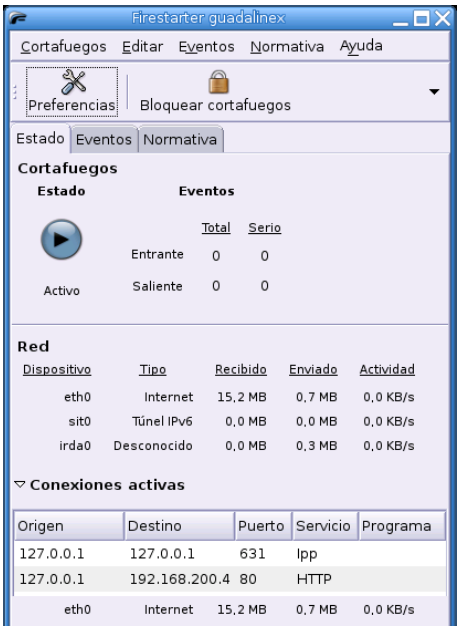
Lo primero que haremos es actualizar a una versión más reciente que la que viene por defecto. Para ello utilizamos `apt-get`

```
#apt-get install firestarter
```

Podemos lanzar la el cortafuegos desde **Aplicaciones→Configuración→firestarter** o ejecutando desde la línea de comandos[14]

```
#firestarter
```

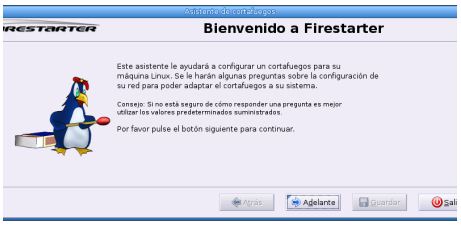
La ventana que nos aparece es la siguiente



Lo más destacable es que estamos viendo las conexiones activas en nuestra máquina. Con lo que no es solamente una ayuda para crear las reglas sino que también nos permite visualizar las conexiones activas en tiempo real.

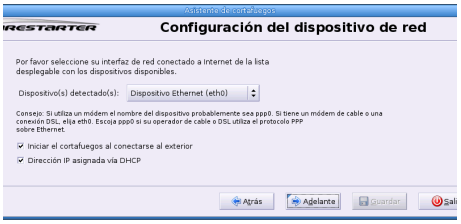
Veamos cómo podemos realizar la configuración. La forma más fácil es con el asistente, que podemos llamar desde la entrada **Cortafuegos** y **Ejecutar asistente**, desde el menú principal.

[12] ¿Recordáis el *three-way andshake* necesario para establecer una nueva conexión TCP?
[13] Solamente protegemos a nuestro propio sistema.
[14] Para ambos métodos, hay que tener privilegios de superusuario



Si pulsamos **Adelante**, pasamos a una ventana en la que podemos seleccionar el interfaz de nuestro sistema que se conecta a Internet. En este caso es `eth0` para conectarnos a un router ADSL. En esta opción estamos seleccionando el firewall personal que protege nuestra máquina del hostil mundo exterior.

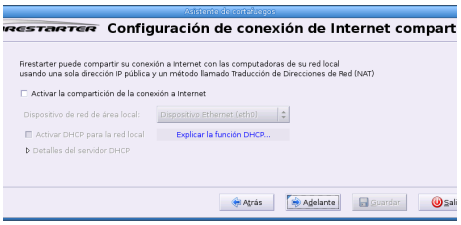
Además, seleccionamos que el cortafuegos se active al conectarnos a Internet.



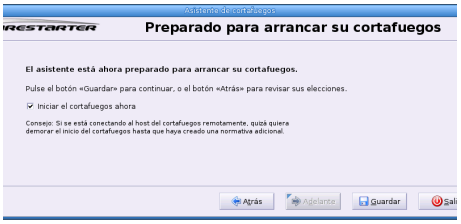
La siguiente pantalla nos permite decidir si el firewall de nuestro sistema además funcionará como pasarela para otros sistemas, convirtiéndose en el guardián de nuestra red.

Para ello, debemos contar con otra interfaz de red distinta de la anterior, que se conectará a la red protegida.

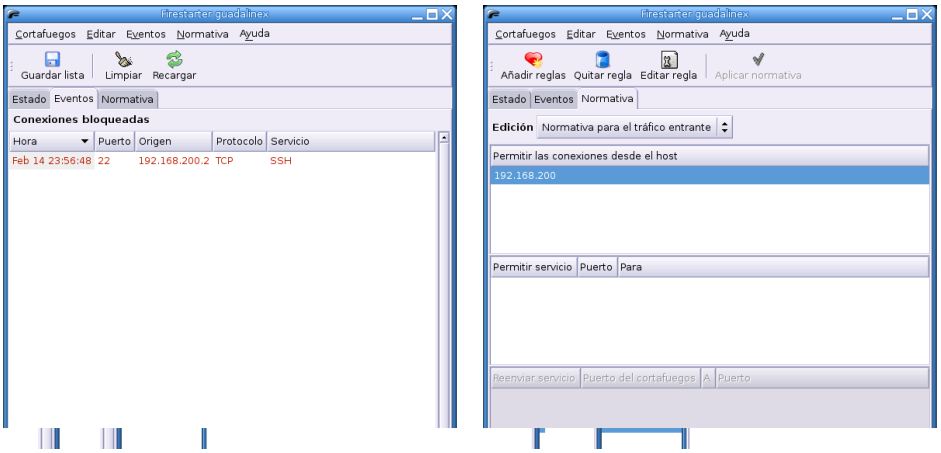
En este caso, como solamente queremos un firewall personal, no activamos la opción **Activar la compartición de la conexión a Internet**. Si quisiéramos que éste sea el cortafuegos de nuestra red, debemos especificar la interfaz conectada a la red de área local interna y si queremos activar el servidor de DHCP para dar direcciones automáticamente.



Pues ya hemos terminado una configuración básica del cotafuegos para uso personal. Podemos guardar las reglas e iniciar el cortafuegos.



La pestaña de **Estado** ya la conocíamos de la primera figura. Las otras dos pestañas son la de **Eventos**, que nos permite ver las conexiones que han sido bloqueadas por el cortafuegos y la de **Normativa**, que nos permite: añadir nuevas reglas y eliminar o modificar reglas, tanto de tráfico entrante como saliente.



Como véis, no es complicado y nos ayuda a mantener a raya a los chicos malos.

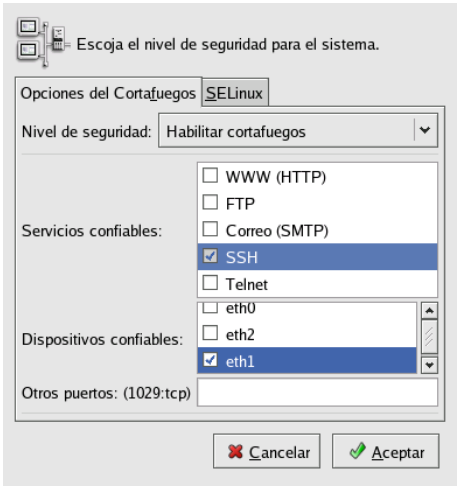
Firewall Personal con Fedora

Veamos las utilidades que Fedora pone a nuestro alcance para configurar un cortafuegos personal.

Para configurar en modo gráfico el firewall, ejecutaremos el programa `system-config-securitylevel` o con menús:

 **→Configuración del sistema→Nivel de seguridad**

La pantalla que se nos presenta es la siguiente[15]:



Las opciones de **Nivel de seguridad** serán *Habilitar cortafuegos* o *Deshabilitado*. En caso de que lo activemos, las conexiones a nuestro ordenador quedarán prohibidas. Deberemos abrir las conexiones que nos interesen, como WWW si tenemos un servidor web que queremos hacer accesible, o permitir conexiones SSH.

El último cuadro nos permite seleccionar los **Dispositivos fiables**. ¿Cuáles son éstos, los que no fallan nunca?. No, seguramente no es la traducción más acertada del término *trusted*, que podría ser más bien confiable y se refiere a los interfaces que, al estar en una red local protegida, consideramos que no nos van a venir ataques a través de ellos y permitimos relajar las conexiones que vengan por esa vía.

[15] En entregas posteriores veremos qué es esto de SELinux

Esta utilidad en realidad es un interfaz gráfico que escribe su configuración en el fichero `/etc/sysconfig/system-config-securitylevel`, como mostramos a continuación.

```
# /etc/init.d/networking restart
[root@linux images]# more /etc/sysconfig/system-config-securitylevel
#Configuration file for system-config-securitylevel
#Copyright (c) 2002 Red Hat, Inc. all rights reserved
--enabled
--trust=eth0
--port=http:tcp
--port=ssh:tcp
```

También existe una utilidad equivalente en modo texto, que ejecutamos mediante[16]

```
#system-config-securitylevel-tui
```

y cuya apariencia mostramos en la figura siguiente:

Si entramos en la opción de **Personalizar**, podemos seleccionar las opciones de apertura de nuestra máquina que deseemos, igual que antes en el modo gráfico.

Los valores de iptables se guardan en memoria cuando se ejecutan y necesitamos que se almacenen en algún sitio de donde poder recuperar la cofiguración al rearrancar el servicio. Mediante las utilidades anteriores, hemos configurado los valores que aparecen en el fichero `/etc/sysconfig/iptables`. Mostremos su valor actual:

```
[root@linux images]# more /etc/sysconfig/iptables
# Firewall configuration written by redhat-config-securitylevel
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:RH-Firewall-1-INPUT - [0:0]
-A INPUT -j RH-Firewall-1-INPUT
-A FORWARD -j RH-Firewall-1-INPUT
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -i eth0 -j ACCEPT
```

[16] También con

```
#lokkit
```

```
-A RH-Firewall-1-INPUT -p icmp --icmp-type any -j ACCEPT
-A RH-Firewall-1-INPUT -p 50 -j ACCEPT
-A RH-Firewall-1-INPUT -p 51 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

¿Podéis reconocer los valores de este fichero con la configuración de la figura anterior? Seguro que sí.

Si queremos almacenar los valores de iptables en un fichero, debemos ejecutar el comando `iptables-save`. Y análogamente, para recuperar desde ese fichero los valores y cargarlos en el kernel, ejecutamos `iptables-restore`.

Para arrancar el servicio iptables, lo hacemos mediante el comando
`#service iptables start`
y para pararlo, mediante
`#service iptables stop`.

Un ejemplo práctico de uso de iptables

Tenemos dos ordenadores conectados en red mediante un cable cruzado. El que tiene acceso a internet (`host0`) dispone de dos tarjetas de red, una que permite salir hacia fuera mediante DHCP (`eth0`) y otra (`eth1`) que se conecta con el otro ordenador de la casa (`host1`).

La configuración del interfaz de red en `host1` es:

IP: 192.168.0.2

MASCARA: 255.255.255.0

PUERTA DE ENLACE: 192.168.0.1

La configuraciòn de la red en `host0` es:

eth0: DHCP

eth1:

IP: 192.168.0.1
MASCARA: 255.255.255.0

Problema: Al hacer ping de un ordenador a otro todo va bien. Pero cuando se hace un ping desde `host1` a una máquina remota, por ejemplo `mileto.cica.es`, nos responde *host desconocido.*

Solución: Un script que nos puede solucionar el problema es

```
[paco@eco ~]#cat ipro
echo 1 > /proc/sys/net/ipv4/ip_forward
#borra las reglas
iptables --flush
iptables --table nat --flush
#Activamos el NAT con enmascaramiento
iptables --table nat --append POSTROUTING -s 192.168.0.0/24 --out-
interface eth0 -j MASQUERADE
iptables --append FORWARD -s 192.168.0.0/24 --in-interface eth1 -
j ACCEPT
#politica para la red local de todo permitido
iptables -A INPUT -s 192.168.0.0/24 -j ACCEPT
```

Si lo ejecutamos como root (`./ipro`) activaremos el filtrado de paquetes. Para que todo funcione hay que poner en `host1` las IP de los servidores de nombres[17] en el fichero `/etc/resolv.conf`

[17] Si no las ponemos saldremos hacia fuera pero no resolveremos nombres

Capítulo 7

Configuración de DHCP

> DHCP es útil para proporcionar de un modo rápido la configuración de red del cliente. ...
>
> Además, si un portátil o cualquier tipo de equipo móvil se configura para DHCP, podrá desplazarse entre distintas oficinas sin tener que volver a configurarlo, siempre y cuando cada oficina tenga un servidor DHCP que permita su conexión a la red. (*Manual de personalización de Red Hat Linux*)

7.1. Introducción

Si a nuestro cargo está el mantenimiento de una red local, el uso de IPs estáticas acarrea algunos inconvenientes de mantenimiento[1]. Estos problemas se acrecientan cuanto más grande es la red ya que la asignación de IPs se puede hacer más compleja. Para intentar solucionar este problema se desarrolló el Protocolo para Configuración Dinámica de Terminales (*Dynamic Host Configuration Protocol* o DHCP [RFC 2131][2]).

Pero, ¿qué es DHCP?

DHCP es un protocolo TCP/IP que proporciona una asignación dinámica y automatizada de direcciones IP además de otro tipo de información añadida, como puede ser la puerta de enlace predeterminada o las IPs de los servidores de nombres. Se basa en UDP y utiliza los puertos 67 y 68, el servidor escucha en el puerto 67 y el cliente en el 68.

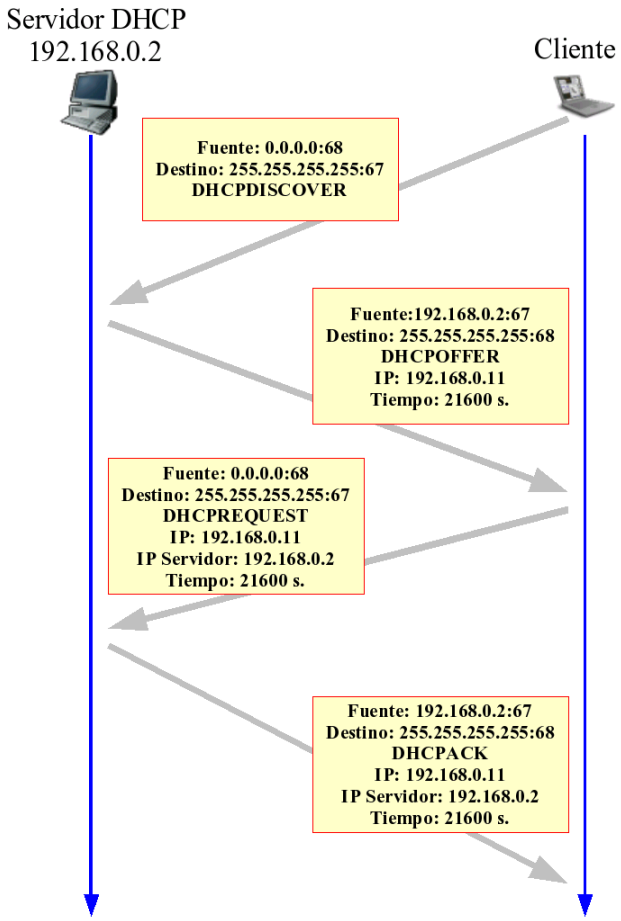
Cuando un ordenador al que no se le ha asignado IP fija se conecta a nuestra red, lo único de que dispone es de la dirección hardware del interfaz de red (*MAC address*). A partir de ese número, la máquina cliente realiza una llamada de difusión que tendrá respuesta por parte de nuestro servidor DHCP, indicándole al cliente, junto con la dirección, algunos parámetros adicionales.

Expliquemos esto un poco más[3]:

[1] Aunque personalmente, en un centro de enseñanaza, prefiero las IP fijas.

[2] Se basó en BOOTP (de *Boot Protocol*) mejorándolo.

[3] Entre paréntesis se incluyen los mensajes DHCP, para ampliar sobre su significado `http://dns.bdat.net/dhcp/x56.html`



1. El cliente difunde un mensaje de descubrimiento DHCP (DHCPDISCOVER), para eso envía un datagrama[4] UDP al puerto 67 con dirección de destino 255.255.255.255 y de origen 0.0.0.0 a la subred local.
2. Los servidores disponibles responden con un mensaje de ofrecimiento DHCP (DHCPOFFER), en ese mensaje se incluye la oferta de IP, el tiempo de concesión de esa IP, ...
3. Tras analizar las ofertas recibidas de los distintos servidores DHCP disponibles en la red, el cliente elige una de las ofertas en función de los parámetros incluidos en el mensaje y envía un mensaje de petición de DHCP (DHCPREQUEST) en que repite los parámetros de configuración.
4. El servidor guarda la asignación y responde con un mensaje (DHCPACK) de reconocimiento ACK en el que se confirman los datos de la configuración.

Como documentación complementaria, además de la ayuda que se instala con el programa, podemos ampliar sobre su uso en:

- Capítulo 18 de *Manual de personalización de Red Hat Linux*
  `http://europe.redhat.com/documentation/rhl9/rhl-cg-es-9/ch-dhcp.php3`
- *DHCPD mini-COMO para Linux*
  `http://es.tldp.org/COMO-INSFLUG/COMOs/DHCPd-Mini-Como/`

[4] Broadcast del cliente para localizar servidores

- `http://www.redes-linux.com/manuales.php`
- Internet Systems Consortium `http://www.isc.org/index.pl?/sw/dhcp/`
- `http://es.wikipedia.org/wiki/DHCP`

7.2. Instalación

En ambas distribuciones se instala por defecto el cliente `/sbin/dhclient`. En Fedora forma parte del paquete de igual nombre (`dhclient`), mientras que en Guadalinex 2004 el paquete que lo contiene es `dhcp3-client`.

Con Linux podemos disponer de servicios DHCP fácilmente, en ambas distribuciones sólo hay que instalar el paquete `dhcp`. La salida que se obtiene en Fedora es

```
# apt-get update; apt-get install dhcp
Leyendo lista de paquetes... Hecho
Creando Árbol de dependencias... Hecho
Nota, seleccionando dhcp en lugar de dhcp
Se instalarán los siguientes paquetes NUEVOS:
  dhcp
0 actualizados, 1 se instalarán, 0 para eliminar y 123 no actualizados.
Necesito descargar 110kB de archivos.
Se utilizarán 311kB de espacio de disco adicional después de desempa-
quetar.
...
Please note that if you are installing the DHCP server for the first
time you need to configure it first. Please stop (/etc/init.d/dhcp
stop) the DHCP server daemon, edit /etc/dhcpd.conf to suit your needs
and particular configuration, and restart the DHCP server daemon
(/etc/init.d/dhcp start).
You also need to edit /etc/default/dhcp to specify the interfaces dhcpd
should listen to. By default it listens to eth0.
NOTE: dhcpd's messages are being sent to syslog. Look there for
diagnostics messages.
Starting DHCP server: dhcpd failed to start -
check syslog for diagnostics.
```

Debian

La salida no se corresponde con las líneas anteriores, y es que en Debian no se iniciará el servicio, ya que antes hemos de configurarlo. Se nos avisa de que antes de ponerlo en marcha hemos de adecuar el fichero `/etc/dhcp.conf` a nuestra red y que podemos restringir los interfces de red a la escucha en el fichero `/etc/default/dhcp`. Para solucionar ambos temas, sigamos leyendo.

7.3. Configuración

7.3.1. De la máquina Linux

El fichero de configuración es `/etc/dhcpd.conf` y el fichero en donde se almacenan las IP asignadas[5] `/var/lib/dhcp/dhcpd.leases`. Analicemos el primero a partir del fichero de ejemplo

[5] Al periodo de préstamo de la dirección IP se le llama alquiler (*lease*). Este tiempo siempre será finito, de esta forma el servicio DHCP puede detectar la retirada del cliente y hacer de nuevo uso de la IP asignada. Si el tiempo se agota, DHCP dispone de un mecanismo que permite la renovación de la IP asignada al cliente por otro periodo de tiempo.

instalado[6] `/usr/share/doc/dhcp-3.0.1/dhcpd.conf.sample` .

Podemos trabajar de dos formas distintas: en modo de actualización DNS *ad-hoc* (no recomendado) y en el modo de actualización en el que interaccionan DHCP-DNS es decir, modo `interim`, es el modo por defecto[7]

```
ddns-update-style interim;
```

Para ignorar las solicitudes de actualización del registro A[8] de los clientes (permite asociar a un nombre la dirección IP)

```
ignore client-updates;
```

Características comunes a la subred especificada

```
subnet 192.168.0.0 netmask 255.255.255.0 {
```

IP de los routers de acceso a Internet (si son varios se separan con comas) y máscara de subred

```
# --- gateway por defecto
     option routers 192.168.0.1;
     option subnet-mask 255.255.255.0;
```

Especifica el nombre del dominio NIS[9] del cliente, el nombre de dominio para la máquina cliente y las IPs de los servidores de nombres (si son varios se separan con comas)

```
     option nis-domain "domain.org";
     option domain-name "domain.org";
     option domain-name-servers 150.214.3.9;
```

Se trata de la forma en que sincronizamos nuestro tiempo en función del Meridiano de Greenwich. Al venir dado en segundos y negativo equivaldría a -5 horas[10].

```
     option time-offset -18000; # Tiempo de la Costa Este Americana
```

IP del servidor NTP[11]

```
#    option ntp-servers 192.168.0.5;
```

IP del servidor WINS[12]. Indica la estrategia que cada cliente de una red NetBIOS debe seguir para realizar el registro de nombres y la resolución[13]. Si no tenemos claro de qué va, es mejor no tocarlo (véanse las notas a pie de página 12 y 13).

```
#    option netbios-name-servers 192.168.0.6;
```

[6] Si bien se estudia para Fedora, es similar para Debian. Para esta distribución, la documentación está en `/usr/share/doc/dhcp`

[7] Para usar el primero tendríamos que quitar esa línea y en su lugar escribir
`ddns-update-style ad-hoc;`
Para saber más de esto: `$man dhcpd.conf`

[8] Veremos qué significa en la siguiente entrega cuando hablemos del DNS.

[9] Normalmente no es utilizado, ni debemos preocuparnos de él.

[10] Para saber exactamente cómo se calcula y qué es este valor, mirar la web
`http://www.cisco.com/warp/public/109/calculate_hexadecimal_dhcp.html`

[11] *Network Time Protocol:* protocolo de comunicaciones que permite sincronizar los relojes de un ordenador con un servidor central de tiempo

[12] WINS es el servicio que resuelve los nombres NetBIOS de las redes "Microsoft" a direcciones IP. Si usamos SAMBA podemos conseguir que nuestro Linux se encargue de este menester.

[13] En una red NetBIOS existen los tipos (la tabla está tomada del libro *Usando Samba,* para ampliar sobre este tema a él os remitimos):

Papel	**Valor**
b-node	Usa registro *broadcast* y sólo resolución.
p-node	Usa registro punto-a-punto y sólo resolución.
m-node	Usa *broadcast* para registro. Si tiene éxito, notifica al servidor NBNS el resultado. Usa broadcast para resolución; usa servidor NBNS si el broadcast no tiene éxito.
h-node (hybrid)	Usa servidor NBNS para registro y resolución; usa *broadcast* si el servidor NBNS no responde o no está operativo.

```
# --- Selects point-to-point node (default is hybrid). Don't change this unless
# -- you understand Netbios very well
#    option netbios-node-type 2;
```

Rango de direcciones IP que el servidor puede asignar en esta subred[14]

```
range dynamic-bootp 192.168.0.128 192.168.0.255;
```

La IP se concede por ese tiempo en segundos, una vez consumido se tiene que renegociar la IP asignada

```
default-lease-time 21600;
```

Duración máxima para disponer de una IP (en segundos)

```
max-lease-time 43200;
```

Le asignamos a la máquina `ns` una IP fija y un nombre, se trata del interfaz de red con la dirección MAC especificada.

```
    # we want the nameserver to appear at a fixed address
    host ns {
       next-server marvin.redhat.com;
       hardware ethernet 12:34:56:78:AB:CD;
       option host-name "ns";
       fixed-address 207.175.42.254;
    }
}
```

Ejemplo

Una vez instalado, vamos a configurarlo partiendo de un ejemplo concreto. Vamos a considerar la situación de que disponemos de un router que nos da acceso a Internet y un Linux que va a ser el servidor DHCP de nuestra clase de informática (con equipos Windows XP[15] y Linux). Es decir, partimos de que disponemos de una red de clase C con las características:

- Red: 192.168.0.0/24
- IP router: 192.168.0.1
- IP Linux: 192.168.0.10
- Máscara de red: 255.255.255.0
- Rango de direcciones libres: 192.168.0.100 - 192.168.0.200
- IP del servidor de nombres: 80.58.0.33

Creemos con un editor el fichero `/etc/dhcpd.conf`[16]

```
$cat /etc/dhcpd.conf
ddns-update-style interim;
ignore client-updates;
subnet 192.168.0.0 netmask 255.255.255.0 {
     option routers 192.168.0.1;
     option subnet-mask 255.255.255.0;
```

[14] El servicio DHCP se configura a partir de la idea de ámbito: un ámbito es un rango de direcciones IP. Podemos tener un ámbito distinto para cada subred. Así, un ámbito debe contener completamente el rango de direcciones de una subred. Dentro de un ámbito, podremos reservar direcciones que están asignadas ya de forma estática (por ejemplo el propio servidor, el router de acceso a internet) y otras podemos asignarlas de forma dinámica.

[15] Con la familia 9x es similar

[16] Si en el Linux hemos configurado SAMBA podemos añadir

```
option netbios-name-servers 192.168.1.5;
```

```
    option domain-name-servers 80.58.0.33;
    range dynamic-bootp 192.168.0.100 192.168.0.200;
    default-lease-time 21600;
    max-lease-time 43200;
}
```

y pongamos el servidor en marcha:

Fedora

```
# service dhcpd start
```

Lo normal es que en Fedora, optemos porque el servicio se active en el arranque.

```
# /usr/sbin/ntsysv
```

Debian:

Para activarlo

```
#/etc/init.d/dhcp start
```

Y para que se inicie en los distintos niveles de ejecución[17]:

```
#update-rc.d dhcp defaults
```

➥ **Para practicar:**

Si disponemos de varias tarjetas de red en nuestra máquina Linux y es ella la encargada de dar servicio a Internet (usando el proxy-caché squid o las posibilidades de enrutar de los núcleos de Linux):

eth0 Servicio ADSL, si está en monupuesto será la IP pública. En caso contrario, una de la subred del router ADSL:

eth1 192.168.1.10 Red "administrativa"

eth2 192.168.2.10 Red de "alumnos"

y deseamos disponer de un servidor DCHP que no permita conexiones por la interfaz `eth0`, optaremos por:

1. Limitar el servicio a los interfaces `eth1` y `eth2`, para eso tendremos que modificar el fichero `/etc/sysconfig/dhcpd`[18], con la línea:

   ```
   DHCPDARGS="eth1 eth2"
   ```

   De esta forma impedimos que el servidor "escuche" en la interfaz de red que sale a Internet.

2. Ajustar el fichero `/etc/dhcpd.conf` a la nueva situación

   ```
   ddns-update-style interim;
   ignore client-updates;
   option domain-name-servers 80.58.0.33;
   subnet 192.168.1.0 netmask 255.255.255.0 {
   ```

[17] En general este comando se ejecutará de forma automática
[18] Sólo para Fedora, en Debian editar y ajustar el fichero `/etc/init.d/dhcp`

```
    #si usamos iptables
    #option routers 192.168.1.10;
    option subnet-mask 255.255.255.0;
    range dynamic-bootp 192.168.1.11 192.168.0.200;
    default-lease-time 21600;
    max-lease-time 43200;
}

subnet 192.168.2.0 netmask 255.255.255.0 {
    #si usamos iptables
    #option routers 192.168.2.10;
    option subnet-mask 255.255.255.0;
    range dynamic-bootp 192.168.2.100 192.168.2.199;
    default-lease-time 21600;
    max-lease-time 43200;
}
```

Ajustando los rangos de IPs a servir dinámicamente a nuestros intereses. Si además deseamos que la máquina con dirección MAC `00:A0:0C:13:5D:2D` tenga la IP fija 192.168.1.50 añadiríamos para la primera subred:

```
host ipfija{
       #Para asignarle también el nombre
       option host-name "ipfija.midominio.org";
       hardware ethernet 12:34:56:78:AB:CD;
       fixed-address 192.168.1.50;
}
```

Para que los cambios sean efectivos en Fedora usaremos[19]:

```
#service dhcpd reload
```

o paramos y rearrancamos[20]

```
#service dhpcd restart
```

Si junto a esta configuración usamos el Linux como enrutador, podemos conseguir que las máquinas obtengan una IP dinámica y que todas puedan salir a Internet usando el Linux (véase iptables en en la página 111)■

7.3.2. Configuración de los clientes:

Linux

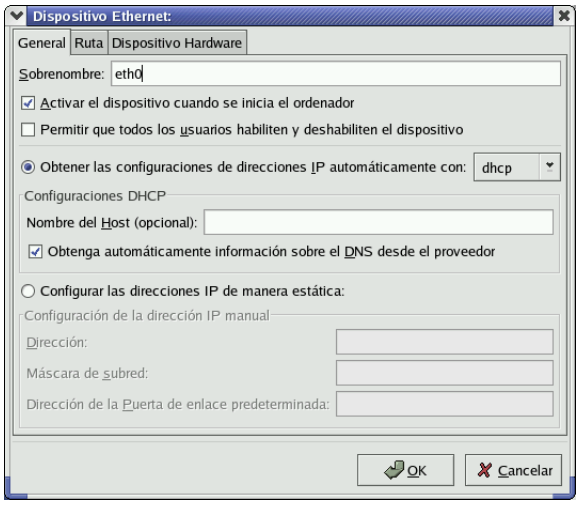
Fedora En modo gráfico no hay problema, ejecutaremos:

`#system-config-network`

y después de seleccionar **Dispositivos**→**Modificar** optaremos por

[19] `#/etc/init.d/dhcp restart`

[20] La opción restart suele ser conveniente, porque si está arrancado el servicio, primero lo para y luego lo arranca. En caso de no estar arrancado, la parada no hace nada y luego lo arranca.

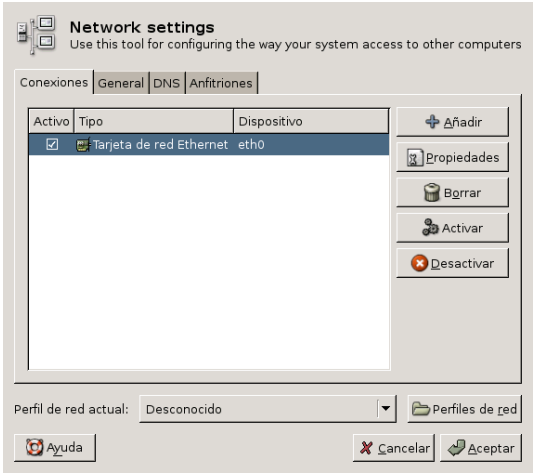


obtener las configuraciones de direcciones de IP automáticamemte con DHCP, obtener automáticamente información sobre DNS y, en su caso, rellenaremos el nombre de máquina que deseemos. Una vez guardados los cambios, habremos modificado el fichero como sigue (se puede hacer a mano, además la última línea sólo aparecerá si especificamos un nombre de host)

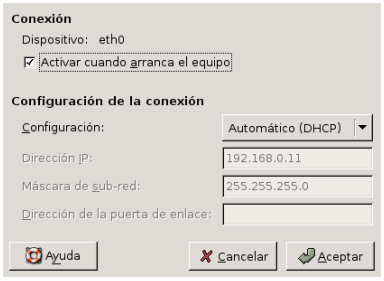
```
# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
USERCTL=no
PEERDNS=yes
TYPE=Ethernet
DHCP_HOSTNAME=cursolinux
```

Guadalinex 04

```
#network-admin
```



y tras optar por **Propiedades**, seleccionar que se inicie en el arranque usando DHCP



Los cambios realizados se traducen en el fichero

```
# cat /etc/network/interfaces
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
        name Tarjeta de red Ethernet
```

➥ **Para practicar:**

Actualizar la IP de la máquina cliente ejecutando:

```
# /sbin/dhclient
```

Podemos comprobar la IP asignada con

```
# /sbin/ifconfig
```

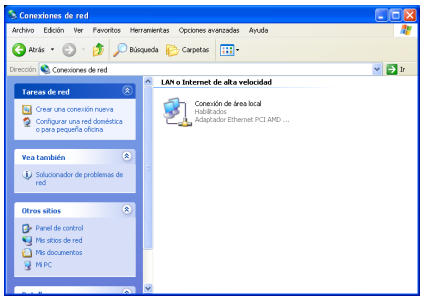
y la tabla de enrutamiento con:

```
# /bin/netstat -ar
```

■

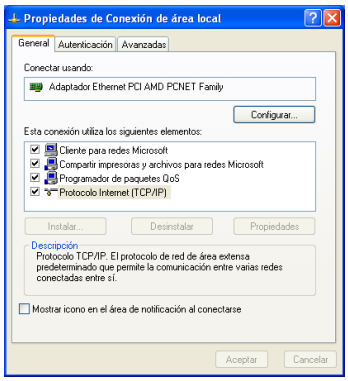
Windows

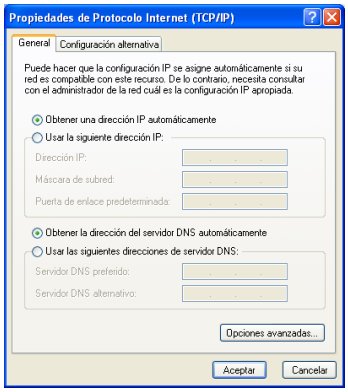
Para configurar la máquina Windows, optaremos por pulsar con el botón derecho[21] sobre **Conexiones de Red**



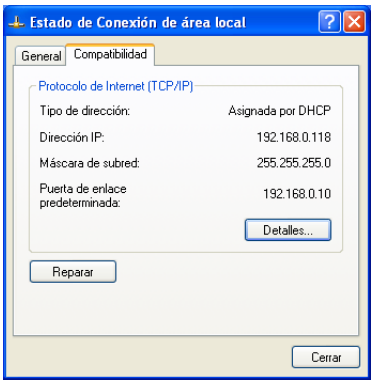
y en **Propiedades** optaremos por marcar las opciones de **Obtener la IP automáticamente** y pondremos como puerta de enlace la adecuada.

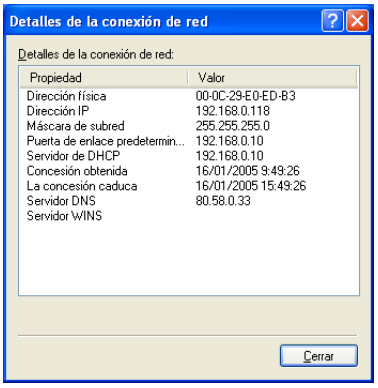
[21] Se parte de la idea de que es un Windows XP y que la red está configurada, al menos en cuanto a la cuestión *hardware*.





Podremos comprobar la configuración con el comando[22] `ipconfig` o en modo gráfico[23], por ejemplo, optando por **Estado** en la ventana emergente que aparece al pulsar con el botón derecho del ratón sobre **Conexiones de Red:**





[22] `winipcfg` en el 9x

[23] Los datos de la captura no se corresponden en su totalidad con los datos de ejemplo.

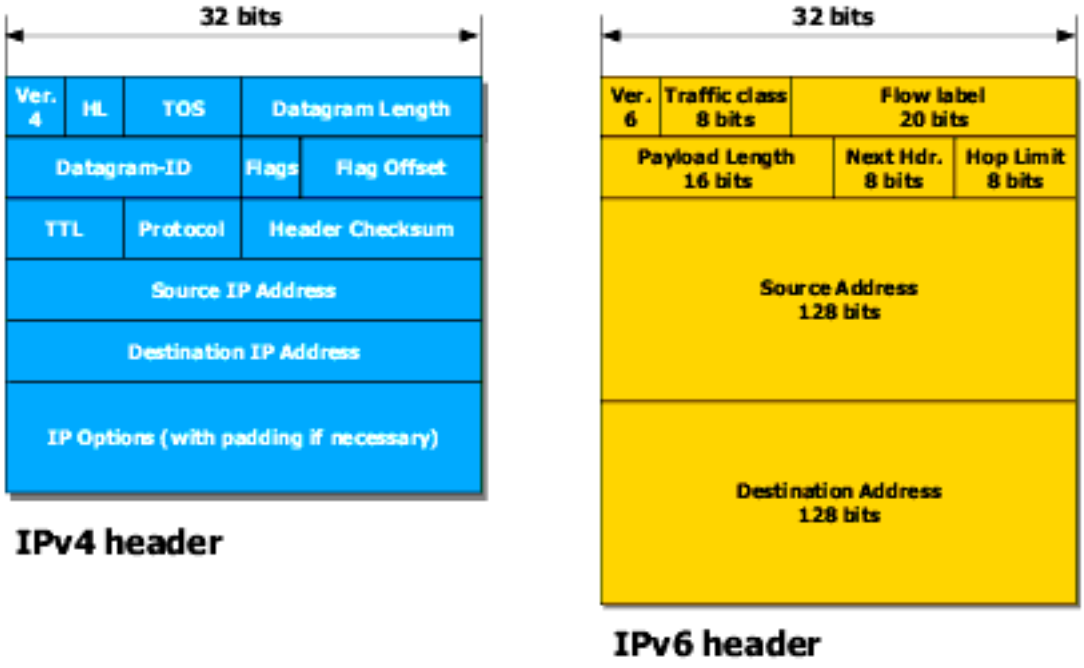
Capítulo 8

IPv6

8.1. Introducción histórica

IPv6 es una evolución del protocolo IPv4, conocido simplemente como protocolo IP. El protocolo IP fue diseñado hace tiempo (el RFC consta desde enero de 1980) y desde su concepción han surgido nuevas necesidades y el rango de direcciones se ha demostrado insuficiente. El cambio más sustancial que se realiza en IPv6 se centra en el rediseño de las cabeceras, incluyendo un aumento en el tamaño de las direcciones de 32 a 128 bits, obteniendo así un direccionamiento muchísimo mayor.

Figura 8.1: Comparación paquete IPv4 y paquete IPv6



Este aumento de direcciones está propiciado por la multitud de dispositivos que en el futuro pueden precisar de una dirección IP (frigoríficos, teléfonos móviles, ...).

La primera pregunta que surge es por qué se ha denominado IPv6 en lugar de IPv5, que

sería la continuación lógica de IPv4. La cabecera de un paquete IP tiene los 4 primeros dígitos reservados para indicar la versión del protocolo. Actualmente el 4 está reservado para IPv4 y el 5 está reservado para *Internet Stream Protocol v2*. Por consiguiente, el próximo número que queda libre para definir el protocolo es el 6, siendo éste el origen de IPv6.

La primera referencia que encontramos en el código fuente de linux fue en el kernel 2.1.8 en noviembre de 1996, siendo su autor Pedro Roque. Sin embargo, no fue hasta Octubre de 2000 cuando encontramos nuevas referencias a IPv6. Éstas estuvieron a cargo de un proyecto japonés, que se encargó de completar la especificación anterior, así como de actualizarla. Debido a su tamaño, en el kernel 2.4.* no se pudo añadir totalmente, por lo que hay algunos elementos que no están completamente definidos. En la versión del kernel 2.5.* se intentó introducir todas las extensiones disponibles, siendo ya en la 2.6.* donde aparece con toda su funcionalidad.

A modo de resumen, las características fundamentales de IPv6 serían:

- Mayor espacio de direcciones.
- "Plug & Play": Autoconfiguración de los interfaces de red.
- Seguridad intrínseca en el núcleo del protocolo (IPsec).
- Calidad de Servicio (QoS) y Clase de Servicio (CoS).
- *Multicast*: Envío de un mismo paquete a un grupo de receptores.
- *Anycast*: Envío de un paquete a un receptor dentro de un grupo.
- Paquetes IP eficientes y extensibles, sin que haya fragmentación en los *routers*, alineados a 64 bits (preparados para su procesado óptimo con los nuevos procesadores de 64 bits), y con una cabecera de longitud fija, más simple, que agiliza su procesado por parte del *router*.
- Posibilidad de paquetes con carga útil (datos) de más de 65.535 *bytes*.
- Encaminado (*routing*) más eficiente en el troncal (*backbone*) de la red, debido a una jerarquía de direccionamiento basada en la agregación.
- Renumeración y *multi-homing*, que facilita el cambio de proveedor de servicios.
- Características de movilidad.

8.1.1. ¿Cómo son las direcciones IPv6?

En IPv4 podemos asignar más de una IP a un interfaz de red si así lo requerimos (aliases, multicast). Manteniendo esta funcionalidad, IPv6 va más allá, permitiendo asignar tantas IP como queramos a una interfaz. El límite vendrá definido por la definición de la pila, con vista a evitar posibles ataques de denegación de servicio.

Las direcciones IPv6 son identificadores para interfaces y conjuntos de interfaces, clasificándose en tres tipos:

- *Unicast*: Identificador para una única interfaz. Un paquete enviado a una dirección unicast es entregado sólo a la interfaz identificada con dicha dirección. Es el equivalente a las direcciones IPv4 actuales.
- *Anycast*: Identificador para un conjunto de interfaces (normalmente pertenecen a diferentes nodos). Un paquete enviado a una dirección anycast es entregado en una (cualquiera) de las interfaces identificadas con dicha dirección (la más próxima, de acuerdo a las medidas de distancia del protocolo de encaminamiento). Permite que varias máquinas puedan ocuparse del mismo tráfico según una secuencia determinada (por el *routing*), si la primera "cae".

- *Multicast*: Identificador para un conjunto de interfaces (normalmente pertenecientes a diferentes nodos). Un paquete enviado a una dirección multicast es entregado a todas las interfaces identificadas por dicha dirección. La misión de este tipo de paquetes es evidente: aplicaciones de retransmisión múltiple (broadcast).

Vemos ahora más concretamente cómo es una dirección IPv6. Tal como se ha indicado anteriormente, las direciones IPv6 tienen una longitud de 128 bits, lo que en decimal equivale a un número de:

```
2^128-1: 340282366920938463463374607431768211455
```

Como puede observarse, estos números no son manejables. Una notación más simple sería en hexadecimal, partiendo de la representación binaria y agrupando los bits en grupos de 4:

```
2^128-1: 0xffffffffffffffffffffffffffffffff
```

Reducimos así la longitud de la dirección IPv6 a 32 caracteres. Aún así, esta representación es mejorable, ya que es fácil olvidar alguno de los caracteres. Se agrupan en bloques de 16 bits, separados por ':'

```
2^128-1: ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
```

De esta forma una dirección IPv6 podría ser la siguiente:

```
3ffe:ffff:0100:f101:0210:a4ff:fee3:9566
```

Simplificando aún más, los `0`'s iniciales de cada bloque se eliminan. De este modo, la dirección IPv6:

```
3ffe:ffff:0100:f101:0210:a4ff:fee3:9566
```

pasaría a ser

```
3ffe:ffff:100:f101:210:a4ff:fee3:9566
```

Otra simplificación más consiste en sustituir la secuencia de 16 bits compuesta por `0`'s por '`::`'. Esto se realizará una sola vez ya que en otro caso la representación no sería única.

```
3ffe:ffff:100:f101:0:0:0:1 -> 3ffe:ffff:100:f101::1
```

De esta forma, la mayor simplificación se produciría en la dirección `localhost` de IPv6:

```
0000:0000:0000:0000:0000:0000:0000:0001 -> ::1
```

8.1.2. Tipos de direcciones

Al igual que IPv4, las direcciones IPv6 pueden ser divididas en parte de red (64 bits superiores) y parte de host (64 bits inferiores), mediante el uso de máscaras de subred.

Direcciones con prefijos especiales

Dirección `localhost` Es la dirección para el interface de *loopback*, que en IPv4 era `127.0.0.1` y que en IPv6 es:

```
0000:0000:0000:0000:0000:0000:0000:0001
```

o en formato comprimido

```
::1
```

Los paquetes que tengan esta dirección como origen o destino nunca deben abandonar el *host*.

Direcciones sin especificar Es una dirección especial "cualquiera" que en IPv4 era `0.0.0.0` y que se utiliza en establecimiento de *sockets* y tablas de enrutamiento:

```
0000:0000:0000:0000:0000:0000:0000:0000
```

o en formato comprimido

```
::
```

Direcciones IPv6 con direcciones IPv4 embebidas Existen dos tipos de direcciones IPv6 que albergan direcciones IPv4 en su interior.

Las direcciones IPv4 mapeadas a IPv6 se utilizan en la creación de *sockets* por parte de demonios IPv6 que se conectarán a direcciones IPv4. Estas direcciones son definidas con un prefijo especial de longitud 96 bits, siendo `a.b.c.d` la dirección IPv4:

```
0:0:0:0:0:ffff:a.b.c.d/96
```

o en formato comprimido

```
::ffff:a.b.c.d/96
```

Así, la dirección IPv4 `192.168.0.32` sería `::ffff:192.168.0.32`

El otro tipo son las direcciones IPv4 compatibles con IPv6, utilizadas para el *tunneling* automático:

```
0:0:0:0:0:0:a.b.c.d/96
```

o en formato comprimido

```
::a.b.c.d/96
```

Prefijos que definen la red

Veamos a continuación los diferentes tipos de prefijos, que definirán tipos de direcciones.

Link local address Son direcciones especiales que son sólo válidas en el enlace de una interfaz. Utilizando esta dirección como destino, el paquete nunca pasará a través de un *router*.

Las direcciones locales de enlace han sido diseñadas para direccionar un único enlace para propósitos de auto-configuración (mediante identificadores de interfaz), descubrimiento del vecindario, o situaciones en las que no hay *routers*. Por tanto, los encaminadores no pueden retransmitir ningún paquete con direcciones fuente o destino que sean locales de enlace (su ámbito está limitado a la red local).

Tienen el siguiente formato:

```
fe8x: la única que se utiliza
fe9x:
feax:
febx:
```

donde `x` es un carácter hexadecimal, normalmente `0`.

Una dirección con este prefijo es encontrada en cada interfaz IPv6-*enabled* después de un estado de autoconfiguración.

Site local address Estas direcciones son similares a las direcciones de IPv4 utilizadas para redes privadas, aunque en este caso tienen la ventaja de que cualquiera que use este tipo de dirección puede usar los 16 bits para un máximo de 65.536 subredes. Es comparable a la subred `10.0.0.0/8` de IPv4.

Las direcciones locales de sitio permiten direccionar dentro de un *local site* u organización sin la necesidad de un prefijo global. Se configuran mediante un identificador de subred de 16 bits. Los encaminadores no deben retransmitir fuera del sitio ningún paquete cuya dirección fuente o destino sea *local site* (su ámbito está limitado a la red local o de la organización).

Otra ventaja añadida proviene de la facultad de asignar más de una direción IPv6 a un dispositivo IPv6, por lo que podría asignarse una IP global y una local.

Este tipo de direcciones tienen la siguiente estructura:

```
fecx: la más usada
fedx:
feex:
fefx:
```

donde `x` es un carácter hexadecimal, normalmente `0`.

A pesar que hay quien considera útiles este tipo de direcciones para su uso en laboratorios de pruebas, existe también la idea de que es mejor no hacer uso de ellas[1].

Tipo de dirección global unicast globales "agregables" Hay un tipo de dirección global definido (el primer diseño, llamado "basado en proveedor", se desechó hace varios años, quedando resquicios de la misma en algunos kernel obsoletos de linux). Su estructura es:

```
2xxx:
3xxx:
```

donde `x` son caracteres hexadecimales.

Direcciones Multicast Las direcciones multicast son utilizadas para servicios del mismo tipo. Siempre comienzan con `ffxy`, donde `xy` determinará un alcance distinto. El alcance multicast es un parámetro que indica la máxima distancia que puede viajar un paquete multicast desde la entidad que lo envió:

```
ffx1: node-local, los paquetes nunca abandonan el nodo.
ffx2: link-local, los paquete nunca son reenviados por rou-
ters, por lo que nunca abandonan el enlace especificado.
ffx5: site-local, los paquetes nunca abandonan el sitio.
ffx8: organization-local, los paquetes nunca abandonan la organización.
ffxe: global scope.
```

el resto de valores que pueden utilizarse están reservados.

Direcciones anycast Las direcciones anycast son un tipo especial de dirección utilizada para cubrir aspectos como el servidor DNS más cercano, el servidor DHCP más cercano, o grupos dinámicos similares.

Las direcciones se toman del espacio de direcciones unicast. El mecanismo anycast, desde el punto de vista del cliente, serán manejados por protocolos de *routing* dinámicos[2].

Un ejemplo de dirección anycast es la dirección anycast *subnet-router*. Asumiendo que un nodo tiene la siguiente dirección global IPv6:

```
3ffe:ffff:100:f101:210:a4ff:fee3:9566/64
```

[1] En inglés se denomina *deprecated*.

[2] Las direcciones anycast no pueden usarse como direcciones de origen, únicamente como direcciones destino.

La dirección anycast *subnet-router* se creará eliminando el subfijo (los 64 bits menos significativos) completamente:

```
3ffe:ffff:100:f101::/64
```

Tipos de direciones de nodos

Para aspectos de autoconfiguración y movilidad, se ha decidio usar los 64 bits más bajos como la parte de nodo de una dirección en la mayoría de los tipos de las direcciones. Aún así, cada subred puede albergar un gran número de direcciones.

Automáticas Con la autoconfiguración, la parte de host de la dirección es procesada convirtiendo la dirección MAC del interfaz (si está disponible), con el método EUI-64, a una dirección única IPv6. Si no hay dirección MAC disponible para este dispositivo (ocurre en dispositivos virtuales), algo distinto (dirección IPv4 o la dirección MAC del interfaz físico) es usado en su lugar.

Consideremos de nuevo el ejemplo anterior:

```
3ffe:ffff:100:f101:210:a4ff:fee3:9566
```

aquí, `210:a4ff:fee3:9566` es la parte de *host* y el resto es obtenido de la MAC `00:10:A4:E3:95:66` usando el IEEE-Tutorial EUI-64 (`http://standards.ieee.org/regauth/oui/tutorials/EUI64.html`), diseñado para identificadores EUI-48.

Manuales Para los servidores es probablemente más sencillo recordar una única dirección. Es posible asignar una dirección IPv6 adicional a un interfaz:

```
3ffe:ffff:100:f101::1
```

Para los sufijos como "`::1`" se requiere que el séptimo bit más significativo sea establecido a `0`. También se reservan otras combinaciones de bits para direcciones anycast.

8.1.3. ¿Estamos preparados para IPv6?

Antes de comenzara trabajar con IPv6 es preciso comprobar que el sistema sobre el que se trabajará está preparado para IPv6.

Soporte IPv6 en el kernel

La mayoría de las distribuciones modernas de Linux tiene el kernel preparado para soportar las funcionalidades IPv6, normalmente en forma de módulos. Estos módulos no siempre son cargados de forma automática.

La primera comprobación que se hará es comprobar que existe `/proc/net/if_inet6`. Su existencia nos indica que el módulo correspondiente está cargado. En caso de no encontrar esta entrada en `/proc` cargaremos el módulo de forma manual:

```
modprobe ipv6
```

En caso de éxito, podemos comprobar la correcta carga del módulo:

```
lsmod | grep ipv6
```

Es importante recordar que la descarga de este módulo del kernel no está soportado, pudiendo dar lugar a un error que deje nuestro sistema bloqueado.

Es posible cargar de forma automática el módulo IPv6, únicamente hay que añadir la línea siguiente a la configuración del cargador de módulos del kernel en `/etc/modules.conf`:

```
alias net-pf-10 ipv6
```

También es posible deshabilitar de forma automática la carga de dicho módulo, cambiando la línea anterior por:

```
alias net-pf-10 off
```

En el caso de que las comprobaciones anteriores hayan dado un resultado negativo, se pueden barajar las siguientes opciones:

- Actualizar la distribución a una que implemente una versión del kernel con soporte IPv6
- Compilar un nuevo kernel a partir de los fuentes

La primera opción es la recomendable, especialmente si no estamos habituados a las labores de recompilar el kernel.

Dispositivos de red con soporte IPv6

No todos los dispositivos de red están preparados para dar soporte a paquetes IPv6. Uno de los puntos que hay que tener en cuenta es que debido a la estructura de la capa de red que implementa el kernel, un paquete IPv6 no se reconoce por su número de cabecera IP (6 en lugar de 4). Se reconoce por el número de protocolo de la capa 2 del protocolo de transporte. Así, cualquier protocolo de transporte que no utilice este número de protocolo no puede despachar paquetes IPv6[3].

Utilidades de configuración dispositivos IPv6

net-tools El paquete `net-tools` incluye utilidades como `ifconfig` y `route` que ayudarán a configurar IPv6 en el interface.

Guadalinex

```
apt-get install net-tools
```

Fedora

```
rpm -Uvh net-tools-1.60-37.i386.rpm
```

iproute Este conjunto de herramientas permiten configurar la red a través de la utilidad ip. Proporciona más funcionalidades que el paquete `net-tools` pero no está tan documentado como éste.

Guadalinex

```
#apt-get install iproute
```

Fedora

```
#rpm -Uvh iproute-2.6.9-3.i386.rpm
```

Utilidades de chequeo IPv6

Una vez configurado el interfaz de red para tener soporte IPv6 es necesario tener acceso a las utilidades que nos permitan comprobar que nuestros paquetes IPv6 llegan a su destino.

[3]El paquete se transmitirá sobre el enlace, pero en la parte de recepción la entrega no funcionará.

ping IPv6 Normalmente este programa está incluido en el paquete iputils.

Guadalinex

```
#apt-get install iputils-ping
```

Fedora

```
#rpm -Uvh iputils-20020927-16.i386.rpm
```

Se utiliza para enviar ICMPv6 *echo request* que sirvan de comprobación.

```
root@guadalinex:~# ping6 -c 1 ::1
PING ::1(::1) 56 data bytes 64 bytes from ::1: icmp_seq=1 ttl=64 ti-
me=12.2 ms
--- ::1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, ti-
me 1ms rtt min/avg/max/mdev = 12.281/12.281/12.281/0.000 ms
```

Puede que si usuarios que no tengan permisos de root intentan ejecutarlo se encuentren con problemas. Esto puede ser debido a:

- `ping6` no está en la ruta del usuario
- `ping6` no se ejecuta adecuadamente debido a los permisos, será necesario ejecutar

  ```
  chmod u+s /usr/sbin/ping6
  ```

Usando direcciones de enlace local para un ping IPv6, el kernel no sabe a través de qué dispositivo (físico o virtual) debe enviar el paquete[4].

```
root@guadalinex:~# ping6 fe80::212:34ff:fe12:3456 connect: Invalid ar-
gument
```

En este caso sería necesario especificar el interface que vamos a utilizar:

```
root@guadalinex:~# ping6 -I eth0 -c 1 fe80::2e0:18ff:fe90:9205
PING  fe80::212:23ff:fe12:3456(fe80::212:23ff:fe12:3456) from
¬ fe80::212:34ff:fe12:3478 eth0: 56 data bytes
64 bytes from fe80::212:23ff:fe12:3456: icmp_seq=0 hops=64 ti-
me=445 usec --- fe80::2e0:18ff:fe90:9205 ping statistics --- 1 pac-
kets transmitted, 1 packets received, 0% packet loss round-
trip ¬ min/avg/max/mdev = 0.445/0.445/0.445/0.000 ms
```

En el caso de realizar ping sobre direcciones multicast IPv6, podemos encontrar los nodos con direcciones IPv6 activos haciendo `ping6` sobre la dirección multicast del enlace local *all-node*.

```
root@guadalinex:~# ping6 -
I eth0 ff02::1 PING ff02::1(ff02::1) from fe80:::2ab:cdff:feef:0123 eth0: 56 da-
ta bytes
64 bytes from ::1: icmp_seq=1 ttl=64 time=0.104 ms
64 bytes from fe80::212:34ff:fe12:3450: icmp_seq=1 ttl=64 ti-
me=0.549 ms (DUP!)
```

A diferencia de IPv4, donde las respuestas a un ping de la dirección de broadcast pueden ser deshabilitadas, el comportamiento en IPv6 es que no puede deshabilitarse excepto IPv6 *firewalling*.

[4]Cada dispositivo tiene una dirección de enlace local

traceroute/tracepath IPv6 Estas utilidades suelen estar incluidas en el paquete iputils.

Guadalinex

```
#apt-get install iputils-tracepath
```

Fedora

```
#rpm -Uvh iputils-20020927-16.i386.rpm
```

El comportamiento de `traceroute6` será similar al traceroute de IPv4, pero referido a direcciones IPv6. En el caso de `tracepath`, es igual que `traceroute`, pero va descubriendo las MTU en el camino al nodo destino.

tcpdump IPv6 Para obtener los datos que viajan por la red utilizaremos la misma herramienta que en IPv4, `tcpdump`. Esta herramienta tiene en las versiones actuales soporte para IPv6. Anteriormente se vio el uso de esta herramienta, así como de `ethereal`, por lo que nos remitimos a lo ya contado.

Clientes y servidores en IPv6

Algunos de los clientes que tienen soporte para IPv6[5] son:

- DNS
- Telnet
- OpenSSH
- Navegadores web[6]

Hay que tener en cuenta que si nuestro navegador web soporta IPv6, pero el acceso a internet lo realiza a través de un proxy IPv4 no accederemos a servidores `http` IPv6.

En el caso de programas servidores con soporte IPv6 habría que tener las mismas consideraciones que para los clientes, junto con una configuración adecuada. Los programas servidores BIND, xinetd, Apache2 son algunos de los que tienen soporte IPv6.

[5] Es posible que por defecto algunos no soporten IPv6, sería necesario recompilarlos con dicho soporte.

[6] Podemos comprobar fácilmente si nuestro navegador es IPv6, basta acceder a `http://www.kame.net`. Si tiene soporte IPv6 la tortuga se moverá, en otro caso permanecerá estática.

Prácticas

TIPO I

E1-I-1

Realizar correctamente todas las preguntas del cuestionario que para esta entrega se ha habilitado en Moodle:

1. Deseamos desintalar un paquete de nombre `sirvepa-to` y eliminar sus paquetes de configuración, para eso hemos de ejecutar:

   a) `apt-get --purge install sirvepa-to-`
   b) `apt-get remove sirvapa-to`
   c) `apt-get install sirvepa-to`
   d) `apt-get install sirvepa-to-`
   e) `apt-get clean sirvepa-to`

2. Respecto de la siguiente dirección IP, 150.214.4.255, con máscara de red 255.255.255.0 podemos decir que:

   a) Es inválida
   b) Puede corresponder a un host
   c) Es una dirección de broadcast
   d) Es de clase D

3. Si en nuestro sistema tenemos una dirección de red 192.168.1.130/255.255.255.192 y el gateway de salida es el 192.168.1.1.

   a) Saldremos a Internet por la dirección correspondiente al localhost
   b) Saldremos a Internet por la dirección 192.168.1.254
   c) El administrador de red debe darnos permisos adicionales para salir a Internet
   d) Debemos cambiar la máscara de red para llegar al gateway

4. La red 192.168.1.32/27 puede tener hasta

   a) 30 hosts
   b) 32 hosts
   c) 16 hosts
   d) Es inválida

5. Si queremos hacer una captura de paquetes (sin truncar) escuchando en el interface `eth1` y guardar dicha captura en el fichero `/tmp/micaptura.txt`. ¿Qué comando utilizaríamos?

a) `tcpdump -i eth1 -w /tmp/micaptura.txt`

b) `tcpdump -eth0 -s 1500 -w /tmp/micaptura.txt`

c) `tcpdump -i eth0 -s 1500 -w /tmp/micaptura.txt`

d) `tcpdump -i eth1 -s 1500 -w /tmp/micaptura.txt`

6. Si ejecutamos el comando `$netstat -nr` en nuestro sistema y obtenemos lo siguiente:

```
Kernel IP routing table
 Destination     Gateway         Genmask         Flags   MSS Win-
dow  irtt  Iface
 172.16.1.0      0.0.0.0         255.255.255.0    U         0 0          0   eth0
 192.168.1.0     0.0.0.0         255.255.255.128  U         0 0          0   eth1
 0.0.0.0        172.16.1.1        0.0.0.0         UG        0 0          0   eth0
```

¿Cuál de las siguientes afirmaciones es correcta?

a) Para llegar a la dirección 192.168.1.135 debemos utilizar el interfaz `eth0`

b) A la dirección 192.168.1.135 llegaremos a través de un gateway

c) No podremos llegar a la dirección 192.168.1.135

d) Para llegar a la dirección 192.168.1.135 debemos utilizar el interfaz `eth1`

7. Tenemos un servidor de red de IP 192.168.0.1 que hace de puerta de enlace a la subred 192.168.0.0/24 y deseamos configurar un servicio de DHCP para nuestra red. Qué línea es la adecuada para que los clientes usen la puerta de enlace al configurar sus interfaces de mediante el servicio de DHCP:

a) `option domain-name-servers 192.168.0.1;`

b) `option routers 192.168.0.1;`

c) `option ntp-servers 192.168.0.1;`

d) `option netbios-name-servers 192.168.0.1;`

8. Determina que afirmación es falsa para la siguiente regla de `iptables`:

```
iptables -A INPUT -i eth1 -s 0.0.0.0/0 -p TCP --dport www -j ACCEPT
```

a) Sólo se aplica a los paquetes que entran a nuestra máquina por el interfaz de red `eth1`

b) Permite que la dirección de origen del paquete sea cualquiera.

c) El protocolo al que se aplica es TCP

d) El puerto de origen de los paquetes es el puerto número 80.

9. ¿Cuál de las siguientes es una dirección IPv6 correcta?

a) `3ffe:fgff:0100:f101:0210:a4ff:fee3:9566`

b) `3ffe:ffff:0100:f101:0210:a4ff:fee3`

c) `3ffe:ffff:100:f101:210:a4ff:fee3:9566`

d) `3ffe:ffff:100:f101:210:a4ff:fee3::9566`

10. La dirección localhost en IPv6 es:

a) `0000:0000:0000:0000:0000:0000:0000:0001`

b) `ffff:ffff:ffff:ffff:ffff:ffff:ffff:fff1`

c) `ffff:ffff:ffff:ffff:ffff:ffff:ffff:0001`

d) `::1`

E1-I-2

- El comando `nmap` es una herramienta de exploración de redes y escáner de seguridad. Mediante ella, podemos ver los puertos que se encuentran abiertos en un sistema, observándolos desde el exterior, es decir, intentando la conexión y comprobando la respuesta. Solamente deberemos ejecutarlo sobre nuestra máquina. Si estamos en una red de una empresa u organismo, puede que la política de seguridad no permita que ejecutemos este comando sobre la red, considerándola un ataque a la seguridad.
- Otro comando, `netstat`, nos permite ver los puertos que están esperando conexiones en nuestra máquina y las conexiones que están establecidas, pero desde dentro de nuestra máquina.

La práctica consiste en lo siguiente:

1. Realizar una conexión ssh a nuestro servidor (`$ssh localhost`). Si no está instalado el paquete de servidor `ssh`, deberemos instalarlo[7].

   Fedora: debería estar instalado y en ejecución. En el caso de que no sea así lo instalaremos con:

   ```
   # apt-get install openssh-server
   ```

   Y para reactivarlo

   ```
   #/etc/init.d/sshd restart
   ```

   Guadalinex: se instala por defecto, sólo hemos de ejecutar

   ```
   #dpkg-reconfigure ssh
   ```

   y marcar las opción deshabilitada de: **Ejecutar el servidor sshd**.

2. Ejecutar los comandos `nmap -sTU localhost` y `netstat -atu`.
3. Enviar el resultado de su ejecución, comentando el resultado de ambos y sus coincidencias y/o diferencias. Comentar todas las conexiones que aparecen como resultado del comando `netstat`.

El resultado de la ejecución de los comandos y las explicaciones deben estar en un fichero de nombre `E1-I-2.txt`.

Tipo II

E1-II-1

Dado el fichero que podéis bajar del servidor, `captura.dump`, correspondiente a una captura de tráfico en una red, identificar lo siguiente.

1. Una sesión telnet, indicando usuario que ha entrado en el sistema, password y comandos ejecutados.
2. Una sesión de recogida de correo electrónico (POP), viendo el contenido del correo electrónico recuperado.
3. Una sesión ssh, indicando si es posible identificar el usuario que ha entrado en el sistema, direcciones MAC de origen y destino, direcciones IP de origen y destino y puertos de origen y destino.

El resultado de la práctica debéis mandarlo en un fichero de nombre `E1-II-1.txt`

[7] En entregas posteriores estudiaremos mejor este servicio de red.

E1-II-2

Instalar un firewall personal en vuestra máquina. La política a implementar será la siguiente:

- Permitir todos los accesos desde las máquinas de nuestra red local.
- Para el resto, solamente permitir conexiones a los servicios http, https y smtp.

El resultado de la implantación de las políticas y las explicaciones debe mandarse en formato OpenOffice un fichero de nombre `E1-II-2.sxw`.

Bibliografía

[1] Linux IPv6 HOWTO. Peter Bieringer. http://www.bieringer.de/linux/IPv6/

[2] Ethereal User's guide. http://www.ethereal.com/

[3] Guía de Administración de Redes con Linux. Olaf Kirch y Terry Dawson. Proyecto LuCAS, traducción al español. http://es.tldp.org/Manuales-LuCAS/GARL2/garl2

[4] Introduction to Linux A Hands on Guide. Machtelt Garrels. http://tille.soti.org/training/tldp

[5] Introducción a la Administración de una Red Local basada en Internet. Charles L. Hedrick. http://es.tldp.org/Manuales-LuCAS/IAR/intro-admon-redes-v1.1.html

[6] Los HOWTO dedicados a las redes: http://www.tldp.org/HOWTO/HOWTO-INDEX/networking.html#NETGENERAL

[7] Rute User's Tutorial and Exposition. Paul Sheer. http://www.icon.co.za/~psheer/book/rute.html.gz

[8] Modelo de referencia OSI. Eduardo T. Sánchez Badillo http://www.geocities.com/txmetsb/el_modelo_de_referencia_osi.htm

[9] Tutorial y descripción técnica de TCP/IP. http://ditec.um.es/laso/docs/tut-tcpip

[10] Telecomunicaciones. http://www.eveliux.com/fundatel/menu_telecom.html

[11] Sitio web oficial de IPtables. http://www.netfilter.org/

[12] TCP/IP Illustrated, Volume 1 The Protocols. W. Richard Stevens. http://av.stanford.edu/books/tcpip/

[13] Enabling Technologies for E-Commerce. http://penguin.dcs.bbk.ac.uk/academic/technology/

[14] Protocolos TCP/IP. Juan Salvador Miravet Bonet. http://www4.uji.es/~al019803/Tcpip.htm

[15] IPTABLES: Manual práctico. Pello Xabier Altadill Izura. http://es.tldp.org/Manuales-LuCAS/doc-iptables-firewall

[16] YoLinux Tutorial - Linux Networking. http://www.yolinux.com/TUTORIALS/LinuxTutorialNetworking.html

[17] Iptables. Guía rápida. http://www.beeeeee.net/nautopia/linux/iptables.htm

[18] Guía de Administración de Redes con Linux. Olaf Kirch y Terry Dawson. Proyecto LuCAS, traducción al español. http://es.tldp.org/Manuales-LuCAS/GARL2/garl2

[19] Introduction to Linux A Hands on Guide. MACHTELT GARRELS. http://tille.soti.org/training/tldp

[20] Introducción a la Administración de una Red Local basada en Internet. CHARLES L. HEDRICK. http://es.tldp.org/Manuales-LuCAS/IAR/intro-admon-redes-v1.1.html

[21] Los HOWTO dedicados a las redes:
http://www.tldp.org/HOWTO/HOWTO-INDEX/networking.html#NETGENERAL

[22] Rute User's Tutorial and Exposition. PAUL SHEER.
http://www.icon.co.za/~psheer/book/rute.html.gz

[23] Modelo de referencia OSI. EDUARDO T. SÁNCHEZ BADILLO http://www.geocities.com/txmetsb/el_modelo_de_referencia_osi.htm

[24] Tutorial y descripción técnica de TCP/IP. http://ditec.um.es/laso/docs/tut-tcpip

[25] Telecomunicaciones. http://www.eveliux.com/fundatel/menu_telecom.html

[26] Sitio web oficial de IPtables. http://www.netfilter.org/

[27] TCP/IP Illustrated, Volume 1 The Protocols. W. RICHARD STEVENS. http://av.stanford.edu/books/tcpip/

[28] Enabling Technologies for E-Commerce. http://penguin.dcs.bbk.ac.uk/academic/technology/

[29] Protocolos TCP/IP. JUAN SALVADOR MIRAVET BONET. http://www4.uji.es/~al019803/Tcpip.htm

[30] IPTABLES: Manual práctico. PELLO XABIER ALTADILL IZURA. http://es.tldp.org/Manuales-LuCAS/doc-iptables-firewall

[31] YoLinux Tutorial - Linux Networking.
http://www.yolinux.com/TUTORIALS/LinuxTutorialNetworking.html

[32] Iptables. Guía rápida. http://www.beeeeee.net/nautopia/linux/iptables.htm

Parte II

Linux como servidor

Capítulo 9

Demonios y Superdemonios

> Par Dios, señor -replicó Sancho-, ya yo los he tocado; y este diablo que aquí anda tan solícito es rollizo de carnes, y tiene otra propiedad muy diferente de la que yo he oído decir que tienen los demonios; porque, según se dice, todos huelen a piedra azufre y a otros malos olores; pero éste huele a ámbar de media legua.(*El ingenioso hidalgo Don Quijote de la Mancha*. Miguel de Cervantes Saavedra).

Como vimos en la primera entrega, cuando se ejecuta un proceso servidor, éste abre un socket y permanece escuchando en un puerto de nuestra máquina Linux, en espera de que se conecten los clientes. Estos procesos reciben el nombre de *demonios*[1]. Muchos procesos se ejecutan de esta forma, como *apache* o *sendmail*. Normalmente suelen ser procesos servidores que por su importancia, merecen un trato diferenciado. Éstos tienen sus propios ficheros de configuración y sus medidas de seguridad ya incluidas.

Algunos de estos demonios funcionan de forma diferente. Hay procesos servidores que atienden ellos mismos a todos los requerimientos de los clientes, y otros procesos servidores que son más comodones y lanzan un proceso hijo para que atienda las peticiones de los clientes y ellos seguir tranquilamente "durmiendo"[2]. Ejemplo del primer caso puede ser el demonio nscd[3], que hace él solo la tarea y el servidor httpd (apache), ejemplo del segundo (lanza varios hijos para servir las peticiones).

```
#ps aux
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
nscd 1870 0.0 0.6 56244 836 ? S 20:28 0:00 /usr/sbin/nscd
root 1886 0.0 0.4 2724 576 ? S 20:29 0:00 /usr/sbin/smartd
root 1901 0.0 0.5 4580 644 ? S 20:29 0:00 /usr/sbin/sshd
apache 2301 0.0 7.2 21468 9108 ? S 21:02 0:00 /usr/sbin/httpd
apache 2302 0.0 7.2 21468 9104 ? S 21:02 0:00 /usr/sbin/httpd
apache 2303 0.0 7.2 21468 9104 ? S 21:02 0:00 /usr/sbin/httpd
apache 2304 0.0 7.2 21468 9104 ? S 21:02 0:00 /usr/sbin/httpd
apache 2305 0.0 7.2 21468 9104 ? S 21:02 0:00 /usr/sbin/httpd
apache 2306 0.0 7.2 21468 9104 ? S 21:02 0:00 /usr/sbin/httpd
apache 2307 0.0 7.2 21468 9104 ? S 21:02 0:00 /usr/sbin/httpd
apache 2308 0.0 7.2 21468 9104 ? S 21:02 0:00 /usr/sbin/httpd
```

Sin embargo, si todos los procesos se ejecutaran así, nuestro sistema se volvería inmanejable y el consumo de recursos por parte de estos procesos sería muy elevado. Cientos de demonios

[1] En el contexto de los sistemas Unix, demonios son procesos que trabajan en segundo plano atendiendo a varias tareas sin intervención humana. El símbolo del demonio de la rama Unix BSD viene de ahí. Ver: `http://www.freebsd.org/es/copyright/daemon.html`

[2] Suele ser más eficiente para procesos con mucha carga.

[3] Fijaos que suelen terminar en la letra "d", de demonio. Éste es el *Name Server Caché Daemon*, demonio que realiza una caché de las peticiones de resolución de nombres para mejorar los tiempos de respuesta.

escuchando en los puertos de nuestra máquina y sin control. Espeluznante. Aquí es donde el superdemonio[4] *inetd* (o *xinetd* en sistemas más modernos) viene en nuestra ayuda.

Así aparecen dos modos de operación para los demonios de red. Éstos son:

autónomo (standalone) el programa demonio de red escucha en el puerto de red asignado y, cuando llega una conexión, se ocupa él mismo de dar el servicio de red (o bien uno de sus hijos). En este modo suele trabajar por ejemplo el servidor web (en nuestro caso Apache).

esclavo del servidor xinetd (o inetd) se trata de un superdemonio[5] (siempre están en ejecución) cuya finalidad es estar a la espera de que se produzca alguna solicitud de conexión del exterior. Si esto pasa, `xinetd` analiza esa solicitud determinando qué servicio le están solicitando y le pasa el control a dicho servicio.

9.1. inetd

Aunque más antiguo y con menos posibilidades que xinetd, el "anciano" inetd aún sigue utilizándose en algunos sistemas, entre ellos Guadalinex.

`/etc/inetd.conf` es el fichero de configuración para el demonio servidor `inetd`. Su función es la de almacenar la información relativa a lo que `inetd` debe hacer cuando recibe una petición de conexión a un servicio en particular. Para cada servicio que deseemos que acepte conexiones de red, debemos decirle a `inetd` qué demonio servidor ejecutar, y cómo ha de hacerlo.

Es un fichero de texto en el que cada línea describe un servicio. Cualquier texto en una línea que siga al carácter # es ignorado y se considera un comentario. Cada línea contiene siete campos separados por cualquier número de espacios en blanco (espacio o tabulador). El formato general es el siguiente:

`<servicio><tipo_socket><proto><flags><usuario><servidor><argumentos>`

servicio es el servicio correspondiente a esta configuración, tomado del fichero `/etc/services` y se corresponde con un número de puerto en el que escuchar.

tipo_socket describe el tipo de socket para esta entrada. Los valores permitidos son: `stream, dgram, raw, rdm` o `seqpacket`. Simplificando, los servicios basados en `tcp` usan `stream`, y casi todos los basados en `udp` usan `dgram`. Sólo algunos demonios servidores muy particulares usarán otros valores.

proto el protocolo considerado válido para este servicio. Debería corresponder con la entrada apropiada en el fichero `/etc/services` y suele ser `tcp` o `udp`. Los servidores basados en RPC (*Remote Procedure Call*) usarán `rpc/tcp` o `rpc/udp`.

flags sólo hay dos valores posibles: `wait` y `nowait`. Este campo le dice a `inetd` si el programa servidor de red libera el socket después de comenzar la ejecución, y si por tanto `inetd` podrá ejecutar otro servidor para la siguiente petición de conexión. Si no se libera, `inetd` deberá esperar y asumir que el demonio servidor que esté ejecutándose controlará las nuevas peticiones de conexión. Por norma general todos los servidores `tcp` deberían tener esta entrada con el valor `nowait` y la mayoría de servidores `udp` deberían tener `wait`.

usuario este campo indica qué cuenta de usuario de `/etc/passwd` será asignada para ejecutar el servidor cuando se lance. Esto es a menudo útil si quiere protegerse ante riesgos de seguridad. Puedes asignar el usuario nobody a una entrada, por lo que si la seguridad del servidor de red se ve comprometida, el posible daño queda minimizado. Habitualmente, sin embargo, este campo está asignado a root, porque muchos servidores requieren privilegios de administrador para funcionar correctamente.

[4] También llamado Superservidor

[5] Red Hat/Fedora utiliza el sistema `xinetd`, que es una mejora del servidor `inetd` (utilizado aún por Guadalinex).

servidor este campo es el camino completo hasta el programa servidor a ejecutar para esta entrada.

argumentos este campo comprende el resto de la línea de órdenes y es opcional. Es en donde se pone cualquier argumento de línea de comandos que se desee pasar al programa demonio servidor cuando es ejecutado.

↬ Un ejemplo de una línea del fichero `/etc/inetd.conf`

```
telnet stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.telnetd
```

La línea anterior significa que para el servicio telnet (puerto 23, que recogerá del fichero `/etc/services`) habrá un servicio `tcp` de tipo stream. El modo es `nowait`, que quiere decir que el demonio `inetd` escucha en el puerto 23. Cuando tiene una petición, la atiende generando un servicio de telnet, pero tras atenderla con un proceso hijo, se pone otra vez a escuchar, sin esperar a que la conexión que ha lanzado anteriormente haya terminado.

El proceso `/usr/sbin/tcpd` realizará un control de seguridad[6] y si es un acceso permitido, lanzará el servidor `/usr/sbin/in.telnetd`. Si no queremos que nuestra máquina proporcione un servicio determinado, debemos comentar la línea en este fichero, precediéndola del carácter `#`. Para que se active, podemos reiniciar la máquina, pero no es estrictamente necesario. Otras formas son:

1. Decirle que vuelva a cargar el fichero de configuración: `/etc/init.d/inetd reload`[7]

2. Localizar el número de proceso del servidor `inetd`.

   ```
   root@guadalinex:~# ps aux|grep inetd

   root 2656 0.0 0.1 2264 204 ? Ss 17:26 0:00 /usr/sbin/inetd

   root 8017 0.0 0.5 2392 704 pts/1 R+ 21:05 0:00 grep inetd
   ```

   Vemos que el número de proceso es el 2656. La línea con el grep es resultado de nuestro comando anterior.

   ```
   #kill -HUP 2656
   ```

   Mandamos la señal -HUP al proceso `inetd`. Al recibirla, este proceso sabe que tiene que volver a leer su fichero de configuración (`/etc/inetd.conf`) y actualizar su modo de funcionamiento ante los cambios.

Por defecto, en Guadalinex inetd viene deshabilitado. Para habilitarlo, lanzamos **Aplicaciones**→ **Configuración** →**Servicios** (o desde la línea de comandos `#services-admin`) y marcar la casilla de activación.

[6] Más adelante hablaremos de Tcp-Wrappers.
[7] También podríamos haber reiniciado `/etc/init.d/inetd restart`

9.2. xinetd

El superdemonio xinetd aparece en los sistemas RedHat/Fedora. Al igual que inetd, es un proceso especial que se queda a la escucha de conexiones TCP en unos puertos determinados. Cuando viene una solicitud de conexión, realiza una serie de comprobaciones y ejecuta el proceso servidor correspondiente.

Su funcionamiento es el siguiente: xinetd al arrancar, lee sus ficheros de configuración, que básicamente le dicen en qué puertos tiene que escuchar y cuando recibe una petición de conexión a uno de esos puertos[8], qué servidor ejecuta y en qué condiciones debe ejecutarlo.

Lo más normal es que xinetd esté ya instalado en nuestro sistema, al ser un soporte básico de red. Su configuración se encuentra en un fichero principal (`/etc/xinetd.conf`) y un directorio (`etc/xinetd.d`) en donde se encuentran ficheros para cada uno de los servicios que controla.

Veamos el fichero `/etc/xinetd.conf`

```
# Simple configuration file for xinetd
#
# Some defaults, and include /etc/xinetd.d/
defaults
{
        instances               = 60
        log_type                = SYSLOG authpriv
        log_on_success          = HOST PID
        log_on_failure          = HOST RECORD
        cps                     = 25 30
}

includedir /etc/xinetd.d
```

Veamos qué nos dice el fichero. Dentro de la sección `defaults`, que se aplica a todos los servicios bajo el control de xinetd por defecto.

instances = 60 Se limita el número de conexiones simultáneas a cada servicio a un máximo de 60. Es una forma de evitar que alguien nos colapse la máquina intentando múltiples conexiones, o prevenir bloqueos por algún fallo.

log_type = SYSLOG authpriv Se utiliza el servicio de *syslog* en su categoría *authpriv* para guardar el registro de actividad (log[9]).

[8] Si es el 23, llama al servidor de `telnet`; si es al 22, llama al servidor de `ssh`. Por ejemplo, si la máquina remota solicita una transferencia de ficheros por el puerto 21, le pasará la solicitud a `wu-ftpd` (proceso del servidor de ftp)

[9] Se determina en `/etc/syslog.conf` con `authpriv`: `/var/log/secure`

log_on_success = HOST PID Si la conexión se lleva a cabo con éxito, guarda la máquina desde la que se realiza y el identificador de proceso.

log_on_failure = HOST Si la conexión no se realiza, guarda la máquina desde la que se realizó el intento de conexión.

cps= 25 30 Limita el número de conexiones por segundo que permite. En este caso, se establece a 25. El segundo valor (30) indica el número de segundos que se espera antes de continuar la actividad, en caso de que el valor de conexiones por segundo se haya sobrepasado.

includedir/etc/xinetd.d Incluye los servicios que se encuentran en el directorio `/etc/xinetd.d`

El formato para cada uno de los ficheros que permiten configurar los distintos servicios, es de la forma[10]:

```
service "nombre_servicio"
{
...
  atributo = valor
  serie_valores -= elimina_valor
  serie_valores += añade_valor
...
}
```

Lo normal es que sólo se use "`=`" para asignar una valor a un atributo. Si el atributo es una serie de valores podemos eliminar un elemento de la serie con "`-=`" o añadirlo con "`+=`". Normalmente son:

disable toma los valores "`yes`" o "`no`". Es donde se activa (disable=no) o desactiva (disable=yes) el servicio.

type toma los valores `RPC`, `INTERNAL` (servicio que ya provee el propio `xinetd` de forma interna, puede ser: `echo`, `time`, `daytime`, `chargen` y `discard`) o `UNLISTED` (no aparece en `/etc/services`)

id nombre unívoco para identificar este servicio

socket_type describe el tipo de socket que esta entrada considerará relevante. Los valores permitidos son: `stream`, `dgram`, `raw` o `seqpacket`. Por regla general casi todos los servicios basados en tcp usan `stream`, y casi todos los basados en udp usan `dgram`. Sólo algunos demonios servidores muy particulares usarán otros valores

protocol el protocolo considerado válido para este servicio obtenido a partir del fichero `/etc/protocols` (suele ser tcp o udp). Si no se especifica, se usa el protocolo por defecto para ese servicio.

wait puede ser `yes` o `no`. Con este atributo indicamos a `xinetd` si el programa servidor de red libera el socket después de comenzar la ejecución (`wait=no`), y si por tanto `xinetd` podrá ejecutar otro servidor para la siguiente petición de conexión o deberá esperar (`wait=yes`) y asumir que el demonio servidor que esté ejecutándose controlará las nuevas peticiones de conexión. Por norma general todos los servidores `tcp` deberían tener esta entrada con el valor `no` y la mayoría de servidores `udp` deberían tener `yes`.

user indica qué cuenta de usuario de `/etc/passwd` será asignada para ejecutar el demonio de red al activarse.

group por si queremos especificar el grupo con que se ejecuta el servicio. Tiene que tener una entrada en `/etc/group`.

[10] Para conocer todas las posibilidades

```
$ man xinetd.conf
```

instances número máximo de peticiones que este servicio puede administrar.

server camino completo hasta el programa servidor a ejecutar para esta entrada.

server_args argumentos pasados al servidor.

no_access IP de máquinas que no podrán acceder a este servicio.

Si vamos al directorio `/etc/xinetd.d`, veremos que existen varios ficheros. Veamos alguno de ellos:

```
[root@linux xinetd.d]# more telnet
# default: on
# description: The telnet server serves telnet sessions; it uses \
#       unencrypted username/password pairs for authentication.
service telnet
{
        disable         = no
        flags           = REUSE
        socket_type     = stream
        wait            = no
        user            = root
        server          = /usr/sbin/in.telnetd
        log_on_failure  += USERID
}
```

Veamos qué significan cada uno de los valores para el servicio telnet.

service telnet Nos indica a qué servicio (y puerto en el que escucha[11]) es aplicable lo siguiente.

disable = no Una forma un poco confusa de decir que está activado. Se le dice que *NO* está deshabilitado.

flags = REUSE Reutiliza el socket abierto para próximas conexiones

socket_type = stream El tipo de socket es stream (TCP)

wait = no No tiene porqué haber finalizado la ejecución anterior para lanzar otro servicio nuevo

user = root Usuario que ejecutará el proceso

server = /usr/sbin/in.telnetd El servidor que arranca será `/usr/sbin/in.telnetd`

log_on_failure += USERID En caso de fallo en la autenticación, registra el usuario que ha intentado entrar.

El del servicio `swat` (interfaz vía web para configuración de SAMBA, véase 14.3.3), será el siguiente.

```
# default: off
# description: SWAT is the Samba Web Admin Tool. Use swat \
#              to configure your Samba server. To use SWAT, \
#              connect to port 901 with your favorite web browser.
service swat
{
        disable = yes
        port    = 901
        socket_type     = stream
        wait    = no
        only_from = localhost
        user    = root
```

[11] En caso de no encontrar un puerto en alguna de las líneas siguientes, cogerá el que tenga este nombre del fichero `/etc/services`.

```
        server  = /usr/sbin/swat
        log_on_failure  += USERID
}
```

En este servicio vemos que por defecto está deshabilitado. En caso de querer activarlo, debemos poner `disable=no` y decirle a `xinetd` que vuelva a releer su configuración. En este caso ejecutando:

```
# /etc/init.d/xinetd reload
```

El control sobre los accesos que veremos después con el complemento `tcpwrappers`, puede ser realizado de otra forma aquí, mediante las opciones `only_from` (para indicar direcciones o nombres de host o dominios desde los que se puede acceder al servicio) y `no_access` (para excluir direcciones o nombres y desde los cuales no se podrá acceder). Por ejemplo, una entrada para el servicio telnet anterior modificada así, quedaría:

```
# default: on
# description: The telnet server serves telnet sessions; it uses \
#       unencrypted username/password pairs for authentication.
service telnet
{
        disable         = yes
        flags           = REUSE
        socket_type     = stream
        wait            = no
        user            = root
        server          = /usr/sbin/in.telnetd
        log_on_failure  += USERID
        only_from       = 172.26.0.0/16
        no_access       = 172.26.1.0/24
        access_time     = 08:30-14:45
}
```

permite el acceso sólo a las máquinas de la red 172.26.0.0/16 del atributo `only_from`, pero impide que se conecten las máquinas de la subred 172.26.1.0/24 y además limita la posibilidad de conexión vía `telnet` al horario establecido en el atributo `acces_time`. Pero que ... ¡si este servicio está desactivado!

9.3. Parando y arrancando demonios

Podemos diferenciar dos formas de arrancar los demonios, ya sean independientes o controlados por inetd/xinetd. La primera forma de hacerlo es configurando el sistema para que los active automáticamente al arrancar. Será la más normal para los servicios que ofrezcamos desde nuestro servidor. Otra forma es arrancarlos manualmente, recordando que deberemos arrancarlos cuando los necesitemos y pararlos cuando no nos hagan falta o vayamos a apagar el servidor.

Antes de entrar de lleno en cómo instalar y configurar determinados servicios de red, vamos a recordar las herramientas de que disponemos para poder activar o desactivar determinados servicios según los distintos niveles de ejecución.

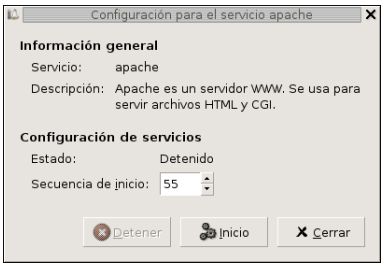
9.3.1. Debian

En Guadalinex contamos con la herramienta `services-admin`. Al ejecutarla, nos aparece la siguiente ventana.

El primer desplegable nos indica en qué nivel de ejecución del sistema será aplicable. Sus valores pueden ser Deteniendo el sistema (nivel 0), Modo gráfico (nivel 2), Modo texto (nivel 3) o Reiniciando el sistema (nivel 6).

Si seleccionamos el servicio, se ejecutará al pasar el sistema a dicho nivel de ejecución.

Las propiedades del servicio nos indican el nombre y descripción del servicio, el estado actual y en qué lugar se encuentra dentro del orden de arranque o parada de todos los servicios.



 # /usr/sbin/update-rc.d utilidad en línea de comandos para activar/desactivar servicios. En general es más fácil trabajar con la anterior.

Usando este comando podemos configurar los enlaces simbólicos de los directorios `/etc/rc?.d` y el script situado en `/etc/init.d/`. Si por ejemplo deseamos que el servicio de nombre `service` se ejecute en el arranque

1. Se pone en el directorio `/etc/init.d/`. En general los programas que instalemos y que sean necesarios en el arranque situan sus scripts de forma automática aquí.
2. Después creamos los enlaces simbólicos mediante el comando

   ```
   update-rc.d servicio defaults 35
   ```

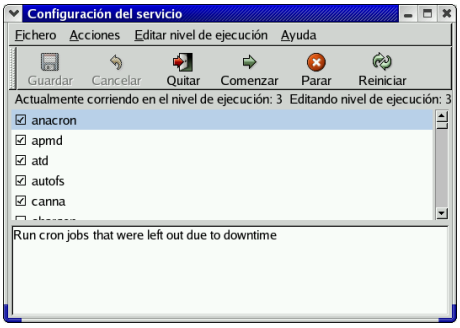
   Al pasarle el parámetro `defaults` forzamos a que lo cree para los niveles de ejecución que van del 2 al 5. Con el `35` obligamos a que service se arranque antes de cualquier script que contenga un número mayor de 36.

9.3.2. Fedora

Para los sistema Fedora disponemos de varias posibilidades.

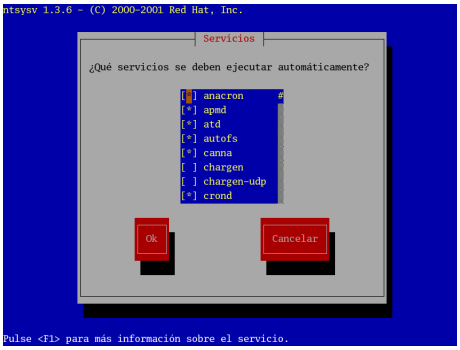
 # system-config-services (desde Gnome: →**Configuración del Sistema**→**Configuración de servidores**→**Servicios**). Utilidad gráfica que permite seleccionar qué servicios están en

activo para los niveles 3, 4 y 5. Permite reiniciarlos, pararlos y activarlos desde el propio programa.



Tambien se puede acceder al programa escribiendo el comando `serviceconf`.

 # /usr/sbin/ntsysv (se puede acceder a él desde el programa `setup`, opción **System services**). Los cambios no se activan en el momento. Además afecta sólo al nivel desde el que se ejecuta.[12]



 # /sbin/chkconfig utlidad en línea de comandos para activar o desactivar servicios. En general es más fácil trabajar con las dos anteriores.

Mediante estas interfaces podemos seleccionar los demonios que arrancarán automáticamente al iniciar el sistema. También se ocuparán de pararlos de forma ordenada cuando apaguemos el sistema.

Pero el haberlos activado aquí, no los activará en el momento, sino que deberemos esperar al próximo arranque del sistema.

Si lo que necesitamos es arrancarlos en el momento, distinguiremos los que se encuentran bajo el control de xinetd y los que son independientes.

Los que se encuentran bajo el control de xinetd podemos activarlos de la siguiente manera. Mediante el comando `setup` visto anteriormente, lo que hacemos es poner el valor de *disable=no* en su fichero de configuración[13]. Posteriormente, debemos reiniciar el servidor xinetd para que vuelva a leer los ficheros de configuración y perciba el cambio.

Mediante el comando

[12] Para conseguir que "afecte" a los niveles 3 y 5 por ejemplo, usar:
`# ntsysv --levels 35`

[13] También podríamos editar el fichero, pero es más cómodo así.

`#/etc/init.d/xinetd reload`[14]. O de forma equivalente, mediante
`#service xinetd restart.`

Respecto a los demonios independientes, podemos ver su fichero de arranque en el directorio `/etc/init.d` y arrancarlos mediante la llamada al script correspondiente. Por ejemplo
`#/etc/init.d/httpd start`[15]
o
`#service httpd start`

9.3.3. Algunos servicios de red usuales

autofs activa el proceso de administración de montaje automático de sistemas de ficheros o unidades (NFS, CD...)

dhcpd inicia un servidor DHCP local que permite asignar direcciones IP de forma dinámica.

httpd (o apache) activa el servidor web Apache

iptables reglas de cortafuegos del kernel

cupsys (o lpd) servidor de impresión.

mysqld para disponer del servidor de bases de datos MySQL

netfs activo permite montar sistemas de archivos de red: NFS, Samba y NetWare.

network para activar interfaces de red de nuestra máquina.

nfs activa servicios NFS

portmap este demonio administra conexiones a servicios basados en RPC.

postgresql activa el servidor de bases de datos postgresql.

sendmail activa el Agente de transporte de correo (MTA) sendmail

samba (o smb) activa el servicio Samba (para compartir archivos e impresoras con redes Windows)

squid permite disponer del proxy HTTP squid

sshd habilita servicios de red seguros (Secure SHell)

syslog demonio para registrar los log (o archivos de auditoría y trazas) del sistema

wu-ftpd activa los servicios ftp

xfs servidor de fuentes para las X

xinetd (o inetd) permite activar múltiples servicios de red

Si tenemos un servicio en nuestra máquina, con:

```
$/etc/init.d/servicio
```

podemos comprobar qué parámetros admite. Por ejemplo, con el servidor de impresión obtendríamos:

```
root@guadalinex:~# /etc/init.d/cupsys
Usage: /etc/init.d/cupsd {start|stop|restart|force-reload}
```

O sea, que si queremos pararlo sólo hay que ejecutar:

```
root@guadalinex:~# /etc/init.d/cupsys stop
Stopping printing system service: cupsd.
```

Y si queremos arrancarlo, hay que ejecutar:

```
root@guadalinex:~# /etc/init.d/cupsys start
Starting printing system service: cupsd.
```

[14] o restart si preferimos parar y arrancar.
[15] `/etc/init.d/apache` en Guadalinex

9.4. TCP-Wrappers

Además del cortafuegos personal con iptables y de las posibilidades de control y registro de `xinetd`[16], nuestra máquina Linux posee otra herramienta para defenderse de los ataques y no permitir nada más que los accesos autorizados.

El sistema TCP-Wrappers añade una capa adicional de protección a los servicios de red, en la que indicamos a qué máquinas permitimos el acceso y a cuáles no.

Unos de los servicios de red que incorporan protección mediante TCP-Wrappers son los superservidores inetd y xinetd. Para ver qué servicios de red tienen control de TCP-Wrappers, podemos ver si en su binario se encuentra la cadena `hosts_access`, indicando que se ha compilado incluyendo su soporte.

```
#strings /usr/sbin/sshd | grep hosts_access
@(#) hosts_access.c 1.21 97/02/12 02:13:22
```

El comando anterior mira si en el ejecutable `/usr/sbin/sshd` existe la cadena `hosts_access`, prueba de que incluye el soporte para tcp_wrappers.

El componente más importante dentro del paquete es la librería `/lib/libwrap.so`. Un servicio que incorpora control de acceso basado en TCP-Wrappers ha sido compilado con dicha librería.

Cuando se realiza un intento de conexión a un servicio con soporte de TCP-Wrappers, se comprueban los ficheros de control de acceso de TCP-Wrappers (`/etc/hosts.allow` y `/etc/hosts.deny`) para determinar si el cliente tiene permitida la conexión.

Si un cliente tras el control de dichos ficheros, tiene permitido el acceso, TCP-Wrappers pasa la conexión (socket) al servicio solicitado y no interfiere más con la comunicación entre el cliente y el servidor.

Además del control de acceso y registro, los TCP-Wrappers pueden activar comandos para interactuar con el cliente antes de negar o liberar el control de la conexión al servicio solicitado.

Los wrappers TCP ofrecen una serie de ventajas:

- Transparencia tanto para el cliente de la red como para el servicio de red controlado por TCP-Wrappers. El cliente que se está conectando no nota que está siendo controlado por TCP-Wrappers, excepto en el caso de que el acceso no se permita y se le cierre la conexión.
- Administración centralizada en los ficheros `/etc/hosts.allow` y `/etc/hosts.deny` del control de acceso para múltiples servicios de red.

Como hemos comentado, para determinar si una máquina cliente tiene permitido conectarse a un servicio, los TCP-Wrappers utilizan los archivos : `/etc/hosts.allow` y `/etc/hosts.deny` (conocidos como archivos de acceso a host, `host_access`).

Cuando una solicitud de un cliente es recibida por un servicio controlado por TCP-Wrappers, sigue los pasos siguientes:

1. TCP-Wrappers analiza secuencialmente el archivo `/etc/hosts.allow` y aplica la primera regla especificada para ese servicio. Si encuentra una regla que coincide, permite la conexión. Si no, se mira el archivo `/etc/hosts.deny`.
2. Se analiza secuencialmente el archivo `/etc/hosts.deny`. Si encuentra una regla que coincide, rechaza la conexión. Si no, se concede acceso al servicio.

Los siguientes son puntos importantes a considerar cuando se usen wrappers TCP para proteger servicios de red:

- Puesto que las reglas de acceso en `hosts.allow` son aplicadas primero, toman precedencia sobre las reglas en `hosts.deny`. Por lo tanto, si se permite el acceso a un servicio en `hosts.allow`, una regla negando el acceso al mismo servicio en `hosts.deny` es ignorada.

[16] No disponible en los sistemas Guadalinex

- Las reglas en cada archivo son leídas de arriba hacia abajo y la primera regla que coincida para un servicio dado es la única aplicada, siendo el orden de las reglas muy importante.
- Si no se encuentra ninguna regla para el servicio en ninguno de los archivos, o si no existe ninguno de los archivos, se concede el acceso al servicio.

9.4.1. Reglas de acceso

Los formatos para `/etc/hosts.allow` y `/etc/hosts.deny` son los mismos.
Las reglas se tienen que formatear de la siguiente manera:
`<demonios>: <clientes>[: <opcion>: <opcion>: ...]`
donde,

`<demonios>` es una lista separada por comas de los nombres de procesos (no de los nombres de servicios asociados a un número de puerto) o el comodín ALL para todos los servicios. Ejemplos de nombre de procesos demonios son: `ftpd`, `telnetd, sshd` o `fingerd`.

`<clientes>` es una lista separada por comas de nombres de host, direcciones IP o comodines, que identifica los hosts afectados por la regla. Por ejemplo: `ciencias.iesmurgi.org` para una máquina específica, `.juntadeandalucia.es` para cualquier nombre de máquina que acabe en esa cadena, `ó 80.32.` para cualquier dirección IP que comience con esos dígitos.

`<opcion>` es una acción o una lista separada con puntos y comas de acciones a realizar cuando la regla es activada. Podría, por ejemplo, ejecutar una instrucción que intentase identificar quién está autenticado en el host que se conecta, o generar un mensaje de correo u otro tipo de alerta a un administrador de sistema avisando de que alguien intenta conectar.

Hay cierto número de expansiones que podemos incluir, ejemplos comunes son:

%h se expande al nombre de la máquina que se conecta o a su dirección si no tiene un nombre,

%d es el demonio que está siendo llamado.

↪ Por ejemplo:

```
sshd : .cica.es
```

Esta regla le dice a TCP-Wrappers que controle las conexiones al demonio SSH (sshd) y las aplique a cualquier máquina del dominio `.cica.es`.

Si esta regla aparece en `hosts.allow`, la conexión será aceptada y si aparece en `hosts.deny`, la conexión será rechazada (si antes no ha sido aceptada en `/etc/hosts.allow`).

Este otro ejemplo de regla de acceso es más compleja y utiliza dos campos de opciones:

```
sshd : .cica.es : spawn /bin/echo ‘/bin/date‘ access denied>>/var/log/sshd.log
: deny
```

Los comodines permiten a los wrappers TCP coincidir más fácilmente con grupos de demonios o hosts. Se pueden utilizar entre otros, los siguientes comodines:

ALL hace corresponder todo. Se puede usar para la lista de demonios o en la lista de clientes.

LOCAL hace corresponder todos los nombres de máquinas que no contengan un punto (.), tal como localhost. Es decir, las que pertenecen a nuestro dominio.

KNOWN se corresponde con cualquier máquina/usuario de IP/nombre conocido

UNKNOWN se corresponde con cualquier máquina/usuario de IP/nombre desconocido

PARANOID se corresponde con cualquier nombre que no se corresponda con su dirección IP (por si se intenta camuflar la identidad real de la máquina que solicita la conexión). Hay una última palabra que también es útil. La palabra

EXCEPT permite proporcionar una lista con excepciones.

Patrones

Los patrones se pueden utilizar en el campo de lista de cliente de las reglas de acceso para especificar de forma más precisa grupos de host clientes. La siguiente es una lista de los patrones más comunes:

- Nombre de host comenzando con un punto (.). Al colocar un punto al comienzo de un nombre de host, se hace coincidir todos los hosts compartiendo los componentes listados del nombre. El ejemplo siguiente se aplicará a cualquier host dentro del dominio cica.es:

  ```
  ALL : .cica.es
  ```

- Dirección IP que termina con un punto (.). Al colocar un punto al final de una dirección IP hace corresponder todos los hosts compartiendo el grupo numérico inicial de la dirección IP. El ejemplo siguiente se aplicará a cualquier host dentro de la red 192.168.x.x:

  ```
  ALL : 192.168.
  ```

- Dirección IP/máscara de red.

  ```
  ALL : 192.168.0.0/255.255.254.0
  ```

- El asterisco (*). Los asteriscos pueden ser usados para coincidir grupos completos de nombres de host o direcciones IP, siempre y cuando no se mezclen en la lista de clientes conteniendo otros tipos de patrones. El ejemplo siguiente aplica a cualquier host dentro del dominio cica.es:

  ```
  ALL : *.cica.es
  ```

Las reglas de control de acceso aceptan además el operador, EXCEPT. Se puede usar tanto en la lista de demonios como en la lista de clientes de una regla.

El operador EXCEPT permite excepciones específicas a coincidencias más amplias dentro de la misma regla.

↬ En el ejemplo siguiente desde un archivo `hosts.allow`, todos los hosts de `cica.es` pueden conectarse a todos los servicios excepto `invitado.cica.es`:

```
ALL: .cica.es EXCEPT invitado.cica.es
```

En el otro ejemplo, situado en un archivo `hosts.allow`, clientes desde la red 192.168.0.x pueden usar todos los servicios excepto el servicio de FTP, vsftpd:

```
ALL EXCEPT vsftpd: 192.168.0.
```

¿Cómo y por qué actúa TCP-Wrappers? Pues muy fácil, actúa porque el demonio así lo requiere. Si el demonio ha sido compilado con soporte de TCP-Wrappers, él solito al ser ejecutado, mirará si existen los ficheros `/etc/hosts.allow` y `/etc/hosts.deny` y comprobará si le es aplicable alguna regla.

Los demonios que actúan bajo inetd, también pueden incluirse con control TCP-Wrappers. Se les especifica en el fichero `/etc/inetd.conf`.

```
telnet stream tcp nowait root /usr/sbin/tcpd in.telnetd
```

Para el anterior ejemplo, cuando se recibe una petición en el puerto 23 (telnet), de tipo stream y tcp por parte de un cliente, el superservidor inetd arranca un nuevo proceso por parte del superusuario. Este proceso es `/usr/sbin/tcpd`, que es el ejecutable de TCP-Wrappers. Después comprueba los ficheros de control de acceso y si se permite, lanza el demonio `in.telnetd`, el servidor original que presta el servicio.

↬**Ejemplos:**

```
# /etc/hosts.allow
#
# Permitir correo de salida a todo el mundo
in.smtpd: ALL
# telnet y FTP sólo a hosts dentro de mi dominio y al host de
# thales
telnetd, ftpd: LOCAL, thales.cica.es
# Permitir finger a cualquiera pero mantener un registro de quién es.
fingerd: ALL: (finger @%h | mail -s "finger desde %h" root)
```

```
# /etc/hosts.deny
#
# Desautorizar a todos los host con nombre sospechoso
ALL: PARANOID
#
# Desautorizar a todos los host.
ALL: ALL
```

La entrada `PARANOID` es redundante porque la otra entrada abarca todo en cualquier caso. Ambas entradas serían razonables por defecto dependiendo de sus requisitos particulares.

La configuración más segura es tener `ALL: ALL` por defecto en `/etc/hosts.deny` para después dar permiso específicamente a aquellos servicios y hosts que se desee en `/etc/hosts.allow`.

Capítulo 10

Terminal remoto. Telnet y SSH

> Si has seguido las secciones con atención, te habrás dado cuenta de que hemos planteado remedios contradictorios para solucionar los problemas del protocolo FTP:
>
> **No utilices el modo activo del FTP. ...**
>
> **No utilices el modo pasivo del FTP. ...**
>
> Por lo tanto, si sólo existen dos métodos para realizar transferencias de datos mediante FTP, y me aconsejáis que no utilice ninguna, ¿qué hago? Muy sencillo:
>
> **Medida de protección: No utilices FTP**
>
> (*Hackers en Linux*, Brian Hatch, James Lee y George Kurtz)

10.1. Visión general

Dentro de las labores de un administrador de sistemas está el acceso remoto a los mismos, ya sea para buscar información en algún fichero del sistema, para copiar información o ejecutando en remoto algún comando. Existen varias utilidades que realizan estas funciones como telnet, ftp, rsh o rlogin.

10.1.1. Acceso remoto: telnet

El protocolo telnet[1] es una herramienta muy útil a la hora de administrar sistemas basados en Unix/Linux en cualquiera de sus sabores. Permite acceder a una máquina remotamente, de la misma forma que lo haríamos si estuviéramos sentados delante de la consola y utilizásemos su teclado para introducir los comandos. Nos proporcionará un mecanismo para conectarnos a un servidor remoto y ejecutar comandos en él de forma totalmente interactiva. De esta forma el sistema local es transparente al usuario, el cual tiene la sensación de estar conectado directamente a la máquina remota.

Los comandos que se teclean por parte del usuario son transmitidos directamente a la máquina remota y la respuesta de ésta es mostrada en la pantalla del usuario. Una conexión interactiva por telnet se conoce también como *login* remoto.

Para ello, el ordenador del usuario debe tener la habilidad de:

- Establecer una conexión con otra máquina
- Emular un terminal compatible con la máquina remota
- Regular el flujo de datos desde el terminal del usuario a la máquina remota y viceversa

[1] El término telnet proviene de *TELecommunication NETwork*

Esto es posible mediante el uso del protocolo TCP, el cual permite transmitir datos entre dos máquinas de forma coherente, y del protocolo IP, que proporciona una dirección única de 32 bits para cada máquina conectada a la red. Sobre estas bases está construido telnet, proporcionando así una emulación local de un terminal compatible con el servidor remoto.

La conexión telnet sobre TCP se establece entre el puerto de la máquina del usuario U y el puerto del servidor remoto S. El servidor remoto escucha en el puerto 23 a la espera de nuevas conexiones. Al ser la conexión TCP *full-duplex* e identificada por el par de puertos anteriores y el par de direcciones origen y destino, el servidor puede mantener tantas conexiones simultáneas que utilizan el puerto 23 y diferentes puertos clientes U como sean necesarias.

El punto débil de este protocolo, tal como hemos visto en los ejemplos del programa Ethereal, es que todos los datos se transmitirán en claro en la red. Si un usuario captura los datos que viajan en la red con programas como `tcpdump` o Ethereal podemos poner en compromiso la seguridad de nuestro sistema.

Para acceder al sistema remoto se nos solicitará la identificación para poder entrar al sistema. Por ejemplo[2] para acceder a la máquina MILETO escribiremos:

```
$telnet mileto.cica.es
```

Con Linux podemos acceder vía el cliente `telnet` a cualquier máquina remota, pero para ser servidor tenemos que cargar el paquete adecuado, éste es:

Fedora: `apt-get install telnet-server`[3]

Además hay que habilitar el servicio (con `ntsysv`, ...), por ejemplo, editando el fichero `/etc/xinetd.d/telnet` y cambiando `disable=yes` por `disable=no`. Después hay que decir a `xinetd` que recargue la configuración mediante:

```
#/etc/init.d/xinetd reload
```

Debian: `apt-get install telnetd`[4]

y reiniciamos el demonio inetd

```
#/etc/init.d/inetd
```

Como ya hemos comentado, telnet "da un paseo" a las contraseñas en texto plano por Internet. Una forma segura de telnet en nuestra intranet puede ser la de usar la seguridad adicional de tcpwrappers de la forma[5]:

```
$cat /etc/hosts.allow
in.telnetd: 172.26.0. 127.0.0.1
$cat /etc/hosts.deny
in.telnetd: ALL
```

O bien usando[6] el propio `xinetd`, añadiendo al fichero `/etc/xinetd.d/telnet` la línea:

```
only_from = 172.26.0.0/24 127.0.0.1
```

y después:

```
#/etc/init.d/xinetd reload
```

➥ Para practicar

1. Cargar un servidor de telnet, modificar el texto de bienvenida (`/etc/issue.net`) y conseguir que los accesos estén limitados a la máquina local y a la subred 172.26.0.0/24.

[2] Previamente debemos haber establecido la conexión con nuestro proveedor de Internet.

[3] Se encuentra en el CD3 de la distribución, es el paquete: `telnet-server-0.17-30`

[4] Podemos optar por instalar mejor el paquete:

`apt-get install telnetd-ssl`

"`telnet(d)-ssl` reemplaza al normal `telnet(d)` empleando autenticación SSL y cifrado. Puede interoperar con el telnet(d) normal en ambas direcciones. Comprueba si el otro extremo también usa SSL, y si no es posible, emplea el protocolo telnet estándar."

[5] Que no se olvide pulsar **INTRO** después de escribir las líneas

[6] Sólo para Fedora.

2. Es interesante probar la posibilidad que nos ofrece Linux de trabajar en modo gráfico con programas situados en otro equipo, para esto tendremos que:
   Desde un Xterm de la máquina local ejecutaremos[7]
   `$ xhost +máquina_remota`
   después haremos un telnet a la máquina remota y una vez conectados escribiremos
   `$ export DISPLAY=máquina_local:0`
   por último ya sólo tenemos que ejecutar el comando que deseemos, por ejemplo, podéis probar con
   `$ mozilla &` ■

Sumarizando, el servicio telnet es inseguro y, aunque las extensiones ssl le puedan aportar seguridad, es mejor utilizar el servicio ssh.

10.1.2. Copia remota: ftp

El servicio ftp se utiliza para cargar y descargar archivos de la red. Este servicio puede verse dividido en dos partes:

- Los usuarios con cuenta en el sistema pueden acceder a su propio sistema de archivos y cargar y descargar información.
- Utilización anónima, en la que se permite que cualquiera (sea o no usuario del sistema) se conecte a una sección del sistema de archivos del servidor para cargar y descargar información.

Las cuestiones relacionadas con la configuración del ftp anónimo por parte de los administradores son numerosas. Si el sistema de archivos y la información de usuario de ftp para el acceso público no se crean con los permisos adecuados, podemos llegar a situaciones en las que usuarios sin cuenta en el sistema pasen del espacio público al privado del servidor.

Existen varios servidores de ftp que pueden instalarse en un sistema linux[8], aunque normalmente el incluido en las distribuciones suele ser `wu-ftpd`. Independientemente del servidor que se decida utilizar, es necesario dedicar un tiempo a diseñar las formas de acceso y a qué partes del sistema de archivos, para los distintos usuarios.

Es posible regular el acceso al usuario así como permitir sólo a determinadas direcciones IP acceder a nuestro servidor por ftp. Sin embargo, tal como ocurre con el protocolo telnet, no es posible evitar[9] que los datos que viajan por la red entre el servidor y el cliente viajen en claro. Así proponemos realizar una captura de una sesión telnet y ftp con Ethereal, tal como se vió en la entrega anterior, para tomar conciencia de qué datos pueden "robarnos" por la red.

wu-ftp

Sólo para que se tenga una referencia, veamos una pocas pinceladas sobre uno de los históricos e inseguros servidores de ftp: `wu-ftp`. Para instalarlo:

Fedora:

Descargamos el paquete `wu-ftpd`[10]. Para un acceso con clave de usuario, se nos ubicará en el directorio `$HOME` del usuario y para un acceso anónimo (sin usuario del sistema) en el directorio `/var/ftp/`. En el directorio `/var/ftp/pub` podemos dejar archivos listos para ser recogidos a través de accesos anónimos[11].

[7] Donde `máquina_remota` es o bien la dirección IP de la máquina remota, o bien, el nombre de esa máquina

[8] Algunos de los más comunes son ftpd, glFtpD, lukemftpd, vsftpd o ProFTPD

[9] Las nuevas versiones van incorporando mecanismos para cifrar los datos en su tránsito por la red.

[10] `http://rpmfind.net/linux/RPM/redhat/updates/8.0/i386/wu-ftpd-2.6.2-12.i386.html`

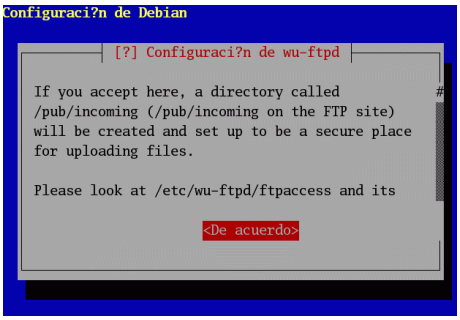
[11] Para permitir el acceso anónimo (*anonymous*) hay que instalar el paquete `anonftp` `ftp://rpmfind.net/linux/redhat/8.0/en/os/i386/RedHat/RPMS/anonftp-4.0-12.i386.rpm`. Sólo deberíamos instalarlo si estamos seguros de que eso es lo que queremos, un acceso anónimo a ficheros, y la seguridad no se va a resentir.

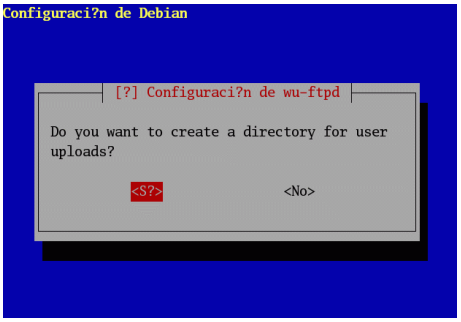
Debian: `apt-get install wu-ftpd`

```
root@guadalinex:~/curso-linux/entrega2/entrega05-2# apt-get install wu-ftpd
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se áinstalarn los siguientes paquetes NUEVOS:
wu-ftpd
0 actualizados, 1 se áinstalarn, 0 para eliminar y 123 no actualizados.
Necesito descargar 275kB de archivos.
Se áutilizarn 770kB de espacio de disco adicional édespus de desempaquetar.
Des:1 http://http.guadalinex.org sarge/main wu-ftpd 2.6.2-17.2 [275kB]
Descargados 275kB en 6s (44,0kB/s)
Preconfiguring packages ...
Seleccionando el paquete wu-ftpd previamente no seleccionado.
(Leyendo la base de datos ...
106874 ficheros y directorios instalados actualmente.)
Desempaquetando wu-ftpd (de .../wu-ftpd_2.6.2-17.2_i386.deb) ...
Configurando wu-ftpd (2.6.2-17.2) ...
Disabling other FTP services in /etc/inetd.confñ
Aadiendo usuario del sistema ftp...
Adding new group 'ftp' (109).
Adding new user 'ftp' (109) with group 'ftp'.
Creando el directorio home /home/ftp.
The anonymous FTP user has been successfully set up.
Starting FTP server: wu-ftpd.
```

Se nos preguntará si deseamos que el servicio esté controlado por `inetd` o `standalone`. Optaremos por dejar el valor por defecto (`standalone`), es decir, que funcione como servidor independiente, ya que de esa manera podemos optimizar la velocidad de respuesta. Después pregunta si vamos a permitir accesos anónimos (login: *anonymous* y password: dirección de correo-e por convención).

Sólo en el caso de que optemos que **<sí>** (¡no deberíamos ser tan atrevidos!) tendremos que optar por crear la cuenta ftp (`/home/ftp`) y después la zona del disco (`/home/ftp/pub/incoming`) de accesos anónimos





En modo gráfico, sería equivalente al proceso anterior.

Configuración y clientes Para ver cómo funciona podemos ejecutar[12]:

```
$ftp localhost
Connected to guadalinex.
220 guadalinex FTP server (Version wu-
2.6.2(1) Sat Aug 21 20:26:00 UTC 2004) ready.
Name (localhost:root): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password:
230-Welcome, archive user anonymous@guadalinex !
230-
230-The local time is: Sat Mar 5 14:06:13 2005
230-
230-This is an experimental FTP server. If have any unusual problems,
230-please report them via e-mail to <root@guadalinex>.
230-
230-If you do have problems, please try using a dash (-) as the first
230-character of your password -- this will turn off the continuation
230-messages that may be confusing your FTP client.
230-
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 24
d--x--x--x 2 0 4096 Mar 5 13:03 bin
d--x--x--x 2 0 4096 Mar 5 13:03 dev
d--x--x--x 2 0 4096 Mar 5 13:03 etc
d--x--x--x 2 0 4096 Mar 5 13:03 lib
dr-xr-xr-x 3 0 4096 Mar 5 13:03 pub
-rw-r--r-- 1 0 346 Mar 5 13:03 welcome.msg
226 Transfer complete.
ftp> bye
221-You have transferred 0 bytes in 0 files.
221-Total traffic for this session was 1104 bytes in 1 transfers.
```

[12]Si no nos sentimos cómodos desde la línea de comandos, es mejor usar los programas: `mc`, `gftp`.

```
221-Thank you for using the FTP service on guadalinex.
221 Goodbye.
```

El demonio FTP se configura en Fedora[13] a través del fichero `/etc/xinetd.d/wu-ftp`[14] y del fichero[15] `/etc/ftpaccess`, analicemos algunas de las entradas de este fichero:

```
# Denegar el acceso a los usuarios/grupos indicados:
# con uid/gid menor que 99 o mayor que 65534
# al resto se les permite el acceso
deny-uid %-99 %65534-
deny-gid %-99 %65534-
allow-uid ftp
allow-gid ftp

# Se define la clase all como aquella formada por los usuarios
# del sistema, los usuarios reales con cuentas de invitados
# y el resto para todas las IP posibles (*)
class   all   real,guest,anonymous  *

# Correo del administrador
email root@localhost

# Si hay 5 intentos fallidos se cierra la conexión
loginfails 5

# Si un usuario se registra o cambia de directorio se le
# avisa (si existen) de ficheros README
readme  README*    login
readme  README*    cwd=*

# Mensaje de bienvenida (hay que crearlos)
message /welcome.msg            login
message .message                cwd=*

# Permite que se compriman/empaqueten ficheros a todos los usuarios
compress        yes             all
tar             yes             all

# Opciones no permitidas a los usuarios que se listan

chmod           no              guest,anonymous
delete          no              anonymous
overwrite       no              anonymous
rename          no              anonymous

# Se registran en /var/log/xferlog las transferencias de los
#usuarios (separados por comas), tanto de carga como de descarga.
log transfers anonymous,guest,real inbound,outbound

# Si se va a cerrar el servidor se avisa a los usuarios conectados de ello
# según las directrices del fichero especificado (man ftpshut)

shutdown /etc/shutmsg
```

[13] En Debian se usa `/etc/inetd.conf` si está controlado por inetd

[14] Hay que comprobar si está inactivo (`disable=yes`), en ese caso poner `disable=no` y hacer que `xinetd` relea la configuración.

```
# /etc/init.d/xinetd reload
```

[15] `/etc/wu-ftpd/ftpacces` en Debian.

```
# Si se conecta un usuario anónimo (anonymous) tendrá que introducir
# como contraseña una direccción de correo según esa norma
# es decir, nombre@host.dominio
passwd-check rfc822 warn
```

10.1.3. Una solución más segura

Como ya hemos visto, estas utilidades tienen un problema muy serio, su falta de seguridad. Si no se transmiten los datos por la red de forma segura, cualquier intruso puede interceptar nuestros datos y utilizarlos de forma fraudulenta. Incluso en el caso de telnet, lo que puede obtener el intruso es nuestro usuario y password del sistema además del contenido de la comunicación y a partir de ahí, intentar obtener la clave de root y el control total de nuestra máquina..

La solución a este problema es utilizar un protocolo alternativo denominado SSH[16]. Este protocolo cifrará los datos antes de pasarlos a la red, descifrándolos cuando lleguen a su destino. El procedimiento de cifrado asegura que el intruso que capture los datos será incapaz de descifrarlos y verlos. El resultado es un cifrado transparente, ya que el usuario no tiene que realizar ningún proceso previo con los datos que va a transmitir.

El protocolo SSH tiene una arquitectura de cliente/servidor. El servidor SSH es un programa, ejecutado normalmente con permisos de administrador, encargado de aceptar y rechazar las conexiones entrantes a la máquina. Por otro lado, el programa cliente SSH permite a un usuario hacer peticiones desde una máquina remota a la que tiene el servidor SSH activo.

A pesar del concepto shell que aparece en el nombre del protocolo, no hay que confundir SSH con una shell o intérprete de comandos. El funcionamiento de SSH se basa en establecer un canal de comunicación seguro con un ordenador remoto. A partir de ahí se pueden ejecutar aplicaciones, entre ellas una shell.

Nos centraremos en el uso de la implementación libre OpenSSH, siendo los ficheros y comandos que utilicemos los correspondientes a esta implementación. Dentro de los clientes que proporciona esta implementación tenemos `ssh` (conexión segura), `scp` (copia de ficheros segura) y `slogin` (login seguro), que sustituyen a `rsh`, `rcp` y `rlogin`:

Login remoto. Supongamos que tenemos cuentas de usuario en distintas máquinas que se encuentran en internet. Normalmente realizamos la conexión con telnet desde nuestro ordenador personal. Además de transmitir el usuario/clave por la red en formato texto plano, toda la información referente a la sesión que hayamos abierto será legible por cualquier intruso que se encuentre escuchando en la red. Al utilizar un cliente SSH nos autenticaremos en el sistema remoto utilizando una conexión cifrada, el servidor SSH nos permitirá el acceso y toda la sesión es cifrada. Los datos viajarán por la red encriptados entre el cliente y el servidor. Al realizarse este proceso de forma transparente, no existirán diferencias entre el uso de telnet y el de un cliente SSH[17].

```
ssh -l legolas 172.26.0.40
```

Soporta de forma completa la ejecución de aplicaciones en el entorno X. Cuando un cliente se conecta a un servidor SSH intercambian claves de cifrado y a partir de ahí el servidor autentica al cliente, bien con RSA o bien mediante contraseña. Cuando se ha iniciado la conexión, el servidor lanza un servidor X ficticio de forma que las aplicaciones que se ejecutan en el servidor SSH se conectan al servidor X ficticio y de "forma segura" SSH reenvía los datos

[16] Existen varias versiones de SSH (SSH-1 y SSH-2), pero para simplificar nos centraremos en el uso de OpenSSH que es una implementación libre que contempla las dos versiones. Aparte, telnet y ftp están incorporando SSL a su funcionamiento.

[17] También podemos escribir

```
ssh legolas@172.126.0.40
```

al servidor X real. Es más seguro y fácil de usar que telnet y para ejecutar una aplicación gráfica (una vez realizada la conexión usando ssh) sólo hay que ejecutar el programa desde la xterm que estemos usando para la conexión.

Ejecución remota de comandos. Supongamos que queremos ejecutar un comando de forma remota en una máquina. En caso de utilizar `rsh` los datos resultantes de la ejecución del comando viajarán en claro por la red. Mediante SSH la salida que obtenemos es idéntica a si utilizásemos `rsh`, pero con la diferencia que estos datos estarán ocultos a cualquier intruso que escucha en la red.

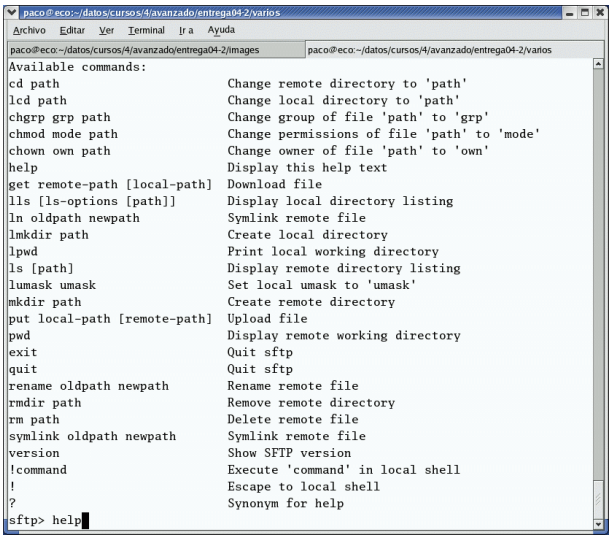
```
ssh 172.26.0.40 /usr/bin/who
```

Transferencia de ficheros. Supongamos que queremos transferir un fichero entre la máquina en la que nos encontramos y un servidor remoto, conteniendo dicho fichero información confidencial que no deseamos que nadie obtenga. Tradicionalmente utilizaríamos las utilidades ftp o rcp, pero cualquier intruso escuchando en nuestra red podría obtener estos datos. Mediante SSH el fichero se puede transmitir con total seguridad, de forma que se cifra antes de salir del origen y se descifra una vez llegado a su destino.

```
scp confidencial.txt legolas@172.26.0.40
```

En general, el comando que usaremos es

```
sftp legolas@172.26.0.1
```



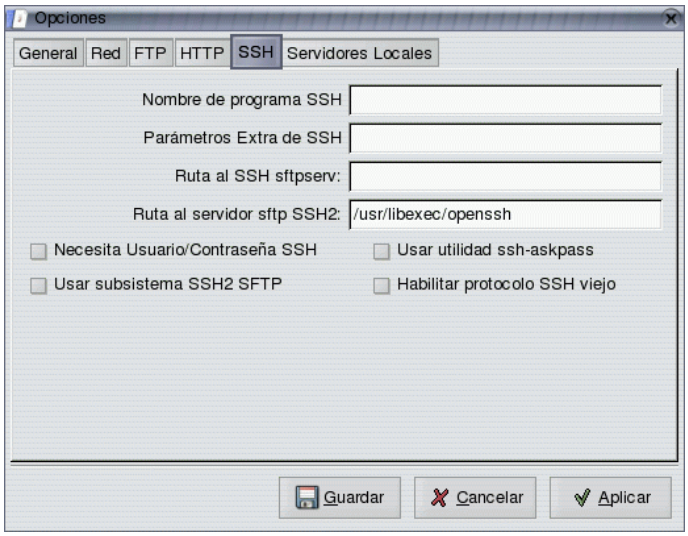
e iniciaremos una comunicación, tipo ftp segura[18] (*secure ftp*). Podemos realizar conexiones sftp en modo gráfico con el programa `gftp`:

- Para que funcione el cliente gftp en modo SSH, podemos instalar el paquete `ssh-askpass`, que gestionará la autenticación.
- Otra forma sería, para trabajar con SFTP, cambiar la configuración del programa, para eso pulsamos en el menú principal sobre **FTP** y en la ventana que aparece sobre **Opciones**

[18] Los comandos disponibles son similares a los del ftp.

Podemos optar por una de la opciones

1. marcar la casilla **Usar subsistema SSH2 SFTP**, o bien
2. pestaña **SSH** y en el último campo escribimos `/usr/libexec/openssh`



- Si deseamos conectar con un servidor con Guadalinex se escribe `/usr/lib`

10.2. La Criptografía llega en nuestra ayuda

Con el uso de Internet y la necesidad de proteger las comunicaciones a través de redes de comunicación inseguras, la protección de la información se transforma en una necesidad y con ello se populariza la criptografía. Es necesario manejar herramientas que nos proporcionen un elevado nivel de seguridad cuando utilizamos Internet. Se vuelven comunes conceptos matemáticos como cifrado, descifrado, criptoanálisis, firma digital, Autoridades de Certificación, que aunque complejos en sus fundamentos, deben hacerse asequibles en su uso.

Definimos la criptografía como una rama inicial de las Matemáticas y en la actualidad muy difundida en la Informática y la Telemática, que utiliza métodos y técnicas con el objeto principal de cifrar y/o proteger un mensaje o archivo por medio de un algoritmo, usando una o más claves[19]. Esto da lugar a diferentes tipos de sistemas de cifrado que permiten asegurar cuatro aspectos de la seguridad informática: la confidencialidad, la integridad, la disponibilidad y el no repudio de emisor y receptor.

Confidencialidad: Los componentes del sistema serán accesibles sólo por aquellos usuarios autorizados.

Integridad: Los componentes del sistema sólo pueden ser creados y modificados por los usuarios autorizados.

Disponibilidad: Los usuarios deben tener disponibles todos los componentes del sistema cuando así lo deseen.

No Repudio: Asociado a la aceptación de un protocolo de comunicación entre emisor y receptor (cliente y servidor) normalmente a través del intercambio de sendos certificados digitales de autenticación. Se habla entonces de No Repudio de Origen y No Repudio de Destino, forzando a que se cumplan todas las operaciones por ambas partes en una comunicación.

[19]Para proteger un mensaje M se cifra con la función f, dando lugar al mensaje cifrado C $f(M) = C$. El descifrado consiste en aplicar la función inversa al mensaje cifrado para obtener el mensaje original $f^{-1}(C) = M$

Los sistemas criptográficos modernos[20] se pueden clasificar en simétricos (o de clave privada) y asimétricos (o de clave pública).

- Criptosistemas simétricos: Existirá una única clave secreta que deben compartir emisor y receptor. Con la misma clave se cifra y se descifra por lo que la seguridad reside sólo en mantener dicha clave en secreto.

- Criptosistemas asimétricos: Cada usuario crea un par de claves, una privada y otra pública, inversas dentro de un cuerpo finito. Lo que se cifra en emisión con una clave, se descifra en recepción con la clave inversa. La seguridad del sistema reside en la dificultad computacional de descubrir la clave privada a partir de la pública. Para ello, usan funciones matemáticas de un solo sentido con trampa.

  Son funciones matemáticas de un solo sentido[21] y que nos permiten usar la función en sentido directo o de cálculo fácil para cifrar y descifrar (usuarios legítimos) y fuerza el sentido inverso o de cálculo difícil para aquellos impostores, hackers, etc. que lo que desean es atacar o criptoanalizar el mensaje cifrado.

  Pongamos como ejemplo el problema de la factorización de números grandes. El cálculo directo es fácil. Dados dos números p y q, por muy grandes que sean, hallar p*q = n es fácil, sobre todo para un ordenador. Sin embargo, el cálculo inverso, que es la descomposición en factores de un número grande n = p*q , para el caso de p y q primos entre sí, es computacionalmente complejo.

¿Por qué utilizamos dos sistemas criptográficos distintos? Porque no existe el sistema perfecto para todos los usos. Los sistemas de clave pública son muy lentos cifrando, pero tienen firma digital y la gestión de claves se puede utilizar cuando podemos tener miles o millones de usuarios. Los sistemas de clave secreta son muy rápidos cifrando, pero no tienen firma digital y mantener la clave secreta es complicado cuando tenemos más de unos cuantos usuarios.

La solución es utilizar cada sistema para lo que es adecuado. Así, para el cifrado de la información se utilizarán sistemas de clave secreta y para la firma digital y el intercambio de claves de sesión, se utilizarán sistemas de clave pública.

➥ **Para practicar:**

Utilizaremos la herramienta criptográfica OpenSSL, que es algo así como una "navaja suiza" que hace de todo con respecto a los procedimientos criptográficos.

- Veamos un ejemplo de criptografía simétrica. Necesitamos un mensaje que cifrar, para ello generamos un fichero de texto, que llamamos `fichero.txt`, con el siguiente contenido:

```
En un lugar de la Mancha, de cuyo nombre no quiero acordarme, no ha mucho tiempo que...

...y os hacen merecedora del merecimiento que merece la vuestra grandeza.
```

Utilizaremos el algoritmo CAST (otros serían el 3DES, Blowfish...) para cifrar el fichero. Para ello nos pedirá una clave[22], que le proporcionaremos.

```
[root@unit3 cep]# openssl cast -in fichero.txt -out fichrc5.bin
enter cast5-cbc encryption password:
Verifying - enter cast5-cbc encryption password:
```

[20] Para distinguirlos de los sistemas utilizados en la antigüedad basados en rotaciones, permutaciones... del mensaje y basados principalmente en el secreto del algoritmo de cifrado

[21] Se llaman así porque la ejecución de la función $f(M) = C$ es siempre fácil, pero la ejecución de la función inversa $f^{-1}(C) = M$ es difícil salvo que se tenga la trampa.

[22] Con ella formará la clave privada

Si deseamos ver el contenido descifrado de nuevo, tendremos que aplicar el algoritmo al fichero cifrado y proporcionar la clave. Vemos que la misma clave que sirve para cifrar, es utilizada para descifrar.

```
[root@unit3 cep]# openssl cast -d -in fichrc5.bin
enter cast5-cbc decryption password:
En un lugar de la Mancha, de cuyo nombre no quiero acordarme, no ha
mucho tiempo que...
...y os hacen merecedora del merecimiento que merece la vuestra grandeza
    .
```

- Ejemplo de criptografía asimétrica:

  Generaremos la pareja de claves (clave privada y clave pública) mediante el algoritmo asimétrico RSA. En este caso existen dos claves, una pública y otra privada, con la propiedad de que lo cifrado con una de ellas sólo puede ser descifrado con la otra.

```
[root@unit3 cep]# openssl genrsa -out clavesrsa.pem 1024
Generating RSA private key, 1024 bit long modulus
.......++++++
..............................++++++
e is 65537 (0x10001)
```

Visualizamos las claves generadas. Se encuentran en formato PEM, que es una codificación en base64, formada solamente por caracteres ASCII, del formato binario ASN.1.

```
[root@unit3 cep]# more clavesrsa.pem
-----BEGIN RSA PRIVATE KEY-----
MIICXAIBAAKBgQC+9Z8yUuIrOMfiB9RUUuu4KoKZCinUflU5MJsdKHmQvVuRJ+oq
A943VYGYIZOEoEv0EWntkmsz0bpB53nY48bDHbSK1DbqrpyC3AytrfdpJbJtwl6L
E8IudlFe5atX0u7mQBzi2UhDoH3anT8xk3YAvSpMyghi9JQtOYW3V/JdLwIDAQAB
AoGAMATT43O0K+o7WTteyVWQsN7/uDw2CJk8FQWD+/ERoLP9MqM23xnZ51A5JmhL
EC6+6sD2QidvjbhPfgMXQNMGR9rJjqF545baG7at35fkKfllBvnN1CZUBjqRv94P
T3GhbTBMIZP+ozBY1XLLA52MBK+M5fct6Hv/n/ic2GbprQECQQDuw6NKgb6qee0W
I+bhYswOYkLBQsHRjd8oIiDDr8OUtrf7kkaUdVBX/MzslCYfH7iYiHps8ZNJ5cfQ
MsQBKAHNAkEAzL6NZMp3z3rHOJZpQdW0G43p528UgLXawqCqQJD+G6iKc18WDaTb
r/XHa5DZHZnZ28jB5Xq7IdsyVP4ubx2O6wJAasg4OVQ5b4jEDcjUsyw6UpyDemde
w1eN3CcXPC1ZbSMiuXI7+p1U52T6STwgqH7JUf6Hsj2AP+ZyLJznqBS6aQJAY3iw
eSdkzgh4gaWRvcp1lm18FlSBQYcYoTYtgPDwg79+hE7OCBLwKKzgFgJpeVgl4AHJ
MhROlkRIT8KuDI3vwQJBAOk0VQPN0eLXEPQ9Zvxr5sO3YTIA3iChWuA+yUt97ZRC
/5PZNS/EY7GRpEECN3yHOy1YPVe2R6hp2/Pndb9ZQA8=
-----END RSA PRIVATE KEY-----
```

Si queremos mostrar las claves generadas en formato legible.

```
[root@unit3 cep]# openssl rsa -in clavesrsa.pem -noout -text
Private-Key: (1024 bit)
modulus:
00:be:f5:9f:32:52:e2:2b:38:c7:e2:07:d4:54:52:
eb:b8:2a:82:99:0a:29:d4:7e:55:39:30:9b:1d:28:
79:90:bd:5b:91:27:ea:2a:03:de:37:55:81:98:21:
93:84:a0:4b:f4:11:69:ed:92:6b:33:d1:ba:41:e7:
79:d8:e3:c6:c3:1d:b4:8a:d4:36:ea:ae:9c:82:dc:
0c:ad:ad:f7:69:25:b2:6d:c2:5e:8b:13:c2:2e:76:
51:5e:e5:ab:57:d2:ee:e6:40:1c:e2:d9:48:43:a0:
7d:da:9d:3f:31:93:76:00:bd:2a:4c:ca:08:62:f4:
94:2d:39:85:b7:57:f2:5d:2f
```

```
publicExponent: 65537 (0x10001)
privateExponent:
30:04:d3:e3:73:b4:2b:ea:3b:59:3b:5e:c9:55:90:
b0:de:ff:b8:3c:36:08:99:3c:15:05:83:fb:f1:11:
a0:b3:fd:32:a3:36:df:19:d9:e7:50:39:26:68:4b:
10:2e:be:ea:c0:f6:42:27:6f:8d:b8:4f:7e:03:17:
40:d3:06:47:da:c9:8e:a1:79:e3:96:da:1b:b6:ad:
df:97:e4:29:f2:25:06:f9:cd:d4:26:54:06:3a:91:
bf:de:0f:4f:71:a1:6d:30:4c:21:93:fe:a3:30:58:
d5:72:cb:03:9d:8c:04:af:8c:e5:f7:2d:e8:7b:ff:
9f:f8:9c:d8:66:e9:ad:01
prime1:
00:ee:c3:a3:4a:81:be:aa:79:ed:16:23:e6:e1:62:
cc:0e:62:42:c1:42:c1:d1:8d:df:28:22:20:c3:af:
c3:94:b6:b7:fb:92:46:94:75:50:57:fc:cc:ec:94:
26:1f:1f:b8:98:88:7a:6c:f1:93:49:e5:c7:d0:32:
c4:01:28:01:cd
prime2:
00:cc:be:8d:64:ca:77:cf:7a:c7:38:96:69:41:d5:
b4:1b:8d:e9:e7:6f:14:80:b5:da:c2:a0:aa:40:90:
fe:1b:a8:8a:73:5f:16:0d:a4:db:af:f5:c7:6b:90:
d9:1d:99:d9:db:c8:c1:e5:7a:bb:21:db:32:54:fe:
2e:6f:1d:8e:eb
exponent1:
6a:c8:38:39:54:39:6f:88:c4:0d:c8:d4:b3:2c:3a:
52:9c:83:7a:67:5e:c3:57:8d:dc:27:17:3c:2d:59:
6d:23:22:b9:72:3b:fa:9d:54:e7:64:fa:49:3c:20:
a8:7e:c9:51:fe:87:b2:3d:80:3f:e6:72:2c:9c:e7:
a8:14:ba:69
exponent2:
63:78:b0:79:27:64:ce:08:78:81:a5:91:bd:ca:75:
96:6d:7c:16:54:81:41:87:18:a1:36:2d:80:f0:f0:
83:bf:7e:84:4e:ce:08:12:f0:28:ac:e0:16:02:69:
79:58:25:e0:01:c9:32:14:4e:96:44:48:4f:c2:ae:
0c:8d:ef:c1
coefficient:
00:e9:34:55:03:cd:d1:e2:d7:10:f4:3d:66:fc:6b:
e6:c3:b7:61:32:00:de:20:a1:5a:e0:3e:c9:4b:7d:
ed:94:42:ff:93:d9:35:2f:c4:63:b1:91:a4:41:02:
37:7c:87:3b:2d:58:3d:57:b6:47:a8:69:db:f3:e7:
75:bf:59:40:0f}
```

Destacar que la clave pública está formada por el módulo (modulus) y el exponente público (publicExponent), y que la clave privada es el exponente privado (privateExponent).■

10.3. SSH como cliente

Tal como acabamos de ver, SSH parte de una idea simple, aunque los componentes que nos ayudan a llegar al objetivo son complejos, como veremos más adelante. Para empezar partiremos de la parte cliente para empezar a profundizar en lo que SSH nos aporta en la seguridad de nuestras comunicaciones.

10.3.1. Sesiones remotas con SSH

Uno de los usos más extendidos es en el uso de sesiones remotas. Veamos con más atención cómo nos conectamos a un servidor remoto con SSH.

```
The authenticity of host 'fedora2 (172.26.0.41)' can't be established.
RSA key fingerprint is a4:78:71:06:60:ea:a9:1b:ed:0a:5e:80:c7:bc:d7:3b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'fedora2' (RSA) to the list of known hosts.
legolas@fedora2's password:
[legolas@fedora2 legolas]$
```

La primera vez que nos conectamos a una máquina remota aparecerá el mensaje anterior. Está relacionado con una función de seguridad de SSH denominada *known hosts* (hosts conocidos). Supongamos que un intruso quiere obtener la clave de entrada a una máquina y sabe que estamos utilizando SSH, por lo que no puede obtener este dato escuchando en la red. En lugar de eso, suplantará el nombre de la máquina por otra que él controle en la cual hay instalada una versión modificada de SSH. Le bastará con esperar a que nos conectemos a la máquina para obtener la clave que busca.

El mecanismo SSH *known hosts* previene estos ataques. Cuando un cliente SSH y un servidor establecen una conexión, cada uno de ellos comprueba la identidad del otro. No sólo el servidor autentica al cliente mediante la clave, el cliente también autentica al servidor mediante una clave pública. Básicamente, cada servidor SSH tiene un identificador único y secreto, llamado *host key*, para identificarse frente a los clientes que se conectan. La primera vez que nos conectamos a un servidor, la parte pública de la *host key* se copia en nuestra cuenta local (asumiendo que respondemos *yes*). Cada vez que nos conectemos a este servidor, el cliente SSH comprobará la identidad del servidor remoto con esta clave pública. Dicha clave pública, así como la del resto de máquinas con las que nos vayamos conectando se encuentra guardada en `$HOME/.ssh/known_hosts`

```
fedora2 ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAIEAxT0bamLgJbuF8cMRoLU1
HPehrK5iPNyiEktZ2ATh85Tq/+7pIGHwYcmiZcS13X4ppR41G1uTAnujB0
/PuX3JNDql0qFlrzNl857DHQuVI2+bfEjNSsjZ2z/u7BQy188Sqyfn3gpd
nm5fgwtMkLBb+MIfWZI04xc+OhBFWoKFNj0=
```

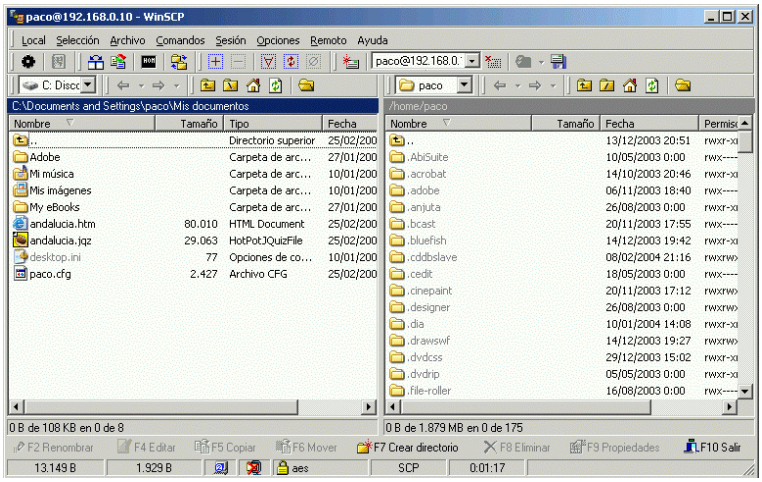
En caso de que el servidor al que nos conectemos no tenga una clave pública coincidente con la que tenemos almacenada en nuestra cuenta, aparecerá el siguiente mensaje:

```
[hugo@fedora hugo]$ ssh -l legolas fedora2
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@       WARNING: POSSIBLE DNS SPOOFING DETECTED!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
The RSA host key for fedora2 has changed,
and the key for the according IP address 172.26.0.41
is unchanged. This could either mean that
DNS SPOOFING is happening or the IP address for the host
and its host key have changed at the same time.
Offending key for IP in /home/hugo/.ssh/known_hosts:1
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that the RSA host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
a4:78:71:06:60:ea:a9:1b:ed:0a:5e:80:c7:bc:d7:3b.
Please contact your system administrator.
Add correct host key in /home/hugo/.ssh/known_hosts to get rid of this
    message.
Offending key in /home/hugo/.ssh/known_hosts:3
RSA host key for fedora2 has changed and you have requested strict checking.
Host key verification failed.
```

El hecho de que la clave pública que proporciona el servidor no coincida con la que tenemos almacenada, puede deberse a varias causas. Por ejemplo, el servidor remoto puede haber cambiado la *host key* por algún motivo especial. Así, el ver esta advertencia no quiere decir necesariamente que el sistema ha sido "hackeado"[23]. El administrador del sistema al que vamos a acceder será el que mejor nos pueda informar al respecto.

Una información añadida, clientes para Windows (gratuitos):

- El "genuino" para ssh, sftp, scp en modo comando:
  putty `http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html`
- Para sftp en modo gráfico, WinSCP:
  `http://winscp.sourceforge.net/eng/`



10.3.2. Autenticación por clave criptográfica

Hasta ahora, los ejemplos que hemos visto se basaban en la autenticación del cliente por medio de usuario/clave. Sin embargo esto tiene algunos inconvenientes, ya que las claves más seguras son las más difíciles de recordar. Además, el sistema operativo únicamente permite una clave por usuario, lo que en el caso de cuentas compartidas (por ejemplo root) presenta dificultades en el cambio de clave ya que debe comunicarse a todo el que utilice esta cuenta.

Para solucionar este problema, SSH proporciona un mecanismo de autenticación basado en clave pública.

Introducción a las claves

Cuando estamos hablando de clave, nos referimos a ella como una identidad digital. Es una cadena de datos binarios que nos identifica de forma unívoca, en el caso de SSh nos identificará ante un servidor.

Una clave de identidad tendrá dos partes:

Clave privada. Es la parte que sólo el usuario al que identifica debe tener. Es utilizada por SSH para probar la identidad ante un servidor.

Clave pública. Como su propio nombre indica, es pública y podemos distribuirla a todos aquellos sistemas que queramos que nos identifiquen. SSH la utiliza durante la autenticación para identificarnos ante un servidor remoto.

[23] Atacado por hackers (o por crackers, que son aún más malvados) que sustituyen los programas por otros con puertas traseras y barrido de pistas.

Veamos cómo sería la secuencia entre un cliente y un servidor SSH para comprender un poco mejor estos conceptos:

1. El cliente dice "Hola servidor, me gustaría conectarme por SSH a una de tus cuentas, la del usuario legolas".
2. El servidor dice "Bien, pero primero te desafiaré a que pruebes tu identidad" y el servidor envía datos conocidos como desafío[24] al cliente.
3. El cliente dice "Acepto tu desafio. Aquí tienes una prueba de mi identidad. La hice yo mismo mediante algoritmos matemáticos usando tu desafío y mi clave privada". Esta respuesta al servidor se conoce como *authenticator.*
4. El servidor dice "Gracias por el autenticador. Ahora examinaré la cuenta legolas para ver si puedes utilizarla". Lo que realmente hace el servidor es comprobar las claves públicas de la cuenta legolas para ver si el autenticador concuerda con alguna de ellas. En caso de que concuerden, el servidor dará su consentimiento para el acceso al sistema. En otro caso, la autenticación falla.

A continuación veremos con más detalle los comandos y ficheros implicados en este proceso.

Generación de claves

Para usar la autenticación criptográfica es condición necesaria e indispensable la generación de la identidad digital, es decir, la pareja clave pública/clave privada. Para generar esta pareja de claves utilizaremos la utilidad `/usr/bin/ssh-keygen`.

```
ssh-keygen [options]
```

`-b bits` Número de bits en la clave que se crea

`-f filename` Nombre del fichero donde se almacenará la clave

`-l` Enseña la marca del fichero de clave

`-p` Cambia la palabra de paso del fichero de clave privada

`-y` Lee la clave privada e imprime la pública

`-t type` Tipo de clave que se creará (dsa o rsa)

`-N phrase` Proporciona una nueva palabra de paso

Empezaremos viendo cómo generar una pareja de claves pública y privada:

```
[legolas@fedora legolas]$ ssh-keygen -t dsa -b 2048
Generating public/private dsa key pair.
Enter file in which to save the key (/home/legolas/.ssh/id_dsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/legolas/.ssh/id_dsa.
Your public key has been saved in /home/legolas/.ssh/id_dsa.pub.
The key fingerprint is:
c0:cb:fe:5f:2b:d2:05:06:90:69:a7:1d:89:71:75:46 legolas@fedora.elpiso.es
```

Una vez creada, podemos realizar varias operaciones. Empezaremos mostrando la marca del fichero de clave:

[24] Proviene del término inglés *challenge*

```
[legolas@fedora legolas]$ ssh-keygen -t dsa -l
Enter file in which the key is (/home/legolas/.ssh/id_dsa):
2048 c0:cb:fe:5f:2b:d2:05:06:90:69:a7:1d:89:71:75:46 /home/legolas/.ssh/
    id_dsa.pub
```

En caso de que queramos cambiar la palabra de paso:

```
[legolas@fedora legolas]$ ssh-keygen -t dsa -p
Enter file in which the key is (/home/legolas/.ssh/id_dsa):
Enter old passphrase:
Key has comment '/home/legolas/.ssh/id_dsa'
Enter new passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved with the new passphrase.
```

Como acabamos de ver, es necesario conocer la antigua palabra de paso. Sin embargo, existe otra forma de cambiarla sin necesidad de proporcionarla, lo cual es especialmente útil en el caso que se nos haya olvidado:

```
[legolas@fedora legolas]$ ssh-keygen -t dsa -N "elfo del bosque"
Generating public/private dsa key pair.
Enter file in which to save the key (/home/legolas/.ssh/id_dsa):
/home/legolas/.ssh/id_dsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /home/legolas/.ssh/id_dsa.
Your public key has been saved in /home/legolas/.ssh/id_dsa.pub.
The key fingerprint is:
72:ee:2e:0b:45:e5:c8:a4:10:5f:e8:d3:e6:59:94:c2 legolas@fedora.elpiso.es
```

10.3.3. Y ahora que tenemos las claves ... ¿qué hacemos con ellas?

Acabamos de ver cómo crear la pareja de claves y cómo obtener o modificar su información, sin embargo, ¿qué utilidad tiene esto?

Cuando las claves son utilizadas para la autenticación, el sistema operativo de la máquina mantiene la asociación entre el usuario y la contraseña. Para las claves criptográficas podemos establecer una asociación similar de forma manual. Después de crear el par de claves con `ssh-keygen` en el nodo local, deberemos instalar la clave pública en la cuenta del servidor remoto.

Volvamos al ejemplo anterior. Deberemos instalar la clave pública dentro de la cuenta `legolas` en el servidor remoto. Esto se hace editando el fichero `$HOME/.ssh/authorized_keys2` y añadiéndole la clave pública.

```
[legolas@fedora .ssh]$ ssh-keygen -t dsa -N "elfo del bosque"
Generating public/private dsa key pair.
Enter file in which to save the key (/home/legolas/.ssh/id_dsa):
/home/legolas/.ssh/id_dsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /home/legolas/.ssh/id_dsa.
Your public key has been saved in /home/legolas/.ssh/id_dsa.pub.
The key fingerprint is:
28:83:99:dc:da:ed:e3:45:be:d9:c9:bb:0a:8d:43:1c legolas@fedora.elpiso.es
[legolas@fedora .ssh]$ scp id_dsa.pub fedora2:.ssh/fedora.pub
legolas@fedora2's password:
id_dsa.pub                                    100%  614    342.2KB/s   00:00
```

Una vez copiada la clave pública en la máquina remota, en nuestro caso la máquina `fedora2`, creamos el fichero `$HOME/.ssh/authorized_keys2` y le damos los permisos necesarios[25]:

```
[legolas@fedora2 .ssh]$ cat fedora.pub >> authorized_keys2
[legolas@fedora2 .ssh]$ ls -l
total 16
-rw-rw-r--    1 legolas  legolas       614 feb 14 06:57 authorized_keys2
-rw-r--r--    1 legolas  legolas       614 feb 14 06:57 fedora.pub
-rw-------    1 legolas  legolas       668 feb 14 06:54 id_dsa
-rw-r--r--    1 legolas  legolas       615 feb 14 06:54 id_dsa.pub
[legolas@fedora2 .ssh]$ chmod 644 authorized_keys2
[legolas@fedora2 .ssh]$ ls -l
total 16
-rw-r--r--    1 legolas  legolas       614 feb 14 06:57 authorized_keys2
-rw-r--r--    1 legolas  legolas       614 feb 14 06:58 fedora.pub
-rw-------    1 legolas  legolas       668 feb 14 06:54 id_dsa
-rw-r--r--    1 legolas  legolas       615 feb 14 06:54 id_dsa.pub
```

A partir de ahora, una conexión que se realiza con la cuenta de usuario `legolas` desde la máquina `fedora` a la máquina `fedora2` utilizará el método de autenticación por clave criptográfica.

```
[legolas@fedora .ssh]$ ssh fedora2
Enter passphrase for key '/home/legolas/.ssh/id_dsa': elfo del bosque
[legolas@fedora2 legolas]$
```

Como acabamos de ver, el proceso resultante es similar a cuando utilizamos autenticación por usuario/clave. La única diferencia, a simple vista, es que hemos sustituido la clave por la palabra de paso de la clave criptográfica. Sin embargo, si examinamos el proceso con mayor profundidad vemos que en el caso de la clave criptográfica lo único que viaja por la red es la palabra de paso. Con la autenticación criptográfica, la palabra de paso sirve únicamente para desencriptar la clave pública y crear el autenticador.

De esta forma, la autenticación mediante clave pública es más segura que la autenticación por clave. Serán necesarios dos componentes secretos (fichero con clave criptográfica y palabra de paso) y será necesario capturar las dos para poder tener acceso al sistema. En el caso del fichero con la clave criptográfica, el intruso tendría que tener acceso físico al mismo.

Podemos seguir sacando partido a este tipo de autenticación. Como hemos visto, al generar las claves criptográficas hemos introducido una palabra de paso, sin embargo no es obligatorio. Dejándola en blanco lograremos acceder a la máquina remota sin necesidad de introducir ninguna palabra de paso. Esto sin embargo presenta un grave problema en lo referente a la seguridad. Si otra persona obtiene nuestra clave privada podrá acceder sin ningún control a los servidores donde hayamos configurado el acceso por clave pública.

10.3.4. El agente ssh

Como acabamos de ver, cada vez que utilizamos la clave pública para acceder a un sistema por ssh o scp es necesario teclear de nuevo la frase de paso. Este proceso puede hacernos pensar que no hay diferencia y que es igual de "cómodo" que el uso de la autenticación por usuario/clave. Sería muy bonito tener un mecanismo por el cual sólo sería necesario introducir la frase de paso una vez y quedaría almacenada para posteriores conexiones. Esta labor la realiza el agente ssh, implementado en la utilidad `ssh-agent`.

El agente almacenará en memoria las claves privadas y proporciona servicios de autenticación a los clientes ssh. Así, cuando estemos en nuestro sistema y deseemos conectarnos a otros, utilizaremos la frase de paso para desencriptar la clave privada y a partir de este momento será el agente el que recibe las peticiones de los clientes.

[25] En caso de que no cambiemos los permisos al fichero `authorized_keys2` con la máscara 644, no lograremos que el proceso de autenticación utilice la clave criptográfica.

En OpenSSH el nombre de este agente es `ssh-agent`. Normalmente ejecutaremos un único ssh-agent en nuestra sesión local, antes de ejecutar cualquier cliente ssh. Estos clientes se comunican con el agente a través del entorno de procesos, por lo que todos los clientes y procesos de la sesión tendrán acceso al agente. Puede arrancarse el agente desde una sesión como puede verse a continuación:

```
ssh-agent $SHELL
```

donde `SHELL` es la variable de entorno que almacena la shell de login. De esta forma el agente se ejecuta e invoca a un shell definido como proceso hijo. El efecto visual es que aparece otro prompt de shell, pero esta shell tiene acceso al agente.

Una vez que el agente está arrancado podemos empezar a cargar las claves privadas con las que accederemos a otros sistemas. La utilidad que proporciona OpenSSH para este fin es `ssh-add`.

```
ssh-agent $SHELL
sh-2.05b$ ssh-add
Enter passphrase for /home/legolas/.ssh/id_dsa: elfo del bosque
Identity added: /home/legolas/.ssh/id_dsa (/home/legolas/.ssh/id_dsa)
```

Ahora los clientes ssh pueden conectarse a los hosts remotos sin la necesidad de teclear de nuevo la frase de paso.

La utilidad `ssh-add` leerá, por defecto, la palabra de paso desde el terminal. Sin embargo, es posible utilizar la entrada estándar para realizar esta entrada de forma no interactiva.

Tal como hemos ejecutado anteriormente `ssh-add` (sin ningún parámetro) añade la clave privada que se encuentra en la localización estándar (`$HOME/.ssh/id_dsa`). En caso de utilizar otro nombre:

```
ssh-add fichero_con_clave_privada
```

Podemos añadir tantas claves como queramos, pudiendo obtener un listado de las mismas con esta misma utilidad:

```
ssh-add -l
```

Igualmente, podemos eliminar alguna de las claves que tiene almacenadas el agente, incluso borrarlas todas:

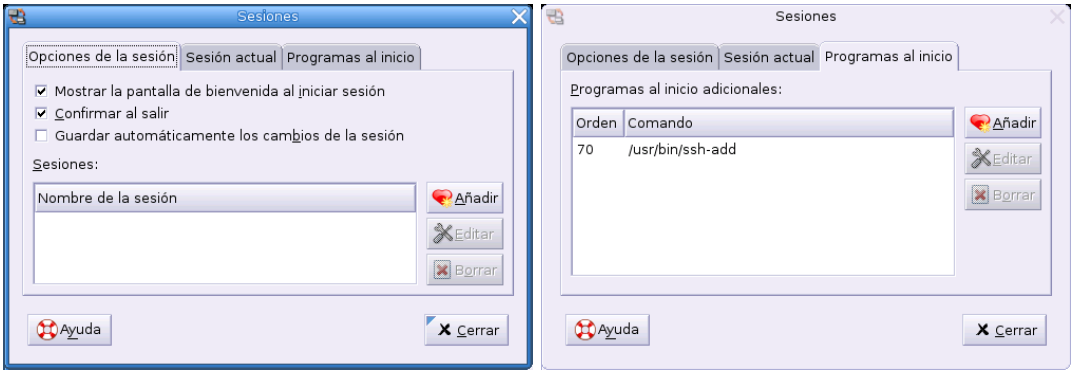
```
ssh-add -d fichero_con_clave_privada
ssh-add -D
```

10.3.5. Uso del agente SSH en GNOME

El procedimiento anterior puede automatizarse. Tanto Fedora como Guadalinex arrancan de forma automática `ssh-agent` al arrancar una sesión X. Solo quedaría que se cargue también en el inicio de sesión la clave privada mediante `ssh-add`.

Lo anterior es posible si modificamos la configuración de la sesión para que `ssh-add` sea uno de los programas que se arranca al inicio de la sesión X del usuario. Para cambiar la configuración de una sesión en Gnome utilizaremos `gnome-session-properties`.

Figura 10.1: Configuración de sesión



Crearemos una nueva entrada dentro de la configuración de la sesión para que se ejecute `ssh-add`. Estableceremos un número de prioridad más alto que cualquiera de los comandos existentes para asegurarnos que se ejecute el último. Mientras más alto el número, más baja será la prioridad. Si tenemos otros programas listados, éste debería tener la prioridad más baja. Para asegurarnos que se ejecute en último lugar le pondremos un número como 70 o superior.

Junto con este cambio en la configuración es necesario instalar tambien el paquete `ssh-askpass-gnome` que proporcionará una interfaz en la que indicar la frase con la que encriptamos la clave privada.

Guadalinex

```
apt-get install ssh-askpass-gnome
```

Fedora

```
rpm -Uvh openssh-askpass-gnome-3.9p1-7.i386.rpm
```

Quedará almacenada durante toda la sesión Gnome, a menos que la borremos de memoria mediante `ssh-add`, como vimos anteriormente.

10.4. Configuración del servidor SSH

Hasta ahora hemos estado viendo la funcionalidad de SSH desde el punto de vista del cliente, sin preocuparnos de la configuración existente en el servidor remoto. Evidentemente, será necesario tener el servicio de SSH activo en un servidor para poder conectarnos a él mediante SSH.

10.4.1. Instalación

Para disponer de la última versión:[26]

- En Guadalinex[27]

```
#apt-get install ssh ssh-askpass
```

[26] Para activarlo: `/etc/init.d/ssh start`

[27] El paquete `ssh` está ya instalado, para activar el demonio `sshd` podemos usar

```
#dpkg-reconfigure ssh
```

- Con Fedora[28]

```
openssh-3.9p1-7
openssh-clients-3.9p1-7
openssh-server-3.9p1-7
```

El último es el que nos permite disponer de un servicio ssh.

10.4.2. Configuración

La configuración del servicio se realiza con el fichero `/etc/ssh/sshd_config`. Para habilitar el uso del servicio SSH para acceder a nuestro servidor no sería necesario hacer ninguna modificación a los valores que vienen definidos por defecto. La mayoría de ellos están comentados, indicando el valor que se está tomando por defecto.

```
#       $OpenBSD: sshd_config,v 1.59 2002/09/25 11:17:16 markus Exp $
# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.
# This sshd was compiled with PATH=/usr/local/bin:/bin:/usr/bin
# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options change a
# default value.
#Port 22
#Protocol 2,1
#ListenAddress 0.0.0.0
#ListenAddress ::
# HostKey for protocol version 1
#HostKey /etc/ssh/ssh_host_key
# HostKeys for protocol version 2
#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_dsa_key
# Lifetime and size of ephemeral version 1 server key
#KeyRegenerationInterval 3600
#ServerKeyBits 768
# Logging
#obsoletes QuietMode and FascistLogging
#SyslogFacility AUTH
SyslogFacility AUTHPRIV
#LogLevel INFO
# Authentication:
#LoginGraceTime 120
#PermitRootLogin yes
#StrictModes yes
#RSAAuthentication yes
#PubkeyAuthentication yes
#AuthorizedKeysFile     .ssh/authorized_keys
# rhosts authentication should not be used
#RhostsAuthentication no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes
# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#RhostsRSAAuthentication no
# similar for protocol version 2
#HostbasedAuthentication no
```

[28] Se instalan por defecto, además de los comentados se instalan también
`openssh-askpass-3.9p1-7`
`openssh-askpass-gnome-3.9p1-7`

```
# Change to yes if you don't trust ~/.ssh/known_hosts for
# RhostsRSAAuthentication and HostbasedAuthentication
#IgnoreUserKnownHosts no
# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
# Change to no to disable s/key passwords
#ChallengeResponseAuthentication yes
# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#AFSTokenPassing no
# Kerberos TGT Passing only works with the AFS kaserver
#KerberosTgtPassing no
# Set this to 'yes' to enable PAM keyboard-interactive authentication
# Warning: enabling this may bypass the setting of 'PasswordAuthentication'
#PAMAuthenticationViaKbdInt no
#X11Forwarding no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PrintMotd yes
#PrintLastLog yes
#KeepAlive yes
#UseLogin no
#UsePrivilegeSeparation yes
#PermitUserEnvironment no
#Compression yes
#MaxStartups 10
# no default banner path
#Banner /some/path
#VerifyReverseMapping no
# override default of no subsystems
Subsystem       sftp    /usr/libexec/openssh/sftp-server
```

Veamos ahora brevemente el significado de algunos de estos parámetros:

`Port` Especifica en qué número de puerto el demonio SSH va a escuchar las conexiones entrantes.

`ListenAddress` Especifica la dirección IP del interfaz de red en el que va a escuchar el demonio SSH.

`HostKey` Especifica la localización del fichero con la clave privada del nodo

`ServerKeyBits` Especifica cuántos bits se van a utilizar en la clave del servidor. Estos bits se utilizan cuando el demonio empieza a generar su clave RSA.

`LoginGraceTime` Especifica el tiempo en segundos que transcurre entre que se realiza una petición al servidor y no se ha conseguido el login satisfactorio al mismo.

`KeyRegenerationInterval` Especifica un intervalo en segundos que el servidor debe esperar antes de regenerar su clave automáticamente. Es una medida de seguridad para prevenir la desencriptación de sesiones capturadas.

`PermitRootLogin` Especifica si el usuario root puede logarse utilizando ssh.

`IgnoreRhosts` Especifica si los ficheros rhosts o shosts no deben ser usados en la autenticación.

`IgnoreUserKnownHosts` Especifica si el demonio SSH debe ignorar el contenido del fichero de usuario `$HOME/.ssh/known_hosts` durante la autenticación RhostsRSAAuthentication.

`StrictModes` Especifica si SSH debe chequear los permisos de usuario en el directorio `$HOME` y el fichero rhosts antes de aceptar su entrada en el sistema.

`QuietMode` Si es sí no hace log de nada.

`X11Forwarding` Especifica si se permite X11 *forwarding* en el servidor. Tendremos que ponerlo a `yes` si deseamos ejecutar aplicaciones gráficas en una conexión ssh.

`PrintMotd` Especifica su el demonio SSH debe imprimir el contenido del fichero `/etc/motd` cuando un usuario accede al sistema de forma interactiva.

`SyslogFacility` Especifica el tipo de log de sistema que va a producir cuando se generen mensajes al sistema desde el demonio SSH.

`LogLevel` Especifica el nivel de log de sistema que es usado cuando el demonio SSH genera mensajes al sistema.

`RhostsAuthentication` Especifica si el demonio SSH puede usar la autenticación basada en rhosts.

`RhostsRSAAuthentication` Especifica si el demonio SSH puede usar la autenticación relacionada con autenticación de nodos RSA.

`RSAAuthentication` Especifica si se permite autenticación RSA.

`PasswordAuthentication` Especifica si puede utilizarse autenticación basada en usuario/clave para acceder al sistema.

`PermitEmptyPasswords` Especifica si el servidor permite la entrada en el sistema a cuentas que tienen una clave nula.

`AllowUsers` Especifica y controla qué usuarios pueden acceder a servicios SSH.

Es recomendable modificar con cuidado este fichero ya que si no tenemos acceso físico a la máquina, una configuración errónea nos dejaría sin acceso remoto a la misma. Igualmente, recordamos tambien que para que las modificaciones en este fichero tengan efecto será necesario reiniciar el servicio. La forma más cómoda de realizar esto es mediante la utilización del script de arranque del servicio que proporciona la instalación de los paquetes mencionados anteriormente[29]:

```
/etc/init.d/sshd restart
```

La parada del servicio deja activas las conexiones existentes, por lo que no tenemos que preocuparnos de dejar sin conexión a los usuarios ya conectados en el proceso de parada/arranque del servicio.

[29] En Guadalinex sustituir `sshd` por `ssh`

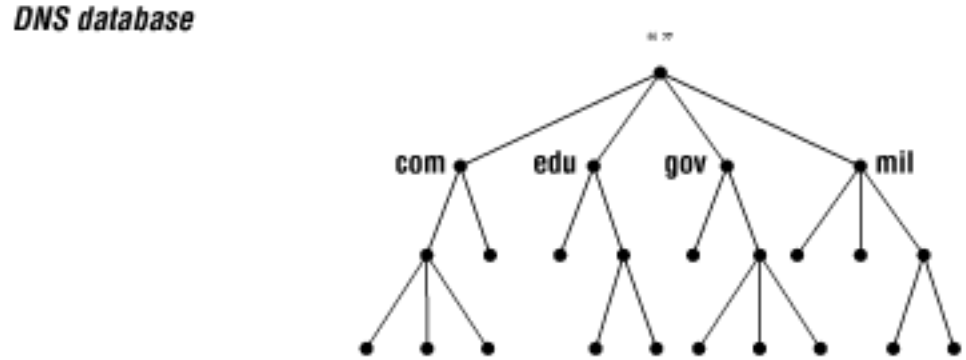
Capítulo 11

Servidor de nombres DNS

> Puede que no conozcas mucho sobre el Sistema de Nombres de Dominio (DNS) –todavía– pero siempre que usas Internet, estás usando el DNS. Cada vez que envías un correo electrónico o navegas por la Web, dependes del DNS. (*DNS and Bind,* Paul Albitz y Cricket Liu)

Llegó la hora de las direcciones simbólicas. Las direcciones IP han campado a sus anchas y la verdad es que para nosotros son difíciles de recordar y propensas a errores. Donde esté un nombre simple y descriptivo como `thales.cica.es`, que se quiten todas las direcciones IP como su equivalente 172.26.0.2 ¿o era 150.214.22.12? ¡Ah! no, es 150.214.5.10. Véis, nuestra capacidad simbólica es superior a nuestra capacidad de recordar números.

El sistema DNS es una base de datos distribuida. Presenta una jerarquía en la que su parte más alta es el "punto" o raíz y de él cuelgan los dominios de primer nivel (.com, .edu, .es, etc).



Su lectura en el orden jerárquico se realiza de derecha a izquierda. Por ejemplo, para la máquina `thales.cica.es`, primero en la jerarquía se encuentra el dominio de primer nivel[1] (.es), luego va el subdominio o subdominios (en este caso, cica) y por último el nombre de la máquina (thales).

En la figura de la página siguiente, podemos ver cómo sería la estructura jerárquica para la máquina `winnie.corp.hp.com`.

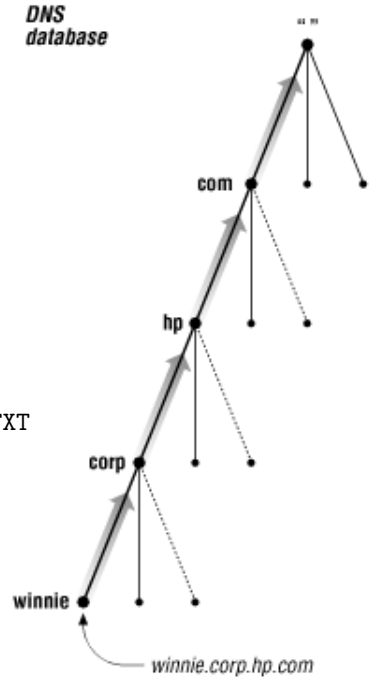
Los dominios genéricos de primer nivel son los .com, .edu, .org... más los correspondientes a los países (.es, .it, .uk, .pt,...). En Noviembre de 2000, ICANN (*Internet Corporation for Assigned Names and Numbers* `www.icann.org`) anunció la aparición de 7 nuevos dominios de primer nivel: .biz, .info, .name, .pro, .aero, .coop y .museum.

Además de estar jerarquizada, esta estructura se encuentra delegada. Veamos qué significa esto aplicándolo a nuestra dirección `thales.cica.es`.

[1] En inglés, *Top Level Domain*

ICANN es una organización sin fines de lucro que opera a nivel internacional, responsable de asignar espacio de direcciones numéricas de protocolo de Internet (IP), identificadores de protocolo y de las funciones de gestión del sistema de nombres de dominio de primer nivel genéricos (gTLD) y de códigos de países (ccTLD).

El dominio de primer nivel .es se encuentra delegado por ICANN a España, más concretamente al Organismo Red.es[2]. A su vez, Red.es delega la administración del subdominio cica al Centro Informático Científico de Andalucía, que se convierte en responsable de todo lo que cuelgue de él, y así por ejemplo, puede darle nombre (y apellidos) a la máquina thales como `thales.cica.es`.



Este sistema hace que a pesar de la distribución y delegación de responsabilidades, todo funcione con la necesaria coordinación a nivel regional y mundial.

Para profundizar en el tema y conocer más sobre el dominio .es, podéis consultar en

`http://plugindoc.mozdev.org/linux.html`

Al principio, con pocas máquinas en Internet, bastaba para mantener este sistema con unos ficheros de nombre `HOSTS.TXT` o `/etc/hosts`, en los que se encontraban los nombres de las máquinas uno a uno. A medida que el sistema fue creciendo, se hacía necesario el soporte de un sistema más potente, que es el basado en *Servidores de Nombres*.

11.1. ¿Qué necesito del DNS?

Ésta es una de las principales cuestiones a las que deberemos responder a la hora de configurar y gestionar nuestros sistemas.

La gran mayoría de vosotros[3], no necesitará montar y configurar un servidor de nombres, pero sí que los utilizaréis prácticamente en cada momento. Por ello, el comprender su funcionamiento y los recursos que ofrece es de gran ayuda.

Como vimos en la primera entrega, nuestra máquina Linux[4] necesita saber cómo resolver las direcciones simbólicas a numéricas. Ello se hacía mediante los ficheros `/etc/hosts`, `/etc/nsswitch.conf` y `/etc/resolv.conf`, o los correspondientes interfaces gráficos.

Debemos diferenciar la utilización que hacemos de los servidores de nombres del hecho de montar un servidor de nombres propio. Es algo así como la diferencia entre utilizar un procesador de textos para nuestro trabajo diario y el desarrollar un procesador de textos nosotros mismos.

11.2. Recursos del Servidor de Nombres

Para ver qué nos ofrece un servidor de nombres utilizaremos la herramienta `dig`[5]. En su forma más simple, le preguntamos como argumento con un nombre de host para conocer la dirección que le corresponde.

```
root@guadalinex:~# dig thales.cica.es
; <<>> DiG 9.2.4rc5 <<>> thales.cica.es
;; global options:  printcmd
;; Got answer:
```

[2] Anteriormente era Rediris la encargada, a través del ES-NIC.
[3] Y la gran mayoría de los mortales
[4] Y las windows también.
[5] *Domain Information Groper*, aunque puede significar "excavar". Esta herramienta sustituye a otra anterior que se llama `nslookup`.

```
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 49051
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 4
;; QUESTION SECTION:
;thales.cica.es.                        IN      A
;; ANSWER SECTION:
thales.cica.es.         172800  IN      A       150.214.5.10
;; AUTHORITY SECTION:
cica.es.                172800  IN      NS      chico.rediris.es.
cica.es.                172800  IN      NS      sun.rediris.es.
cica.es.                172800  IN      NS      dns1.cica.es.
cica.es.                172800  IN      NS      dns2.cica.es.
;; ADDITIONAL SECTION:
sun.rediris.es.         13337   IN      A       130.206.1.2
dns1.cica.es.           172800  IN      A       150.214.5.83
dns2.cica.es.           172800  IN      A       150.214.4.35
chico.rediris.es.       10872   IN      A       130.206.1.3
;; Query time: 80 msec
;; SERVER: 150.214.4.35#53(150.214.4.35)
;; WHEN: Sat Mar  5 18:54:39 2005
;; MSG SIZE  rcvd: 196
```

↬ Ésta es la salida del comando `dig`, bastante parlanchina, por cierto. La respuesta principal es la línea:

`thales.cica.es. 172800 IN A 150.214.5.10`

que nos dice que la máquina thales.cica.es tiene la dirección IP 150.214.5.10. Además, nos dice que es una dirección de tipo INternet (IN) y es un recurso de tipo A (Address). El valor 172800 es un valor de tiempo de vida (ttl) del servidor de nombres.

Además, dentro de su cortesía nos regala información adicional, como las líneas

`cica.es. 172800 IN NS sun.rediris.es.`

que nos indican cuáles son los servidores de nombres "oficiales" para la zona cica.es, que son cuatro, con el tipo de recurso NS (Name Server), también nos ofrece sus direcciones

`sun.rediris.es. 13337 IN A 130.206.1.2`

y añade el tiempo que ha tardado la consulta, a quién y cuándo. La siguiente línea

`;; SERVER: 150.214.4.35#53(150.214.4.35)`

nos dice que la consulta ha sido realizada al servidor con dirección IP 150.214.4.35 por el puerto 53, que es el que utiliza el servicio DNS. Como curiosidad, comentar que las consultas a los servidores DNS pueden realizarse tanto por TCP como por UDP.

El comando `dig` nos será de gran ayuda para consultar a los servidores de nombres. Una llamada típica al comando `dig` es de la forma:

`dig @servidor_de_nombres recurso tipo_del recurso`

donde:

servidor_de_nombres es el servidor de nombres al que vamos a preguntar. En caso de que no lo especifiquemos, preguntará a los servidores de nombres que estén en el fichero `/etc/resolv.conf`

recurso es el nombre o dirección del que queremos consultar información

tipo_del_recurso es el tipo del recurso que buscamos. Si no especificamos ninguno, buscará el tipo A por defecto.

Si el puerto del servicio de nombres (53 o domain) está cortado por nuestro proveedor de acceso o red interna, podemos utilizar un interfaz web en `http://us.mirror.menandmice.com/cgibin/DoDig`.

Un servidor de nombres nos ofrece varios tipos de recursos. Veremos a continuación los más importantes.

A (*Address*) Nos da la correspondencia de dirección simbólica a dirección IP

CNAME (*canonical name*) Nos especifica un alias o apodo para una dirección simbólica

MX (*mail exchanger*) Indica la máquina o las máquinas que recibirán el correo

NS (*name server*) Indica los servidores de nombres oficiales para el dominio

PTR (*pointer*) Nos da la resolución inversa de una dirección IP a una dirección simbólica

SOA (*start of authority*) Autoridad sobre el Dominio de nombres.

Exprimamos un poco más el comando `dig`. Le preguntaremos al servidor de nombres 150.214.5.83[6], que como vimos en el anterior comando, es un servidor de nombres oficial[7] para el dominio cica.es.

```
root@guadalinex:~/curso-linux# dig @150.214.4.35 ANY cica.e s
; <<>> DiG 9.2.4rc5 <<>> @150.214.4.35 ANY cica.es
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 51712
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 7, AUTHORITY: 0, ADDITIONAL: 6
;; QUESTION SECTION:
;cica.es.                       IN      ANY
;; ANSWER SECTION:
cica.es.                172800  IN      SOA     dns1.cica.es. hostmaster.
    cica.es . 2005022401 86400 7200 2592000 172800
cica.es.                300     IN      MX      15 smtp2.cica.es.
cica.es.                300     IN      MX      10 smtp.cica.es.
cica.es.                172800  IN      NS      sun.rediris.es.
cica.es.                172800  IN      NS      dns1.cica.es.
cica.es.                172800  IN      NS      dns2.cica.es.
cica.es.                172800  IN      NS      chico.rediris.es.
;; ADDITIONAL SECTION:
smtp.cica.es.           172800  IN      A       150.214.5.84
smtp2.cica.es.          172800  IN      A       150.214.5.100
sun.rediris.es.         12959   IN      A       130.206.1.2
dns1.cica.es.           172800  IN      A       150.214.5.83
dns2.cica.es.           172800  IN      A       150.214.4.35
chico.rediris.es.       10494   IN      A       130.206.1.3
;; Query time: 129 msec
;; SERVER: 150.214.4.35#53(150.214.4.35)
;; WHEN: Sat Mar  5 19:00:57 2005
;; MSG SIZE  rcvd: 295
```

Los registros A y NS ya nos son conocidos. Aparece el registro SOA

`cica.es. 172800 IN SOA dns1.cica.es. hostmaster.cica.es . 2005022401 86400 7200 2592000 172800`

que indica quién es la autoridad para el dominio cica.es y parámetros para los servidores de nombres que veremos más adelante.

[6] Podríamos haber puesto `dns1.cica.es`

[7] El nombre en inglés es *authoritative*

También nos encontramos con registros MX, que a pesar de tener una gran importancia no son muy conocidos[8].

```
cica.es. 300 IN MX 15 smtp2.cica.es.
cica.es. 300 IN MX 10 smtp.cica.es.
```

¿Por qué dijimos que eran muy importantes?, pues sencillamente porque dirigen los correos electrónicos. ¿Quién hoy día si le quitan el correo electrónico se quedaría igual?. Pues estos registros dicen que para todas las direcciones de correo electrónico del dominio cica.es[9], como por ejemplo *jperez@cica.es*, deben dirigirse a los "intercambiadores de correo[10]". Como es algo muy crítico, se suelen poner varios con una preferencia y en caso de fallo de alguno, los correos van al siguiente. En este caso irían preferentemente a `smtp.cica.es` y en caso de fallo de éste a `smtp2.cica.es`.

Preguntemos por un registro CNAME. El registro CNAME se suele utilizar como un alias o pseudónimo de otra u otras máquinas. ¿Qué utilidad puede tener ésto? Por ejemplo, los servicios de Internet suelen prestarse en direcciones estandarizadas. Si queremos ver el Boletín Oficial del Estado y no sabemos con certeza la dirección, una de las primeras que probaremos si tenemos cierta experiencia con internet será `www.boe.es`. Nuestra máquina con el servidor web, no tiene porqué llamarse www[11] y además nos permite cambiar rápidamente a otra máquina sin demasiados problemas en nuestra red. Veamos lo que hace el CICA.

```
root@guadalinex:~# dig CNAME www.cica.es
; <<>> DiG 9.2.4rc5 <<>> CNAME www.cica.es
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 31238
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 4
;; QUESTION SECTION:
;www.cica.es.                   IN      CNAME
;; ANSWER SECTION:
www.cica.es.            3600    IN      CNAME   ataman.cica.es.
;; AUTHORITY SECTION:
cica.es.                172800  IN      NS      chico.rediris.es.
cica.es.                172800  IN      NS      sun.rediris.es.
cica.es.                172800  IN      NS      dns1.cica.es.
cica.es.                172800  IN      NS      dns2.cica.es.
;; ADDITIONAL SECTION:
sun.rediris.es.         12495   IN      A       130.206.1.2
dns1.cica.es.           172800  IN      A       150.214.5.83
dns2.cica.es.           172800  IN      A       150.214.4.35
chico.rediris.es.       10030   IN      A       130.206.1.3
;; Query time: 80 msec
;; SERVER: 150.214.4.35#53(150.214.4.35)
;; WHEN: Sat Mar  5 19:08:41 2005
;; MSG SIZE  rcvd: 198
```

La línea importante en esta consulta es la que nos dice que `www.cica.es` es un apodo (CNAME) de la máquina `ataman.cica.es`. Si esa máquina se cae, una posible solución es cambiar el registro CNAME de `www.cica.es` a `atamon.cica.es`, que es una máquina que tenemos preparada para ello. El resto de usuarios (de todo el mundo) seguirán apuntando sus navegadores a `www.cica.es` sin enterarse del problema.

El recurso PTR es un poco más complicado. Veamos. Para que el mismo sistema funcione tanto para pedir conversiones de direcciones simbólicas a direcciones IP, como al revés, de direcciones IP a direcciones simbólicas se crea el recurso PTR y un dominio especial de nombre `in-addr.arpa`.

[8] Bueno, tú ya sé que eres un experto y sí los conoces ;-)
[9] Y de sus subdominios en caso de que no tengan especificados los suyos propios.
[10] Que eso es *Mail eXchanger*, de donde viene MX.
[11] Sería un nombre bastante feo

Un comando sencillo para saber el nombre que le corresponde a una dirección IP es el comando host

```
[root@linux images]# host 150.214.5.10
10.5.214.150.in-addr.arpa domain name pointer thales.cica.es.
```

Vemos que nos devuelve que se corresponde con la dirección simbólica `thales.cica.es`, pero antes da una información un poco rara. Como en las direcciones simbólicas la jerarquía va de derecha a izquierda y en las direcciones IP de izquierda a derecha, se emplea un truco. Todas las direcciones IP se colocan bajo el `dominio in-addr.arpa` y se va poniendo cada uno de los bytes de la dirección IP de derecha a izquierda. Así 150.214.5.10 queda como `10.5.214.150.in-addr.arpa`. Veamos qué dice nuestro amigo `dig` sobre esto:

```
root@guadalinex:~/curso-linux# dig PTR 10.5.214.150.in-addr.arpa
; <<>> DiG 9.2.4rc5 <<>> PTR 10.5.214.150.in-addr.arpa
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17607
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2
;; QUESTION SECTION:
;10.5.214.150.in-addr.arpa.      IN      PTR
;; ANSWER SECTION:
10.5.214.150.in-addr.arpa. 3600 IN      PTR     thales.cica.es.
;; AUTHORITY SECTION:
5.214.150.in-addr.arpa. 3600    IN      NS      dns2.cica.es.
5.214.150.in-addr.arpa. 3600    IN      NS      dns1.cica.es.
;; ADDITIONAL SECTION:
dns1.cica.es.           172800  IN      A       150.214.5.83
dns2.cica.es.           172800  IN      A       150.214.4.35
;; Query time: 155 msec
;; SERVER: 150.214.4.35#53(150.214.4.35)
;; WHEN: Sat Mar  5 19:15:56 2005
;; MSG SIZE  rcvd: 141
```

Correcto, es un hacha este `dig`. Nos dice que estamos hablando de `thales.cica.es` y es un registro de tipo PTR (*Poin Te R*).

11.3. Servidores de Nombres

Seguro que el DNS os ha deparado muchas sorpresas. Pues aún hay más. El hecho de configurar un Servidor de Nombres es una auténtica odisea.

El servidor de nombres por excelencia es el demonio `named`, que es parte del paquete BIND, preparado y coordinado por el *Internet Software Consortium*.

Un servidor de nombres puede estar configurado de alguna de estas formas:

master Es el "dueño" del dominio[12], en el que se hacen las modificaciones para ese dominio, responde las consultas que se le hagan y se encarga de propagarlo al resto.

slave Son servidores de nombres del dominio y así se encargan de resolver las preguntas que se les hagan. Pero cada cierto tiempo le preguntan al "master" del que dependen para actualizar su información.

caching-only Solamente constituyen un caché de datos para optimizar las respuestas[13]. Por ejemplo, podemos montar uno de este tipo en nuestro equipo u organización para que todos los puestos clientes le pregunten a él. Sirve para optimizar las respuestas y el uso de la línea

[12] Zona es el término empleado.

[13] En algunos sistemas (por ejemplo, fedora) se incluye una caché local mediante el demonio `nscd` (*name server cache daemon*).

de comunicaciones, pero además simplifica la política de seguridad. Para las peticiones de resolución DNS, los clientes no pueden atravesar el cortafuegos y sí esta única máquina.

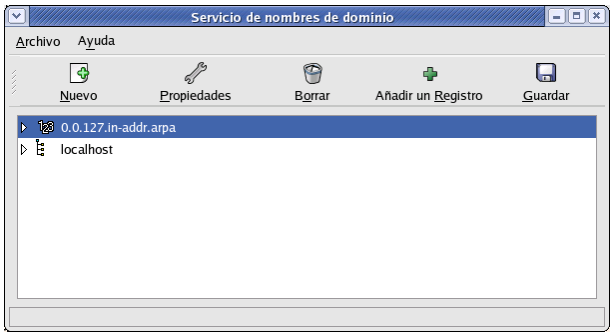
forwarding Redirige las peticiones a otros servidores de nombres. Es poca la diferencia con el de caché.

En el terreno árido, BIND guarda su configuración en los siguientes sitios:

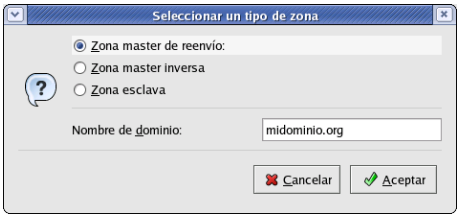
`/etc/named.conf` Fichero de configuración del demonio named.

`/var/named/` Directorio en el que almacena el resto de ficheros.

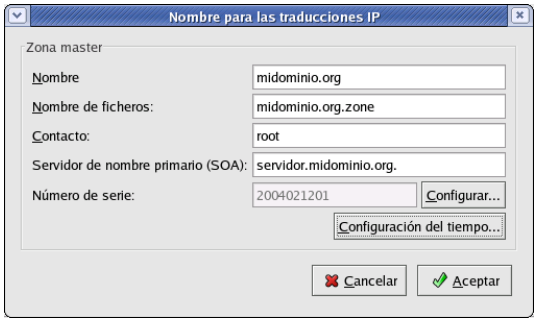
Afortunadamente, las herramientas gráficas de configuración nos son de mucha ayuda. Por ejemplo, **system-config-bind** es la herramienta gráfica de los sistemas tipo redhat y fedora.



Vemos que ya nos presenta una zona master para localhost y una zona master inversa para los registros PTR. Si deseamos crear una nueva zona, nos presenta tres opciones: una zona master, una zona master inversa o una zona esclava. Veremos el caso más completo que es el de una zona master. Crearemos el especial dominio *midominio.org*.



La herramienta ya por sí sola generará los ficheros necesarios y solamente tendremos que decirle el servidor dueño de la zona (`servidor.midominio.org.`[14]). El número de serie es un número que se utiliza cuando un master traspasa información a un servidor de nombres esclavo. El esclavo si ve que el número de serie ha cambiado, pedirá la información. Si no ha cambiado el número de serie, se supone que la información tampoco ha cambiado.



[14]Fijaos que termina en un punto. Es una forma de indicarle dónde está la raíz principal y evitar fallos como `midominio.org.midominio.org`

Una vez que hemos creado la zona para nuestro dominio, le añadiremos registros, que pueden ser de los tipos vistos anteriormente (A, CNAME, NS o mX).

En la primera figura estamos creando un registro de tipo A. La dirección simbólica `servidor.midominio.org` la asignamos a la dirección IP 192.168.12.2. Para la resolución inversa (PTR) tendremos que crear el dominio inverso `12.168.192.in-addr.arpa`.

En el segundo gráfico, añadimos un registro CNAME y creamos un alias entre las direcciones simbólicas `www.midominio.org` y `servidor.midominio.org`.

Pasemos a ver qué ha hecho la configuración gráfica sobre los ficheros. Empezamos con `/etc/named.conf`

```
[root@linux images]# more /etc/named.conf
// generated by named-bootconf.pl
//
// a caching only nameserver config
//
options {
/*
* If there is a firewall between you and nameservers you want
* to talk to, you might need to uncomment the query-source
* directive below. Previous versions of BIND always asked
* questions using port 53, but BIND 8.1 uses an unprivileged
* port by default.
*/
// query-source address * port 53;
directory "/var/named";
};
controls {
inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};
zone "." {
type hint;
file "named.ca";
};
zone "localhost" {
allow-update { none; };
type master;
file "localhost.zone";
};
zone "0.0.127.in-addr.arpa" {
allow-update { none; };
type master;
file "named.local";
};
zone "midominio.org" {
```

```
type master;
file "midominio.org.zone";
};
include "/etc/rndc.key";
```

Especifica que los ficheros de zonas y configuración adicional estará en el directorio `/var/named`. La configuración que viene por defecto crea un servidor de nombres que funciona como caché. De ahí provienen las zonas ".", `localhost` y `0.0.127.in-addr.arpa.`

Para la zona que hemos creado, `midominio.org`, especifica que es de tipo master y que el resto de la configuración se encuentra en el fichero `midominio.org.zone`, que se encontrará en el directorio... bien, has acertado: `/var/named`. Veámoslo

```
[root@linux images]# more /var/named/midominio.org.zone
$TTL 86400
servidor.midominio.org. IN SOA localhost root (
2004021207 ; serial
28800 ; refresh
14400 ; retry
3600000 ; expire
86400 ; ttl
)
servidor IN A 192.168.12.2
.midominio.org IN MX 1 servidor.midominio.org.
```

La autoridad para el dominio (SOA) es `servidor.midominio.org`, vemos que el serial es 2004021207. Normalmente, por convención se pone en formato año, mes, día y modificación dentro del día. El resto de valores son el tiempo en segundos, en que se refresca la información a los esclavos, que se reintenta en caso de no poder conectar, tiempo de expiración y máximo tiempo que lo pueden tener las cachés.

Hemos creado un registro tipo A que une las direcciones `servidor.midominio.org` y 192.168.12.2 y también un registro MX que indica que el correo dirigido al dominio `midominio.org`, será recogido por el servidor `servidor.midominio.org`.

En el fichero `named.local` podemos observar una típica zona de registros inversos tipo PTR.

```
[root@linux images]# more /var/named/named.local
$TTL 86400
@ IN SOA localhost. root.localhost. (
1997022703 ; serial
28800 ; refresh
14400 ; retry
3600000 ; expire
86400 ; ttl
)
@ IN NS localhost
1 IN PTR localhost.
2 IN PTR servidor.midominio.org.
```

Capítulo 12

Servicio de Directorio LDAP

> El problema de nombrar y direccionar entidades no admite una única solución. En el fondo, el problema es éste: la entidad 1 debe conocer el nombre de la entidad 2 para intercambiar datos con ella, ¿pero cómo puede obtener ese nombre a menos que ya lo conozca? (William Stallings, *Data and Computer Communications*)

En los albores de Internet, la ISO (*International Standards Organization*) comprendió la necesidad de tener un sistema de directorio en el que poder tener de forma ordenada y organizada información sobre una organización, ya fuera esta información sobre personas, máquinas o servicios. Así surgió X.500, un sistema de Directorio potente y muy completo. Pero que adolecía de los problemas de la mayoría de los protocolos ISO: su complejidad y dificultad de implantación.

Hasta hace poco tiempo, los sistemas de Internet han carecido de sistemas de directorio, pero en el momento actual, en organizaciones de tamaño mediano y grande se necesita de las facilidades que incorporan estos sistemas. Incluso en redes pequeñas también aportan muchas ventajas. Por ejemplo, los nombres de usuario y palabras de paso se pueden controlar de forma centralizada en un directorio y ser utilizadas por todos los sistemas. ¿No os gustaría como usuarios tener una sola password para todos los sistemas? ¿Y como administradores de una red por pequeña que sea, no os gustaría controlar las passwords desde un solo sitio para todos los usuarios?. Pues añadámosle a la coctelera las direcciones y nombres de las máquinas, los certificados digitales de los usuarios, teléfonos y direcciones electrónicas, etc. Resultado: ¿cómo hemos podido vivir hasta ahora sin los directorios?. Bueno, es un poco exagerado, pero los directorios han venido para quedarse e integrarse en nuestras aplicaciones.

Un sistema de directorio puede ser utilizado para almacenar un amplio rango de datos: dirección de correo electrónico, números de teléfono, ubicación, claves públicas de seguridad, listas de contactos, y prácticamente lo que se nos ocurra o necesitemos. El directorio LDAP se convierte en un punto de integración de información del organismo o empresa.

El Protocolo de Acceso Ligero a Directorio, más conocido como LDAP[1], está basado en el estándar X.500, pero es significativamente más simple y adaptado a TCP/IP.

Hasta aquí, alguien podría pensar que para esto ya tenemos las bases de datos. En cierto modo un directorio es una base de datos, pero con unas características especiales:

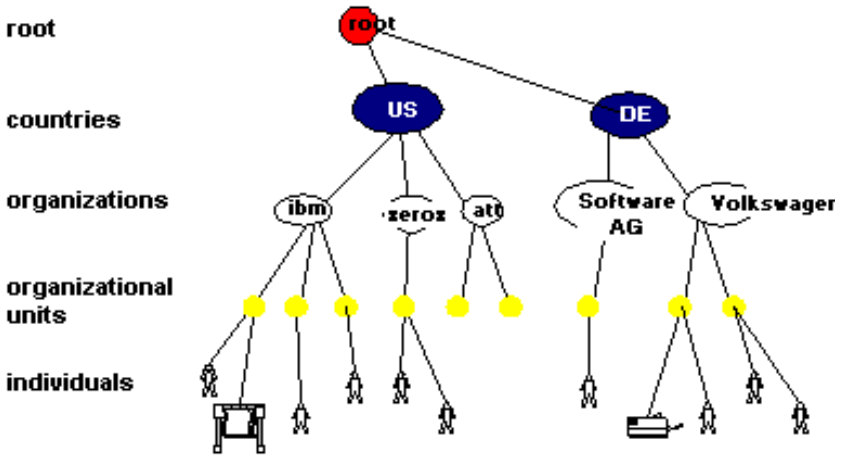
- Datos relativamente estáticos. Los datos son modificados con poca frecuencia. ¿Cada cuánto tiempo cambiamos de número de teléfono o de correo electrónico?
- Las operaciones de lectura (recuperación) deben ser muy rápidas. El directorio está optimizado para lecturas frecuentes y concurrentes.
- Distribuido. Los datos se ubican físicamente en varios sistemas de la red aportando redundancia, altas prestaciones y escalabilidad

[1] Siglas en inglés de *Lightweight Directory Access Protocol*

- Jerárquico. Asegura una gestión descentralizada y delegada de la información, de forma similar a como vimos con el sistema DNS.
- Orientado a Objetos. El directorio representa objetos que pertenecen a clases. Las clases son una colección de atributos, no necesariamente normalizados como en una base de datos tradicional.
- Esquema estándar, disponible para todas las aplicaciones que usan el directorio.
- Atributos multivaluados. Los atributos del directorio pueden tener un único o múltiples valores.
- Replicación Multi-master. A diferencia de las bases de datos tradicionales, los directorios se distribuyen por la red. Si un sistema no está disponible, el cliente accede a otra réplica de la información.

12.1. Estructura del Directorio

En un directorio LDAP, las entradas se disponen en una estructura jerárquica en forma de árbol. Por ejemplo, esta estructura puede reflejar componentes geográficos u organizacionales. Los países pueden estar en la parte superior del árbol bajo la raíz. Debajo de cada país los estados o comunidades autónomas. Bajo cada organización puede haber más unidades organizacionales y bajo ellas, personas, impresoras, documentos o lo que podamos necesitar.



La forma de almacenar información en un directorio es en base a entradas (*entries*). Una *entrada* es una colección de atributos que tiene un identificador único llamado *Distinguished Name* (DN). Mediante el DN podemos referirnos a la entrada de forma no ambigua.

Una entrada pertenece a una clase de objeto (*objectclass*), que define el conjunto de atributos que puede tener. Cada uno de los atributos de una entrada es de un tipo[2] y puede poseer cero, uno o más valores. Los tipos se denominan con nombres que nos dan una idea de su contenido. Por ejemplo, *cn* para nombre común[3], *o* para Organización, *c* para país[4] o *mail* para dirección de correo electrónico.

Algunos atributos son obligatorios para la clase, mientras que otros son opcionales. Una definición de la clase de objeto (*objectclass*) determina qué atributos son obligatorios y cuáles no para cada una de las entradas que pertenecen a esa clase. Las definiciones de clases de objetos se

[2] O clase
[3] *Common Name* en inglés
[4] *Country*

encuentran en varios ficheros de esquema, que se encuentran en el directorio `/etc/openldap/schema/`[5].

El valor de un atributo depende del tipo que posea. Un atributo de tipo cn puede tener el valor "Manuel Pérez Pérez", y un atributo de tipo mail puede tener el valor "manuel.perez@midominio.org".

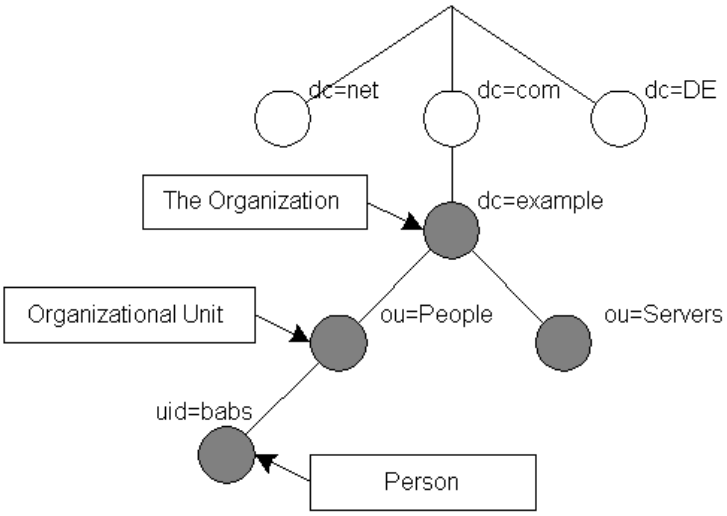
A continuación presentamos el contenido de una entrada:

```
dn: o=Sociedad Matemática Thales, c=ES
o: Sociedad Matemática Thales
objectclass: organization
```

El dn es el identificador de la entrada, que puede ser visto como su clave o su nombre completo, que debe ser único en el directorio. En este caso nos indica que es una organización de nombre "Sociedad Matemática Thales", que pertenece al país España.

La entrada *objectclass* nos dice que es de un tipo denominado organization, que permitirá, según su definición, que la entrada pueda tener unos determinados atributos. Esta definición de los tipos se realiza en el esquema (*schema*).

La tendencia actual es a nombrar la estructura basándose en los nombres de dominio de Internet. En el árbol siguiente podemos ver un ejemplo.



El DN de una entrada se construye cogiendo el nombre de la entrada en el árbol[6] y concatenando los nombres de cada entrada superior hasta llegar a la raíz del árbol. En la figura anterior, la persona con `uid=babs`, que es su RDN, tendrá un DN de `uid=babs`, `ou=People`, `dc=example`, `dc=com`. Significa que el identificador de usuario babs, es de una persona porque es su tipo, perteneciente a la unidad organizativa[7] *Personas*[8], de la Organización *example*, dentro del dominio de primer nivel *.com*.

El servicio de directorio LDAP se basa en un modelo cliente-servidor. Uno o más servidores LDAP contienen los datos que forman el Árbol de Información del Directorio (DIT[9]). Los clientes se conectan a un servidor y le realizan una consulta. El servidor responde con la respuesta o un puntero a dónde encontrarla si él no la tiene.

[5] En un sistema RedHat o Fedora.
[6] llamado el RDN, de *Relative Distinguished Name*
[7] *Organizational Unit*
[8] La otra rama indica los servidores
[9] *directory information tree*

12.2. Servidor OpenLDAP

El servidor OpenLDAP es una implementación en Software Libre del protocolo LDAP, proporcionando tanto los servidores, como clientes básicos para trabajar con ellos y librerías para enlazar con otras aplicaciones.

En un sistema Fedora, nos encontramos con los siguientes paquetes:

openldap Contiene las librerías que necesitan tanto el servidor OpenLDAP como las aplicaciones clientes.

openldap-clients Contiene clientes básicos de línea de comandos para consultar y modificar la información que almacena el servidor LDAP.

openldap-servers Contiene los servidores OpenLDAP y utilidades para su configuración.

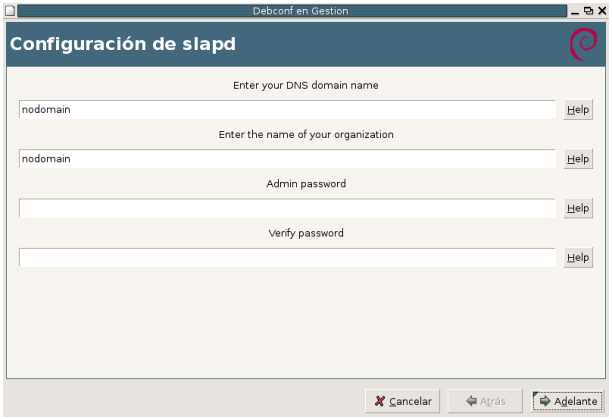
En el paquete de servidores realmente vienen dos servidores, el propiamente dicho demonio de LDAP (`/usr/sbin/slapd`) y el demonio de replicación (`/usr/sbin/slurpd`). El demonio de sincronización `slurpd` se utiliza para sincronizar cambios de un servidor LDAP a otros que hayamos definido. Para nuestros propósitos no necesitaremos el demonio de replicación.

En el caso de tener una distribución basada en debian como el caso de Guadalinex2004 debemos realizar la instalación de los paquetes

slapd Contiene los servidores ldap (tanto slapd como slurpd)

ldap-utils Contiene herramientas de cliente básicas.

A la hora de realizar la instalación nos aparece una serie de pantallas de configuración que podemos rellenar o cancelar y definirlo posteriormente en el fichero de configuración `/etc/ldap/slapd.conf`



Por defecto el servidor ldap se ejecuta como root, si deseamos cambiar el usuario debemos modificar los parámetros SLAPD_USER y SLAPD_GROUP del fichero de configuración `/etc/default/slapd` para que se ejecute el demonio con la opción `-u usuario_definido`

12.2.1. Configuración del Servidor OpenLDAP

La configuración se encuentra en el fichero `/etc/openldap/slapd.conf` o en `/etc/ldap/slapd.conf` dependiendo de la distribución. Dentro de la configuración existen múltiples opciones, muchas de ellas dependerán de los tipos de clientes y otras de la finalidad que tenga el ldap. En principio recomendamos evitar la versión 2 del protocolo por facilitar la configuración, así como el *backend* `ldbm`. Trabajaremos con los ficheros de fedora, que son similares a los de Guadalinex salvo en la ubicación.

Veamos su contenido más interesante.

`include /etc/openldap/schema/core.schema`
`include /etc/openldap/schema/cosine.schema`
`include /etc/openldap/schema/inetorgperson.schema`
Incluye los esquemas que podemos utilizar para los tipos de entradas.
`database ldbm`
Tipo de base de datos que utiliza para almacenar su contenido.
`suffix "dc=midominio,dc=org"`
Definimos la estructura principal de nuestro directorio. En este caso para una organización cuyo dominio en Internet es midominio.org.
`rootdn "cn=Manager,dc=midominio,dc=org"`
Definimos quién será el administrador del directorio
`rootpw {SSHA}X1XTJTGJvKseb+AXnX/XY8iHqxqO3EPV`
La contraseña del administrador del directorio. Podemos ponerla aquí en texto claro (`rootpw secreto`) pero es preferible ponerla cifrada[10].
`directory /var/lib/ldap`
Directorio donde se va a almacenar la información del directorio.
Ya estamos listos para arrancar el servidor de directorio.

Debian `#/etc/init.d/sldap restart`

Fedora `#service ldap restart`

Comprobamos que ya funciona, aunque todavía no tengamos datos en el directorio.

```
#ldapsearch -x -b " -s base '(objectclass=*)' namingContexts
  # extended LDIF
  # LDAPv3
  # base <> with scope base
  # filter: (objectclass=*)
  # requesting: namingContexts
  dn:
  namingContexts: dc=midominio,dc=org
  # search result
  search: 2
  result: 0 Success
  # numResponses: 2
  # numEntries: 1
```

La consulta no es normal, ya que estamos preguntando a un directorio vacío. La opción

`-x` es utilizar autenticación normal,

`-b` indica la base de búsqueda,

`-s` indica el ámbito de la búsqueda y nos contentamos con cualquier cosa que nos devuelva (`objectclass=*`).

Al menos vemos que el servidor está funcionando y nos devuelve algo.

Ya tenemos nuestro servidor ldap funcionando con una estructura organizativa `dc=midominio`, `dc=org`. Pero el directorio está vacío. Tenemos que alimentarlo de entradas.

El formato **LDIF** es el estándar para representar entradas del directorio en formato texto. Una entrada del directorio consiste en dos partes. El DN o nombre distinguido, que debe figurar en la

[10] Para obtener la clave cifrada, ejecutamos:
```
# slappasswd
New password:
Re-enter new password:
{SSHA}X1XTJTGJvKseb+AXnX/XY8iHqxqO3EPV
```

primera línea de la entrada y que se compone de la cadena dn: seguida del DN de la entrada. La segunda parte son los atributos de la entrada. Cada atributo se compone de un nombre de atributo, seguido del carácter dos puntos, :, y el valor del atributo. Si hay atributos multievaluados, deben ponerse seguidos.

No hay ningún orden preestablecido para la colocación de los atributos, pero es conveniente listar primero el atributo `objectclass`, para mejorar la legibilidad de la entrada.

El DN debe estar en ASCII (el código ASCII puro es de 128 caracteres). Para conseguirlo seguimos estos criterios: cambiamos una vocal acentuada por la misma vocal no acentuada sustituimos la letra ñ por n y transformamos la letra ü en u.

```
dn: uid=Fernando G.,ou=AIT,o=midominio,c=org
objectclass:person
objectclass:organizationalPerson
objectclass:inetOrgPerson
objectclass:posixAccount
objectclass:top
uid: Fernando G.
sn: Gordillo
cn: Fernando Gordillo
description: usuario de ejemplo
loginshell: /bin/sh
uidnumber:502
gidnumber:1000
mail: fernandogs@midominio.org
telephonenumber: 55555555
homedirectory: /home/fernando
```

Las líneas excesivamente largas pueden partirse con un retorno de carro y añadiendo un espacio al principio de la siguiente línea.

Si un atributo contiene valores no ASCII, como por ejemplo una imagen JPEG, se codifica en formato Base64

El formato LDIF también puede ser utilizado para realizar actualizaciones y/o borrar entradas del directorio. El formato en este caso contiene en la primera línea el DN de la entrada sobre la que se aplica el cambio. La segunda línea indica el cambio a realizar y las siguientes líneas contienen los pares atributo-valor que componen el cambio.

Para añadir una entrada

```
Dn: nombre distinguido
Changetype: add
Tipo_atributo: valor
```

Para borrar una entrada basta indicar el cambio delete

```
Dn: nombre distinguido Changetype: delete
```

Para modificar una entrada

```
Dn: nombre distinguido Changetype: modify
TipoCambio: atributo atributo: valor
```

Varias operaciones se pueden combinar en un único fichero si las separamos por un guión. Veamos un fichero en este formato que nos servirá para iniciar nuestro servidor ldap en un primer momento:

```
dn: dc=midominio,dc=org
objectclass: dcObject
objectclass: organization
```

```
o: Organismo Ejemplo
dc: midominio

dn: cn=Manager,dc=midominio,dc=org
objectclass: organizationalRole
cn: Manager
```

Es importante la línea en blanco porque separa dos entradas del directorio. La primera es la de la organización en sí, con DN `dc=midominio,dc=org`. Las dos entradas `objectclass` nos indican a qué tipos pertenece la entrada: pertenece al tipo organization y al tipo `dcObject`. En este caso incluye un atributo que es `o` (organización), para introducir el nombre de ésta. Si queremos ver qué atributos permiten dichas clases, deberemos consultarlos en el directorio de esquemas `/etc/openldap/schema`.

La otra entrada es para el administrador del directorio, el *Manager*.

Insertemos en el directorio el contenido de este fichero. Utilizaremos el comando `ldapadd`.

```
#ldapadd -x -D "cn=Manager,dc=midominio,dc=org" -W -f entrada1.ldif
Enter LDAP Password:
adding new entry "dc=midominio,dc=org"
adding new entry "cn=Manager,dc=midominio,dc=org"
```

Las opciones con las que lo invocamos son:

-x para utilizar autenticación simple

-D más el DN del usuario con el que nos conectamos, que es el Manager

-W para que nos pregunte la clave, en vez de ponerla en la línea de comandos y que alguien pueda verla

-f para indicarle el fichero, que será `entrada1.ldif`

Si todo ha funcionado correctamente, como es el caso del comando anterior, ya tenemos dos entradas en nuestro directorio. Podemos utilizar cualquier cliente LDAP para consultarlas.

Un cliente básico de línea de comandos es `ldapsearch`. Lo utilizaremos para hacer esta consulta inicial. Con la opción `-b` le indicamos la base de búsqueda. Si no se la indicamos, no sabe por dónde hacer la búsqueda y no nos devolvería resultados.

Además no nos ponemos muy exigentes y le decimos que nos devuelva cualquier cosa que encuentre.

```
#ldapsearch -x -b 'dc=midominio,dc=org' '(objectclass=*)'
  # extended LDIF
  # LDAPv3
  # base <dc=midominio,dc=org> with scope sub
  # filter: (objectclass=*)
  # requesting: ALL
  # midominio.org
  dn: dc=midominio,dc=org
  objectClass: dcObject
  objectClass: organization
  o: Organismo Ejemplo
  dc: midominio
  # Manager, midominio.org
  dn: cn=Manager,dc=midominio,dc=org
  objectClass: organizationalRole
  cn: Manager
```

Ha encontrado las dos entradas que existen en el directorio.

Hay una forma de especificar por defecto la base de búsqueda para los clientes, que es en el fichero `/etc/openldap/ldap.conf o /etc/ldap/ldap.conf`

```
HOST 127.0.0.1
BASE dc=midominio,dc=org
```

o

```
BASE dc=midominio,dc=org
URI ldap://ldap.example.com:389
```

Le estamos diciendo a qué servidor se tiene que conectar el cliente si no le especificamos otro y cuál será la base de búsqueda por defecto.

Seamos un poco más exigentes y alimentemos un fichero LDIF con empleados o funcionarios de nuestra organización.

```
root@guada04:~# more entrada3.ldif
dn: cn=Juan Lopez Perez, dc=midominio,dc=org
objectClass: person
objectClass: inetOrgPerson
mail: juan.lopez@midominio.org
telephoneNumber: +34-954-55-55-55
sn: Lopez
cn: Juan Lopez Perez
cn: Juan Lopez

dn: cn=Laura Jimenez Lora, dc=midominio,dc=org
objectClass: person
objectClass: inetOrgPerson
mail: laura.jimenez@midominio.org
telephoneNumber: +34-959-59-59-59
sn: Jimenez
cn: Laura Jimenez Lora
cn: Laura
```

Si realizamos una nueva búsqueda, ahora será mucho más rica:

```
root@guada04:~# ldapsearch -x -
b 'dc=midominio,dc=org' '(objectclass=*)'
# extended LDIF
# LDAPv3
# base <> with scope sub
# filter: (objectclass=*)
# requesting: ALL
# midominio.org
dn: dc=midominio,dc=org
dc: midominio
objectClass: dcObject
objectClass: organization
o: Organismo Ejemplo
# Manager, midominio.org
dn: cn=Manager,dc=midominio,dc=org
objectClass: organizationalRole
cn: Manager
# Juan Perez Perez, midominio.org
dn: cn=Juan Perez Perez,dc=midominio,dc=org
telephoneNumber: +34-950-50-50-50
```

```
mail: juan.perez@midominio.org
objectClass: person
objectClass: inetOrgPerson
sn: Lopez
cn: Juan Perez Perez
cn: Juan Perez
```

Las herramientas administrativas con que nos encontramos son:

slapadd Añade entradas de un fichero LDIF a un directorio LDAP directory.

slapcat Recupera entradas de un directorio LDAP y las almacena en un formato LDIF.

slapindex Reindexa el contenido del directorio.

slappasswd Genera una palabra de paso cifrada.

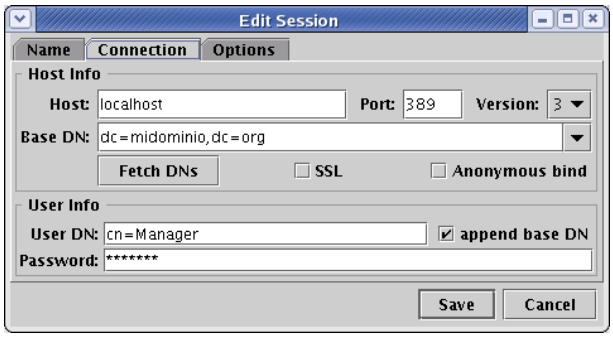
Sin embargo, recomendamos el cliente `ldapbrowser` que veremos a continuación para la modificación, importación y exportación de los datos del directorio.

12.3. Clientes LDAP

Tenemos una gran cantidad de clientes que pueden conectarse a nuestro servidor de directorio y además, cada uno puede utilizarlo para un propósito distinto. Un cliente de correo electrónico puede utilizarlo para buscar direcciones o el certificado del destinatario para enviarle el correo cifrado. Un sistema de proxy-caché puede autenticar al usuario basándose en el directorio para permitirle navegar por Internet. Además, estos clientes pueden ser clientes windows, Linux o de cualquier otro sistema. Solamente tienen que cumplir las reglas del protocolo LDAP.

Un cliente muy versátil que nos permitirá consultar y modificar el directorio es LDAP Browser/Editor. Es una herramienta construida en Java que podemos obtener de `http://www.iit.edu/~gawojar/ldap`.

Nos descargamos el fichero `Browser282b2.tar.gz`[11], lo descomprimimos y ejecutamos en el directorio `ldapbrowser` el fichero `lbe.sh`. Es necesario que tengamos una máquina virtual de Java instalada en nuestro sistema. Si no es así podemos descargarnos de `http://java.sun.com` el *Java Runtime Environment*, por ejemplo el paquete `j2re-1_4_2_03-linux-i586.rpm`.

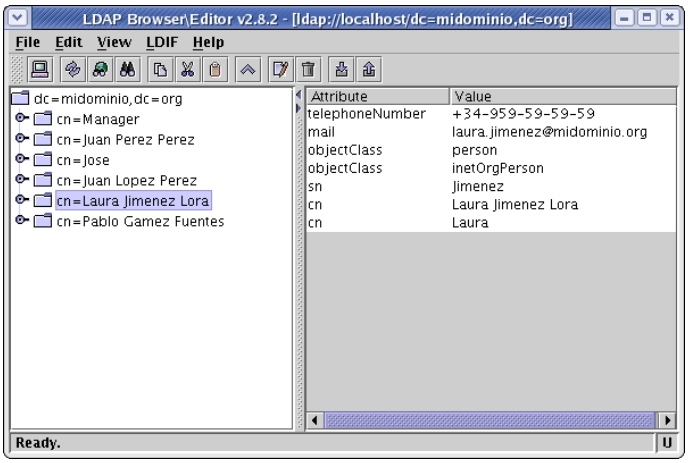


Para definir nuestra nueva conexión, debemos darle el host (localhost en este caso), el puerto (el 389 es el estándar del protocolo), la versión de protocolo (la 3 es la actual), la base de búsqueda (`dc=midominio,dc=org`) y si hacemos un acceso anónimo marcamos el cuadro [**Anonymous bind**].

Si queremos entrar como un usuario del directorio, en este caso utilizaremos el Manager, y podremos modificar entradas y atributos. Marcamos la opción de que le añada la base de búsqueda. Quedaría

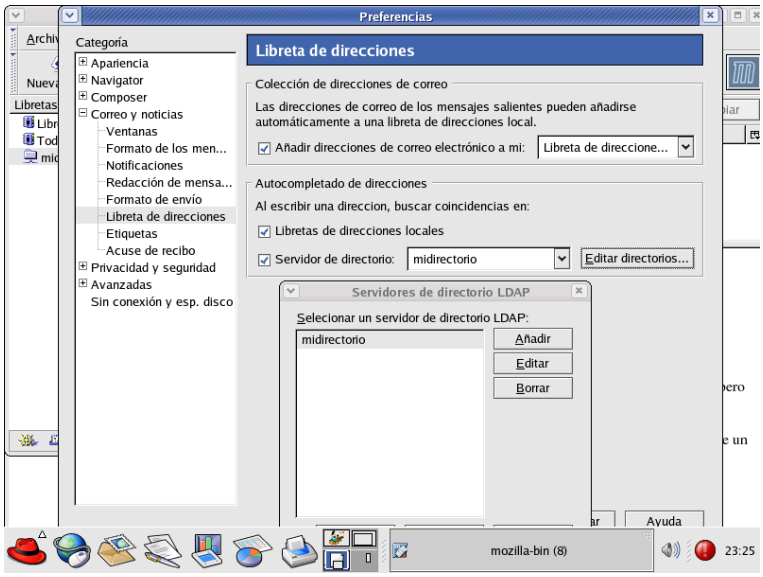
`cn=Manager,dc=midominio,dc=org`, pero hemos escrito menos ;-).

[11] Para cualquier sistema operativo, ya que está hecho en Java



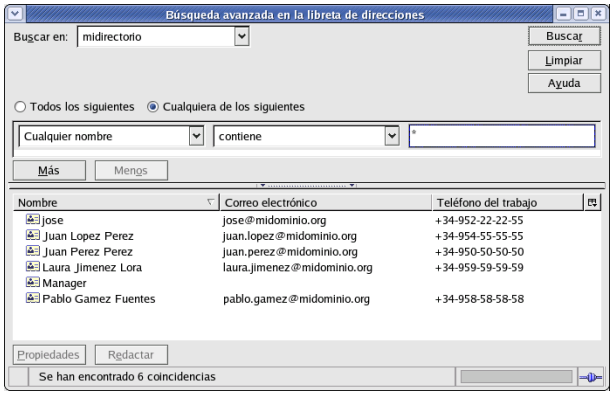
¿Queda bonito verdad? Pues además es fácil de manejar y es gratis. Se cumple lo de bueno, bonito y barato. Es una estupenda herramienta de consulta y modificación de directorios LDAP.

Veamos además un cliente de correo electrónico. Utilizaremos el correo electrónico de Mozilla. Para decirle a la libreta de direcciones que utilice el directorio, seleccionamos **Editar→Preferencias→Correo y Noticias→Libreta de Direcciones**. En la sección de Autocompletado de direcciones, seleccionamos **Servidor de Directorio** y editamos para crear uno nuevo.



En la ventana de **Propiedades del servidor de directorios**, especificamos un nombre, el nombre del servidor al que nos tenemos que conectar (hostname), la base de búsqueda (`dc=midominio,dc=org`) y el puerto (389 por defecto). Tendremos opción de entrar como usuario autenticado o de usar conexión segura mediante SSL.

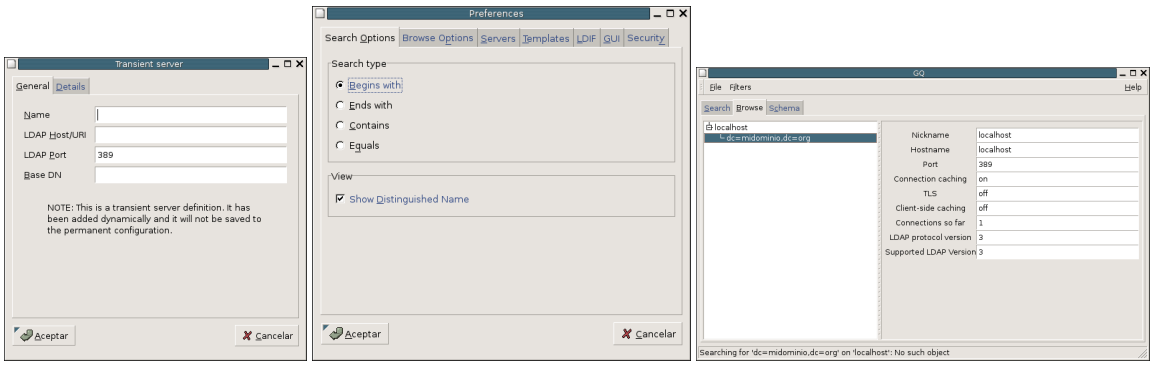
Una vez configurado, podemos utilizar el directorio para buscar personas, sus direcciones, teléfonos o los atributos que hayamos alimentado en el directorio.



Los servicios de directorio tenderán a concentrar cada vez más funciones y a convertirse en el alma de la mayoría de las organizaciones.

Otro cliente que puede utilizarse de forma rápida en una máquina Linux es **GQ**. Este paquete se instala de forma sencilla con `apt-get install gq` o `rpm -Uhv gq-version.sistema.rpm` y nos permite conectarnos a un directorio ldap y gestionarlo de forma gráfica.

Desde este programa una vez configuradas la preferencias y dentro de ellas los parámetros de conexión al servidor, podemos realizar búsquedas, navegar en el directorio y ver el schema.



El formato básico de los filtros de búsqueda de LDAP es el siguiente:

```
Atributo operador valor
```

El atributo se refiere al atributo sobre el que vamos a realizar la operación de comparación. El operador puede ser uno de los siguientes:

= Devuelve las entradas cuyo atributo tiene el valor especificado.

>= Devuelve las entradas cuyo atributo sea mayor o igual que el valor especificado

<= Devuelve las entradas cuyo atributo sea menor o igual que el valor especificado

=* Devuelve las entradas que tienen valor asignado en el atributo especificado

~= Devuelve las entradas cuyo atributo tenga un valor similar al especificado

Operadores de filtros

El carácter * tiene el significado de cualquier valor y puede ser empleado con el operador =. Pero además, los operadores de búsquedas pueden combinarse utilizando los operadores booleanos, dando lugar a expresiones de búsqueda más complejas. La sintaxis para combinar filtros de búsquedas es la siguiente:

```
(operador (filtro1) (filtro2) (filtro3) ? )
(operador (filtro))
```

Los operadores son:

- & : AND lógico de los filtros.
- | : OR lógico de los filtros.
- ! : NOT lógico del filtro.

Algunos ejemplos de filtros:

- `(& (uid=*2202) (cn=Fernando*))` Busca entradas cuyo campo uid termine en 2202 y cuyo campo cn empiece por Fernando.
- `(| (cn=Fernando) (cn=Irene))` Busca entradas cuyo campo cn sea Fernando
- `(! (cn=Emilio))` Busca todas las entradas cuyo campo cn no sea Emilio.

Para realizar búsquedas sobre atributos cuyos valores contienen alguno de los caracteres reservados para la construcción de los filtros, deben utilizarse secuencias de escape.

12.4. Caso práctico: Autenticación mediante directorio LDAP

Acabamos de ver cómo podemos tener una base de datos para almacenar información acerca de los usuarios de nuestro sistema. Ya comentamos en el comienzo de este capítulo que es posible utilizar el servicio de directorio LDAP para almacenar de forma centralizada la clave de acceso de los usuarios.

Usando una base de datos centralizada de información de usuarios se simplifica enormemente la gestión de claves. Supongamos el siguiente escenario de trabajo, que no tiene nada de extraordinario, en el que gestionamos un servidor de correo y un servidor web. Cada usuario tendrá una cuenta de correo y un espacio, respectivamente, en los servidores anteriores. Sin otra infraestructura adicional, cada vez que un usuario quiera cambiar la contraseña de un sistema deberá cambiarla en el otro. Si queremos que se utilice la misma contraseña en los dos sistemas tendremos que crear un mecanismo de replicación, con el consiguiente trabajo adicional.

Mediante un directorio LDAP podemos tener un repositorio común al que accederán los sistemas de autenticación de los servidores que sean configurados de esta forma. En esta sección, basándonos en lo visto anteriormente, daremos contenido a un directorio LDAP y posteriormente configuraremos el sistema para que lo utilice como base de datos de autenticación.

12.4.1. Crear un directorio para autenticación

En primer lugar vamos a modificar un poco el esquema anterior. Los siguientes ficheros en formato `ldif` nos definirán el esqueleto de los datos que utilizaremos. Primero crearemos la raíz de nuestro árbol LDAP:

```
dn: dc=midominio,dc=org
objectclass: dcObject
objectclass: organization
o: Organismo Ejemplo
dc: midominio
dn: cn=Manager,dc=midominio,dc=org
objectclass: organizationalRole
cn: Manager
```

A continuación crearemos dentro de nuestro árbol una serie de grupos que nos permiten tener una estructura más organizada dentro del directorio.

```
dn: ou=grupos, dc=midominio, dc=org
objectclass: top
objectclass: organizationalUnit
ou: grupos
dn: ou=personas, dc=midominio, dc=org
objectclass: top
objectclass: organizationalUnit
ou: personas
dn: cn=usuarios, ou=grupos, dc=midominio, dc=org
objectclass: top
objectclass: posixGroup
cn: usuarios
gidnumber: 2000
```

Aparecen otros objetos que no aparecieron anteriormente como son `organizationalUnit` y `posixGroup`. Su función es crear ramas hijas de la raíz que nos permitirán distribuir nuestros usuarios de una forma más ordenada. Inicialmente nuestros usuarios colgarán de la rama *personas*.

```
dn: cn=Juan Lopez Perez, ou=personas, dc=midominio, dc=org
objectclass: person
objectclass: inetOrgPerson
mail: juan.lopez@midominio.org
telephoneNumber: +34-954-55-55-55
sn: Lopez
cn: Juan Lopez Perez
cn: Juan Lopez
dn: cn=Laura Jimenez Lora, ou=personas, dc=midominio, dc= org
objectclass: person
objectclass: inetOrgPerson
mail: laura.jimenez@midominio.org
telephonenumber: +34-954-59-59-59
sn: Jimenez
cn: Laura Jimenez Lora
cn: Laura
dn: cn=Jose Fernandez Diaz, ou=personas, dc=midominio, dc=org
objectclass: person
objectclass: inetOrgPerson
objectclass: posixAccount
mail: jose.fernandez@midominio.org
telephoneNumber: +34-954-56-56-56
sn: Diaz
```

```
cn: Jose Fernandez Diaz
cn: Jose Fernandez
uid: jose.fernandez
userPassword: hola
gecos: Jose Fernandez Lopez
uidNumber: 2000
gidNumber: 2000
homeDirectory: /home/jose.fernandez
loginShell:/bin/bash
```

Los usuarios que definimos anteriormente no presentan variación, únicamente los hemos situado dentro de la rama *personas*, pero el nuevo usuario que introducimos tiene en su definición una nueva clase de objeto, fundamental para nuestro objetivo. La clase `posixAccount` tiene como atributos los necesarios para almacenar los datos de una cuenta de sistema. Los campos que utilizamos en este usuario nos recuerdan a los que se almacenan en el fichero `/etc/passwd` y serán los que utilizaremos para la definición de las cuentas de nuestros sistemas en el directorio LDAP.

`uid` Define el identificador de usuario en el sistema

`userPassword` Define la clave correspondiente al uid

`gecos` Almacena la descripción de la cuenta de usuario

`uidNumber` Almacena el número de identificación del usuario

`gidNumber` Almacena el número de grupo del usuario

`homeDirectory` Almacena la ruta del directorio $HOME

`loginShell` Almacena la ruta completa de la shell del usuario

12.4.2. Configuración de Name Service Switch

La gestión de los permisos de los ficheros del sistema se hace a partir los número que definen el uid (*user identification*) y gid (*group identification*). Para simplificar su identificación se le asignan nombres a los usuarios y a los grupos de forma que podamos identificarlos de forma más sencilla[12].

Para que el resto de utilidades del sistema sepan a qué nombres de usuarios y grupos pertenecen los ficheros se realizan una serie de llamadas a las funciones de la librería glibc. Estas llamadas darán como resultado la relación uid/gid nombre identificativo[13].

En el fichero `/etc/nsswitch.conf` se indica al sistema dónde debe buscar el propietario de un fichero o directorio a partir de su uid. Las fuentes para las "bases de datos" y su orden de búsqueda se especificarán en este fichero.

```
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the 'glibc-doc' and 'info' packages installed, try:
# 'info libc "Name Service Switch"' for information about this file.
passwd:         compat
group:          compat
shadow:         compat
hosts:          files dns
networks:       files
protocols:      db files
services:       db files
```

[12]Es más fácil identificar un fichero o directorio como perteneciente al usuario `legolas` que al `uid 2546`.. Lo mismo ocurre con los grupos.

[13]También pueden gestionarse otras relaciones como las referidas a dirección IP y nombre de `host`

```
ethers:         db files
rpc:            db files
netgroup:       nis
```

Modificaremos las primeras entradas de este fichero de la siguiente forma:

```
passwd:         compat ldap
group:          compat ldap
shadow:         compat ldap
```

De esta forma cuando un programa solicite "el nombre de usuario con uid 2345" se buscará primero en el fichero `/etc/passwd` y posteriormente en el directorio LDAP definido en `/etc/ldap/ldap.conf`.

En otras distribuciones podremos tener los parámetros **files ldap**

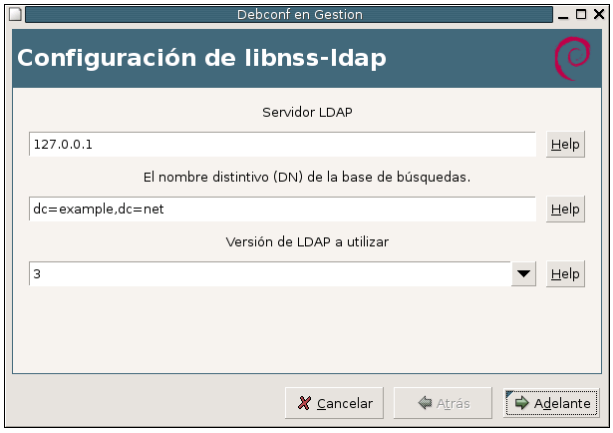
Para realizar esta función es necesario tener instalada la librería `libnss-ldap` que ejercerá de cliente para dar soporte LDAP al servicio NSS (*Name Service Switch*).

Con Guadalinex:

`libnss-ldap`

El fichero de configuración es `/etc/libnss-ldap.conf`, donde estableceremos los parámetros necesarios para realizar la autenticación

A la hora de la instalación en Guadalinex nos aparecerá un gestor de configuración en el que debemos introducir los datos del directorio, decir si queremos que se exija contraseña para consultar, si sólo el propietario podrá escribir y los datos del usuario de consulta que deberá tener los suficienes permisos.



Una vez terminada la preconfiguración podemos encontrar un ejemplo de `nsswitch.conf` en `/usr/share/doc/libnss-ldap/examples/nsswitch.ldap`. Además todos los parámetros del asistente pueden modificarse y deben repasarse en `/etc/libnss-ldap.conf` . Donde también se definen las ramas del directorio donde consultar cada uno de los elementos (passwd,shadow,...) y los parámetros de pam para la autenticación.

`uri ldap://ldap.midominio.org/ ldap://ldap-copia.midominio.org/`

`base dc=midominio, dc=org`

`nss-ldap` espera que las cuentas sean objetos con los siguientes atributos: uid, uidNumber, gidNumber, homeDirectory, y loginShell. Estos atributos son permitidas por el objectClass posixAccount.

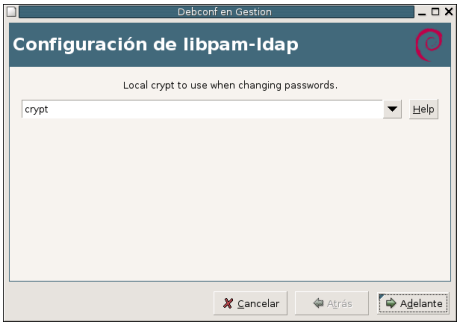
12.4.3. Módulos PAM

Lo siguiente será configurar los módulos PAM (*Pluggable Authentication Modules*). Para poder configurar el cliente PAM tendremos que instalar el paquete `libpam-ldap`

Guadalinex

`libpam-ldap`

En esta instalación de la misma forma que las anteriores se utiliza el asistente de configuración para los parámetros más importantes (administrador, password, tipo de cifrado,...)



Este asistente lo que configura son los ficheros `/etc/pam_ldap.conf` y `/etc/ldap.secret`.

Hay varios programas que pueden usar un método de autentificación "centralizado" usando los módulos PAM. Son unas librerías que sirven de interfaz contra varios métodos de autentificación (en nuestro caso LDAP).

Estos módulos son unas librerías que sirven como interfaz entre los programas y distintos métodos de autenticación. En nuestro caso los configuraremos para la autenticación contra LDAP.

La configuración de estos módulos en Guadalinex se realiza en el directorio `/etc/pam.d` y tenemos un fichero de configuración por cada servicio.

`uri ldap://ldap.midominio.org/`

`base dc=midominio,dc=org`

`pam_password [exop/crypt]`

Las directivas uri y base trabajan del mismo modo que en `/etc/libnss_ldap.conf` y `/etc/ldap/ldap.conf`.

Si es necesario que la conexión sea con privilegios, se usará la contraseña almacenada en `/etc/ldap.secret`.

Vemos algunos ejemplos.

Consideremos primeramente una distribución genérica con el programa login, el cual maneja el login desde una consola de texto. Una pila típica de PAM, que chequea las contraseñas, tanto en `/etc/passwd` como en la base de datos LDAP. En el fichero `/etc/pam.d/login`:

```
auth required pam_nologin.so
auth sufficient pam_ldap.so
auth sufficient pam_unix.so shadow use_first_pass
auth requred pam_deny.so
```

Para aquellas aplicaciones que utilicen las contraseñas en el fichero `/etc/pam.d/passwd`

```
password    required      pam_cracklib.so
password    sufficient    pam_ldap.so
password    sufficient    pam_unix.so
password    required      pam_deny.so
```

En el caso de una distribución Debian como Guadalinex:

/etc/pam.d/common-auth:

```
auth sufficient pam_ldap.so
auth required pam_unix.so nullok_secure try_first_pass
```

/etc/pam.d/common-account:

```
account sufficient pam_ldap.so
account required pam_unix.so
```

/etc/pam.d/common-password:

```
password sufficient pam_ldap.so
password required pam_unix.so nullok obscure min=4 max=8 md5
```

Una herramienta importante para utilizar ldap para autenticación es el paquete de Migrationtools. Este paquete, instalable mediante `apt-get`, contiene una serie de script en perl que nos permiten una vez configurado el fichero `/etc/migrationtools/migrate_common.ph` convertir nuestros ficheros de usuarios a ficheros en formatos ldif para cargarlos en el ldap.

```
#> cd /usr/share/migrationtools
#> ./migrate_group.pl /etc/group /tmp/group.ldif
#> ./migrate_passwd.pl /etc/passwd |grep -
v 'objectClass: account' > /tmp/passwd.ldif
```

12.4.4. nscd

Aunque a continuación se explique el demonio nscd, no se recomienda su utilización, ya que puede haber problemas al autenticarse porque dicho servicio deje bloqueada la máquina. Es conveniente, probar en la distribución y versiones que estemos utilizando antes de ponerlo en producción.

Para evitar que sea consultado el servidor LDAP cada vez que es ejecutado un comando como `ls -l` dentro de nuestra organización, es una buena idea configurar en nuestras estaciones de trabajo un sistema de cache para algunos datos de usuario. Mientras los datos en la cache sean lo suficiente recientes, las estaciones de trabajo utilizarán estos en vez de preguntar al servidor LDAP otra vez. El demonio servidor de caché de nombres (nscd) cumple exactamente esta tarea.

Para instalar nscd:

```
# apt-get install nscd
```

El fichero de configuración de nscd es `/etc/nscd.conf`.

```
/etc/nscd.conf
enable-cache passwd yes
positive-tive-to-live passwd 600
negative-time-to-live passwd 20
suggested-size passwd 211
check-files passwd yes
```

Capítulo 13

Compartir impresoras:Cups

> Durante mucho tiempo, la configuración de impresoras ha sido uno de los dolores de cabeza para los administradores de linux y UNIX. (*El Libro Oficial de Red Hat Linux: Guía del administrador*)

13.1. Introducción

En Linux cuando queremos imprimir un trabajo el sistema lo envía a la impresora en lenguaje Postscript, se trata de un lenguaje que le dice a la impresora cómo tiene que imprimir la página. Aquí es donde pueden surgir problemas dependiendo del tipo de impresora que tengamos:

- impresoras Postscript: la impresora interpreta directamente las páginas enviadas por el sistema y no vamos a tener ningún problema con ellas. La única "pega" suele ser su precio, ya que la mayoría de ellas son de gama alta.
- impresoras que tienen su propio lenguaje: como no interpretan el lenguaje Postscript, es el sistema el que tiene que traducir la salida Postscript al lenguaje propio de la impresora. Si los fabricantes lo han dado a conocer no suele haber problemas con ellas (algunas empresas proporcionan controladores libres para sus modelos) y están soportadas bajo Linux. Este tipo de impresoras suelen ser las más frecuentes por su bajo precio.
- Winimpresoras: utilizan drivers específicos de Windows para interpretar el lenguaje Postscript, son las más difíciles (¡o imposibles!) de configurar bajo Linux.

CUPS (*Common Unix Printing System*) es un servidor de impresión pensado para gestionar una red en la que una o varias impresoras tienen que dar servicio a todos los equipos interconectados por esa red. Con él disponemos de un sistema de impresión portable y estándar (IPP/1.1[1]) dentro del mundo GNULinux.

IPP define un protocolo estándar para impresoras y para el control de los servicios de impresión. Como todos los protocolos basados en IP, IPP se puede usar tanto en una red local como en una intranet. A diferencia de otros protocolos, soporta control de acceso, autentificación y encriptación, proporcionando soluciones más idóneas y seguras que con los antiguos protocolos.

Características más destacadas de CUPS:

- Posee un interfaz web para poder configurar el servidor
- Cuotas y administración de trabajos y páginas.
- Detección automática de impresoras de red.

[1] *Internet Printing Protocol.*

- Servicios de directorio de impresoras en red.
- Soporta control de acceso, autentificación y encriptación.

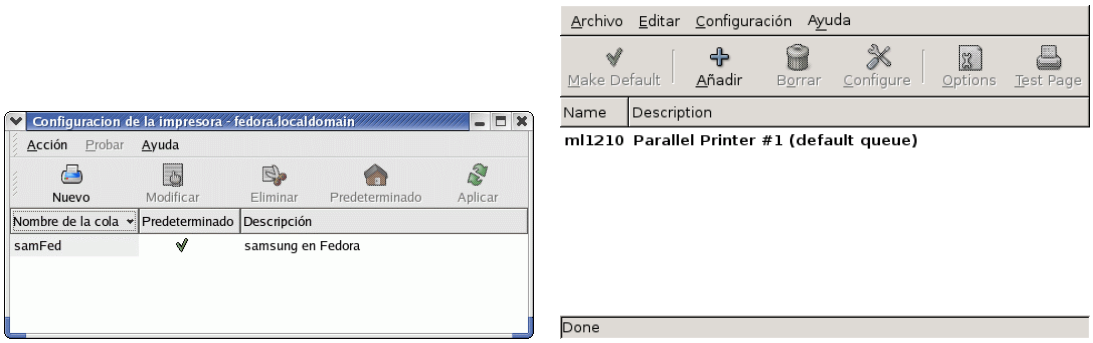
Sitios Web de utilidad

- GNU/Linux Printing `http://www.linuxprinting.org` todo lo que necesitamos conocer sobre la posibilidad de configurar nuestra impresora con Linux
- `http://www.cups.org/` Web de CUPS.

Tanto Fedora como Guadalinex disponen de herramientas gráficas de configuración de la impresora. No vamos a trabajar con ninguna de ellas ya que no suponen una ventaja sustancial sobre el método general que vamos a estudiar. Las herramientas gráficas son:

Fedora **system-config-printer**[2]. Si bien su uso es simple y es una herramienta potente, tenemos que tener en cuenta que sobreescribe los cambios que realicemos "a mano" en el fichero de configuración de CUPS.

Debian **foomatic-gui**[3]



(a) system-config-printer (b) foomatic-gui

Figura 13.1: Herramientas gráficas de configuración

Hemos optado por el método más general, consistente en el interfaz Web de CUPS y el fichero de configuración del servidor.

13.2. Instalación

En ambos sistemas supondremos que la instalación se hace con conexión a Internet. Si no es así, la única diferencia estriba en bajar los paquetes y, una vez que están en nuestra máquina, usar `rpm -ivh paquete` (para Fedora) o `dpkg -i paquete` (para Debian).

[2]En modo texto `system-config-printer-tui`

[3]La aplicación gráfica por excelencia para configurar las impresoras, `gnome-cups-manager`, tiene un "bug" que impide añadir impresoras. Ese problema no se resuelve (por ahora) aunque actualicemos el sistema.

13.2.1. Fedora

Se instala por defecto, aunque siempre es recomendable que comprobemos si tenemos instalada la última versión del paquete con[4]:

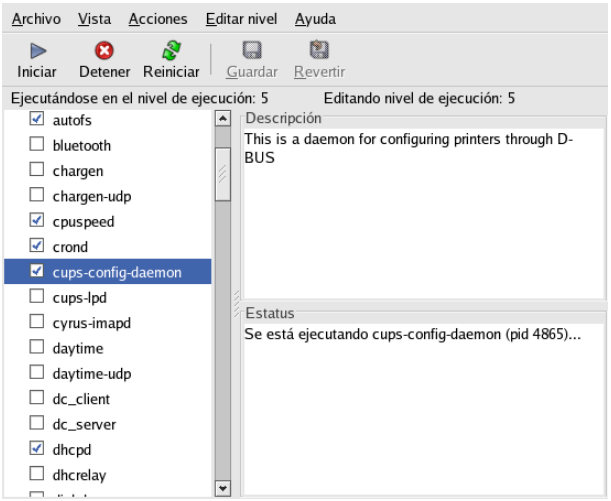
```
# apt-get update; apt-get install cups cups-libs
```

Para reiniciar el servicio podemos usar:

```
#service cups restart
```

O en modo gráfico:

```
#system-config-services
```



Una vez instalado, además de las páginas man del programa es obligado visitar el directorio:

```
/usr/share/doc/cups-x.x.x/
```

13.2.2. GuadaLinex

Si bien debe de estar instalado, lo mejor es que actualicemos a la última versión disponible:

```
# apt-get install cupsys
```

y que instalemos también el paquete[5]

```
# apt-get install cupsys-client
```

cupsys Servidor de impresión CUPS

cupsys-bsd comandos BSD para CUPS

cupsys-client programa cliente de CUPS

Para reiniciar el servicio

```
# /etc/init.d/cupsys start
```

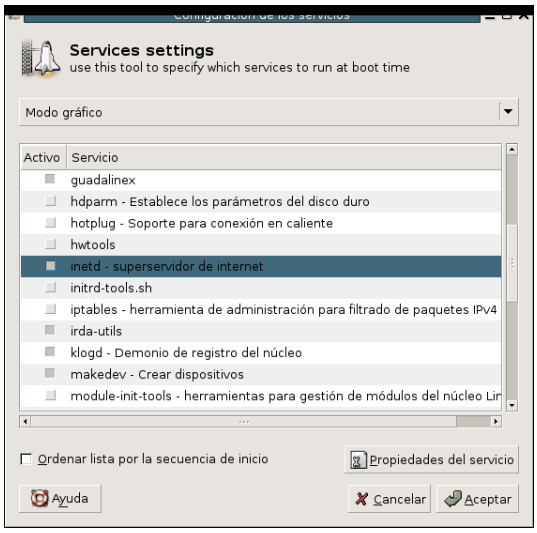
o en modo gráfico lo podemos hacer con

```
# services-admin
```

[4] O con

```
# yum install cups
```

[5] No es imprescindible. Además, deberámos garantizarnos de que los paquetes `foomatic-filter` (filtros usados por las colas de impresión para convertir los datos de entrada PostScript en el formato nativo que usa una impresora en concreto) y `foomatic-db` (base de datos con filtros de impresora) estuviesen también actualizados.



- Una vez instalado, además de las páginas man del programa es obligado visitar los subdirectorios:

  `/usr/share/doc/cups*`

- Por defecto sólo permite que imprimamos desde la propia máquina. Tendremos que adecuar el fichero de configuración (lo vemos en un par de páginas) para que nos permita imprimir desde la red.

13.3. Configuración de CUPS

Para entender cómo funciona CUPS necesitamos introducir algunos conceptos:

Cola de impresión lista de trabajos pendientes de imprimir. Se imprime el primero que llega (como en una cola de un cine, el primero que se pone en la fila es el primero que saca la entrada)

Clases se trata de una abstracción sobre la idea de impresora. Una clase (Dpto de matemáticas, Dpto de Inglés, ...) puede estar compuesta por varias impresoras. Si mandamos un trabajo a una clase, CUPS se encarga de repartir el trabajo de forma óptima entre las impresoras que componen esa clase.

Filtros reglas que usa CUPS para traducir los trabajos a imprimir al modelo concreto de impresora.

Siguiendo un orden cronológico, para imprimir un trabajo lo haremos con `lpr`[6], que lo manda a un directorio de spool. Del directorio de spool lo coge el demonio `cupsd` que lo enviará a la impresora física correspondiente, pasándole el filtro adecuado. Si no lo puede mandar inmediatamente a la impresora, lo dejará en el directorio de spool en espera de que llegue su turno o la impresora esté preparada.

Los ficheros de configuración de CUPS se localizan en `/etc/cups`. Los ficheros estándar de configuración son:

client.conf opciones de configuración para los clientes de impresión.

[6] O los comandos o iconos gráficos que a su vez llaman a éste

cupsd.conf permite configurar el demonio de impresión de CUPS (`/usr/sbin/cupsd`)

Además de los dos ficheros anteriores hay otros. En general estos últimos no se modifican "a mano" ya que su contenido se obtiene a partir del interfaz Web (o del comando `lpadmin`). Los más importantes y que aparecen de forma estándar en ambas distribuciones son:

classes.conf en él se almacena la información sobre las clases de impresión.

mime.convs lista de programas conversores a usar para convertir de un tipo MIME a otro

mime.types le dice a CUPS cómo reconocer un tipo de dato a partir de números mágicos dentro del archivo

printers.conf archivo de configuración de las impresoras

13.3.1. client.conf

En este fichero configuramos las opciones de las máquinas clientes. Las opciones en este fichero en general, no deberíamos cambiarlas.

```
#ServerName myhost.domain.com
```

CUPS puede configurarse para que los trabajos pendientes de imprimir se almacenen en la máquina servidor y que no se use el directorio de spool local. Si lo que deseamos es que los trabajos pendientes se almacenen sólo en el servidor, debemos activar la directiva nombre del servidor, en ella pondríamos el nombre de nuestro servidor de impresión (si es una red local lo normal es que aquí esté la IP local del servidor de impresión). En general, es mejor dejarlo todo como está ya que una caída del servidor acarrearía la pérdida de los trabajos de impresión.

Con la directiva `Encryption`, podemos optar por el tipo de encriptación a usar, los valores posibles son:

Always usar siempre encriptación (SSL)

Never no usar nunca encriptación

Required usa actualización de encriptación TLS

IfRequested usa encriptación si el servidor lo pide, es la directiva por defecto.

```
#Encryption Always
#Encryption Never
#Encryption Required
#Encryption IfRequested
```

13.3.2. cupsd.conf

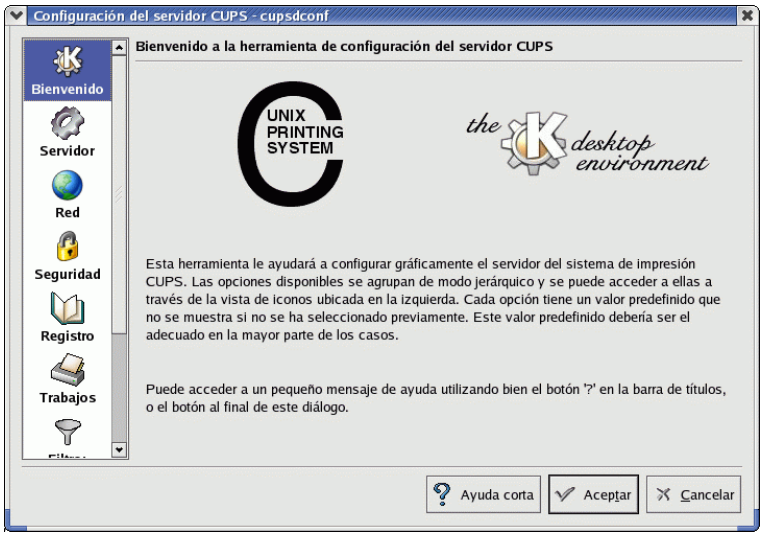
Las entradas de este fichero son sencillas de manipular[7] y entender (se parecen a las del fichero de configuración de Apache).

Con ellas podemos configurar, además de aspectos generales del servidor, el registro, las rutas de los directorios del servidor, el cifrado y soporte de certificados, ... En general no hay que modificarlas para disponer del servicio de impresión. Sólo puede ser necesario hacerlo para permitir accesos vía red y en este caso nos limitaremos a añadir las entradas adecuadas al final del fichero (véanse los ejemplos en la página 216)

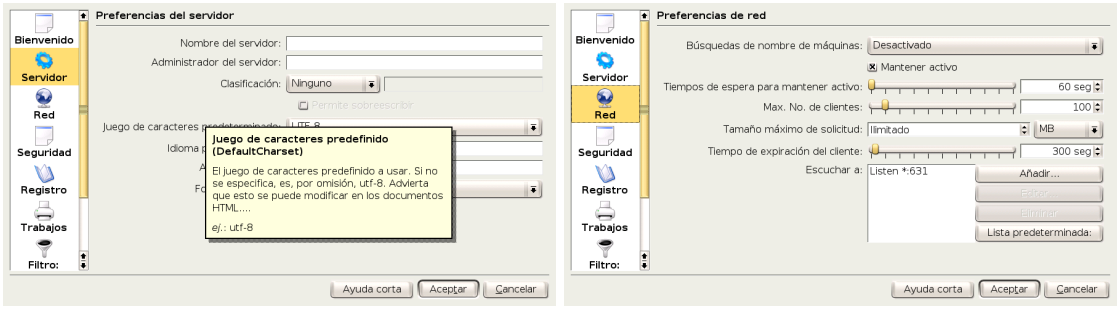
[7] Si se va a modificar es conveniente crear antes una copia del original por si "metemos la pata " en algo.
`#cp cups.conf cups.conf.original`

cupsdconf

Si hemos instalado el KDE, podemos usar la herramienta gráfica de configuración de CUPS[8]
`#cupsdconf`



para ajustar nuestro fichero `/etc/cups/cupsd.conf`. Si bien no soporta todas las opciones de configuración[9] sí que permite configurar las más usuales. La ayuda corta que acompaña al programa es muy buena y hace que usar esta herramienta sea un "juego de niños".



Desde ella podemos configurar fácilmente el demonio de impresión, pero para explicar las distintas opciones hemos optado por el método más general, es decir, usar el fichero de configuración. Es un buen ejercicio usar la herramienta gráfica anterior y después revisar qué directivas de configuración son las que hemos escrito en el fichero `/etc/cups/cupsd.conf`

El fichero

La mayoría de las directivas están muy bien comentadas en el propio fichero. Estudiemos el contenido del archivo[10], hemos mantenido el `#` en aquellas directivas que aparecen comentadas

[8] En general debe estar a nuestra disposición en ambas distribuciones, si no es así, los instalamos con:

Debian `# apt-get install kdelibs-bin`

Fedora `# apt-get install kdelibs`

[9] Si detecta una directiva no soportada, nos avisa de ello y de que no permitirá su modificación.

[10]

- Para Fedora, en el de Debian cambian un poco los valores por defecto, no las directivas de configuración.
- Está claro que no hay que conocerlas ni todas ni de memoria -:). El incluirlo completo es con la idea de que dispongáis de una referencia rápida de las posibilidades de configuración de CUPS

por defecto, los valores por defecto los escribiremos entre paréntesis y además, para simplificar su lectura, mantendremos la organización por secciones:

`Server Identity` En esta sección podemos configurar el nombre del servidor, por defecto se usará el nombre del sistema. También podemos determinar la cuenta de correo a la que se envíen las quejas cuando los clientes tengan problemas con la impresión.

```
#ServerName myhost.domain.com
#ServerAdmin root@your.domain.com
```

`Server Options` En primer lugar configuramos el fichero en que se almecenarán los `logs` del sistema, si bien aparece comentado se toma como valor por defecto. En él se almacenan los datos en el llamado "Formato de registro común", esto significa que podemos usar cualquier herramienta de informe de registro de acceso para generar informes de la actividad del servidor (por ejemplo Webalizer[11]).

```
#AccessLog /var/log/cups/access_log
```

Las directivas que siguen (por defecto están inhabilitatadas) permiten que en los trabajos de impresión aparezca el texto que sigue a `Classification` junto con páginas informativas sobre el trabajo de impresión (identidad del trabajo, usuario, etc).

```
#Classification classified
#Classification confidential
#Classification secret
#Classification topsecret
#Classification unclassified
```

Si deseamos permitir que los usuarios puedan modificar los parámetros de la directiva anterior (`Classification`) pondremos la directiva que sigue en `on` (`off`). En este caso podrán limitar la primera y última página que se añade al trabajo, así como cambiar el tipo de clasificación.

```
#ClassifyOverride off
```

Directorio de datos de CUPS, codificación (por defecto UTF-8) y lenguaje por defecto.

```
#DataDir /usr/share/cups
#DefaultCharset utf-8
#DefaultLanguage en
```

Está claro que una primera modicación consiste en adecuar el lenguaje por defecto a `es`.

Directorio donde se localiza la ayuda del programa[12].

```
#DocumentRoot /usr/share/doc/cups-1.1.22
```

Archivo de registro de errores y permisos del fichero en donde se almacenan, tal cual está sólo el root podrá ver su contenido.

```
#ErrorLog /var/log/cups/error_log
LogFilePerm 0600
```

Para permitir imprimir sobre un archivo pondremos la directiva que sigue en `Yes` (por defecto es `No` para limitar problemas de seguridad), de esa forma se admitirán URIs de la forma `file:/tmp/print.prn`. Si no se tiene claro es mejor dejarlo en `No`.

[11] Se estudia en entregas posteriores.

[12] En Guadalinex es `/usr/share/cups/doc-root`

```
#FileDevice No
```

Especifica la ruta por defecto de los archivos de fuentes.

```
#FontPath /usr/share/cups/fonts
```

Nivel de registro de errores, por defecto `info`. Puede ser:

`debug2`	Registra todo.	`warn`	Registro de errores y avisos.
`debug`	Registra casi todo.	`error`	Sólo registra errores.
`info`	Registra las peticiones y los cambios de estado.	`none`	No registra nada.

En general está bien así y sólo en el caso de que tengamos problemas que no sabemos resolver podemos optar por `debug2`. En ese caso el fichero de registro de errores se puede hacer enorme[13]. Con `MaxLogSize` establecemos el límite del tamaño, en bytes, de los archivos de registro de errores (si se comenta toma de valor por defecto `1048576=1MB`) antes de que sean rotados (si optamos por poner 0 desactivamos la rotación).

```
LogLevel info
MaxLogSize 200000000
```

Archivo en dónde almacenar el registro de páginas[14], en cada línea de ese fichero se almacena el nombre de la impresora, el de usuario, los IDs de los trabajos, fecha de impresión, número de páginas del trabajo, y número de copias de cada página.

↬ Por ejemplo, el trabajo con ID 284 tenía 4 páginas y 1 copia impresa, el trabajo con ID 285 tenía 1 copia de sólo 1 página.

```
# tail -f /var/log/cups/page_log
lp0 paco 284 [21/Feb/2004:16:11:14 +0100] 3 1
lp0 paco 284 [21/Feb/2004:16:11:14 +0100] 4 1
lp0 paco 285 [21/Feb/2004:17:09:11 +0100] 1 1
```

```
#PageLog /var/log/cups/page_log
```

Las tres opciones que siguen nos permiten:

- preservar el historial de un trabajo (`Yes`) después de finalizar
- preservar los archivos de un trabajo (`No`)
- purgar (con el sistema de cuotas activo) automáticamente los trabajos completados que no se necesitan más (`No`)

```
#PreserveJobHistory Yes
#PreserveJobFiles No
#AutoPurgeJobs No
```

Las directivas que siguen nos permiten:

[13] Parar poder ver qué está pasando podemos ejecutar:
`#tail -f /var/log/cups/error_log`

[14] Para que se calcule correctamente el número de páginas el trabajo ha de pasar a través del filtro `pstops`. Si el trabajo se ha pasado a través de una cola "en bruto", el número de páginas no se contabilizará bien (en general, aparecen como 1 trabajo de 1 página con múltiples copias).

- Número máximo de copias que un usuario puede soliciar (100)
- Número máximo de trabajos que se mantienen almacenados (500), una vez que el número de trabajos alcanza el límite se borran los más antiguos. Si los trabajos están aún sin finalizar se rechaza el trabajo nuevo. Si optamos por el valor 0 no hay límite.
- Número máximo de trabajos activos para cada impresora o clase (por defecto 0).
- Número máximo de trabajos por usuario (0)
- Si se pone a cero se desactiva. Controla el número máximo de colecciones de históricos del atributo `printer-state-history`

```
#MaxCopies 100
#MaxJobs 500
#MaxJobsPerPrinter 0
#MaxJobsPerUser 0
#MaxPrinterHistory 10
```

El nombre del archivo *printcap*. Es necesario mantenerlo activo para que funcionen bien ciertas aplicaciones antiguas que precisan un archivo de ese tipo. Con la directiva siguiente optamos por seleccionar el formato de este archivo, por defecto BSD

```
Printcap /etc/printcap
#PrintcapFormat BSD
#PrintcapFormat Solaris
```

Con las 4 directivas que siguen determinamos:

- Directorio donde se almacenan las solicitudes de impresión
- Nombre de usuario que se asigna para accesos no autentificados desde sistemas remotos. Por defecto es remroot. Este nombre aparecerá en archivos de registro y solicitudes para todos aquellos recursos y direcciones del servidor que permiten acceso sin autentificación. Las entradas autentificadas contendrán los nombres autentificados.
- Directorio para los ejecutables del servidor de impresión, por defecto `/usr/lib/cups`
- Directorio que contiene los archivos de configuración (`/etc/cups`)

```
#RequestRoot /var/spool/cups
#RemoteRoot remroot
#ServerBin /usr/lib/cups
#ServerRoot /etc/cups
```

Con la directiva `ServerTokens` especificamos qué información se devuelve en el encabezado de las peticiones HTTP, por defecto es `Minor`.

```
# ServerTokens None
# ServerTokens ProductOnly       CUPS
# ServerTokens Major             CUPS/1
# ServerTokens Minor             CUPS/1.1
# ServerTokens Minimal           CUPS/1.1.22rc1
# ServerTokens OS                CUPS/1.1.22rc1 (uname)
# ServerTokens Full              CUPS/1.1.22rc1 (uname) IPP/1.1
```

`Fax Support` Con las directivas de este apartado establecemos el número de veces que se reintenta un trabajo de fax (5) así como el intervalo de tiempo (300 segundos) entre reintentos.

```
#FaxRetryLimit 5
#FaxRetryInterval 300
```

`Encryption Support` Si usamos comunicaciones seguras, ficheros que contienen el certificado y la clave del servidor (respectivamente)

```
#ServerCertificate /etc/cups/ssl/server.crt
#ServerKey /etc/cups/ssl/server.key
```

`Filter Options` Establecemos el usuario y grupo bajo los que se ejecuta el servidor de impresióm (lo usual es dejar los valores por defecto: `lp` y `sys` respectivamente).

```
#User lp
#Group sys
```

Cantidad de memoria que cada RIP (*Raster Image Processo*r) debe utilizar para usar el caché de mapa de bits (por defecto 8m). El valor puede ser un número seguido de

- k para kilobytes
- m para megabytes
- g para gigabytes
- t para "baldosas", donde 1t=256 x 256 pixels.

```
#RIPCache 8m
```

Directorio para archivos temporales

```
#TempDir /var/spool/cups/tmp
```

Establece el coste máximo de todos los filtros de los trabajos que se están ejecutando al mismo tiempo. Un trabajo medio de una impresora no PostScript necesita un límite de filtro de al menos 200 (la mitad si es PostScript). Límites inferiores al mínimo requerido por un trabajo hacen que sólo se imprima un trabajo cada vez. El valor por defecto es 0 (ilimitado).

```
#FilterLimit 0
```

`Network Options` Las directivas de esta sección nos permiten configurar las opciones de red del servidor. Con `Port` establecemos el puerto en que escucha el servidor (631). La directiva `Listen` nos permite establecer la dirección en la que escuchar. Podemos tener varias entradas para cada una de ellas.

```
# Port 80
# Port 631
# Listen hostname
# Listen hostname:80
# Listen hostname:631
# Listen 1.2.3.4
# Listen 1.2.3.4:631
#Port 80
#Port 443
Listen 127.0.0.1:631
```

Nos permite optar por trabajar con nombres de máquina totalmente cualificados. Para mejorar las prestaciones por defecto está en `Off`

```
#HostNameLookups On
```

Con esta directiva y la siguiente obligamos a que el servidor se mantenga activo 60 segundos en espera de servir nuevos clientes desde la misma conexión.

```
#KeepAlive On
#KeepAliveTimeout 60
```

Las cuatro directivas que siguen permiten determinar:

- Número máximo de clientes a la vez (100)
- Número máximo de clientes para un host determinado, por defecto 0: significa 1/10 del número máximo de clientes.
- Tamaño máximo del trabajo de impresión (0, significa ilimitado)
- Tiempo de espera (300 segundos) antes de que expiren las peticiones

```
#MaxClients 100
#MaxClientsPerHost 0
#MaxRequestSize 0
#Timeout 300
```

`Browsing Options` Por defecto se difunden las impresoras a las máquinas de la red, usando el protocolo CUPS. Con `BrowseAddress` podemos establecer la dirección de multidifusión.

```
Browsing On
BrowseProtocols cups
#BrowseAddress x.y.z.255
#BrowseAddress x.y.255.255
#BrowseAddress x.255.255.255
#BrowseAddress 255.255.255.255
#BrowseAddress @LOCAL
#BrowseAddress @IF(name)
```

Con la opción siguiente (`Yes`) optamos por usar nombres cortos para impresoras remotas (`impresora` en lugar de `impresora@host`)

```
#BrowseShortNames Yes
```

Podemos permitir o denegar la exploración usando las dos directivas que siguen. Por defecto se permite la entrada a paquetes de cualquier dirección local y no se deniega ningún paquete.

```
BrowseAllow from @LOCAL
#BrowseDeny address
```

Los parámetros permitidos son[15]:

[15] Para que podamos usar nombres de máquina o dominios es necesario que esté activada la resolución de nombres.

```
All
None
*.domain.com
.domain.com
host.domain.com
nnn.*
nnn.nnn.*
nnn.nnn.nnn.*
nnn.nnn.nnn.nnn
nnn.nnn.nnn.nnn/mm
nnn.nnn.nnn.nnn/mmm.mmm.mmm.mmm
@LOCAL
@IF(name)
```

A continuación podemos configurar el tiempo entre las actualizaciones de la exploración en segundos (30), si lo establecemos a 0 se desactivan las difusiones. También podemos determinar el orden en que se evalua la directiva `BrowseOrder`, por defecto `deny,allow`: se deniega todo lo que no se autorice explícitamente.

```
#BrowseInterval 30
#BrowseOrder allow,deny
BrowseOrder deny,allow
```

Puerto utilizado para las difusiones UDP (por defecto IPP)

```
#BrowsePoll address:port
#BrowsePort 631
```

Si deseamos pasar la información desde una red a otra usaremos `BrowseRelay`, ejemplos de uso:

```
#BrowseRelay 172.26.0.2 192.168.0.255
#BrowseRelay 172.26.0.0/24 192.168.0.255
#BrowseRelay source-address destination-address
#BrowseRelay @IF(src) @IF(dst)
```

Caducidad en segundos de las impresoras de red (300 segundos), si en este tiempo no se obtiene una actualización se elimina de la lista de impresoras.

```
#BrowseTimeout 300
```

Permite usar clases implícitas, de esta forma las impresoras de la red que tienen el mismo nombre se ponen en una clase de igual nombre que la impresora[16]. Por defecto no creamos una clase implícita para cada una de las impresoras (`ImplicitAnyCLasses Off`) y no mostramos las impresoras que componen las clases implícitas.

```
#ImplicitClasses On
#ImplicitAnyCLasses Off
#HideImplicitMembers On
```

`Security Options` Grupo de administración del sistema[17] y frecuencia con que se regenera el certificado de autentificación (300)

```
#SystemGroup sys
#RootCertDuration 300
```

Con la directiva `location` podemos establecer controles de autentificación y acceso a determinados recursos del servidor[18]:

[16] Así disponemos de colas redundantes múltiples en nuestra red configuradas de forma automática. Si un usuario envía un trabajo a la clase implícita, el trabajo irá a la primera impresora disponible que compone la clase.

[17] En Debian `lpadmin`

[18] No están todas las opciones, para conocer todas las posibilidades os remitimos a la documentación del programa

classes clases de impresoras

classes/name clase *name* del servidor

jobs todos los trabajos

printers todas las impresoras

printers/name la impresora de nombre *name*

admin todas las cuestiones relacionadas con la administración

/ todas las operaciones de configuración del servidor de impresión

Los valores permitidos son:

AuthType tipo de autentificación, pueder ser

- **none** no se establece ningún método de autentificación
- **basic** en este modelo es necesario un nombre de usuario y contraseña (que se pasan en texto plano) para autentificarse ante el servidor
- **digest** más segura que la anterior[19].

AuthClass nivel de autentificación, puede ser

- **Anonymous** valor por defecto, no se necesita autentificación
- **User** es necesario un nombre de usuario y contraseña
- **System** es necesario un nombre de usuario y contraseña, el usuario ha de pertenecer al grupo del sistema (`sys` y `lpadmin` para Fedora o Guadalinex respectivamente)
- **Group** es necesario un nombre de usuario y contraseña, el usuario ha de pertenecer al grupo definido en la directiva `AuthGroupName`

AuthGroupName nombre del grupo para la autorización comentada antes

Order orden en que se analizan las directivas `allow` y `deny`

Allow From permite el acceso desde determinados máquinas, dominios, ...

Deny From deniega el acceso desde determinados máquinas, dominios, ...

La notación permitida para las dos directivas anteriores se ha comentado ya. Se puede consultar en la 13.3.2 en la página 215

```
#<Location /classes>
#</Location>

#<Location /classes/name>
#</Location>

#<Location /jobs>
#</Location>
```

```
#<Location /printers>
#</Location>

#<Location /printers/name>
  #AuthType None
  #AuthType Basic
  #AuthClass User
  #AuthType Digest
  #AuthClass User
```

[19] Para:

- Conocer mejor las diferencias `http://www.faqs.org/rfcs/rfc2617.html`.
- Saber cómo rabajar con ella: `http://www.cups.org/doc-1.1/sam.html`

```
  #Order Deny,Allow
  #Deny From All
  #Allow From .mydomain.com
#</Location>

<Location /admin>
  AuthType Basic
  AuthClass System
  Order Deny,Allow
  Deny From All
  Allow From 127.0.0.1
  #Encryption Required
</Location>

<Location />
  Order Deny,Allow
  Deny From All
  Allow From 127.0.0.1
</Location>
```

↬ Veamos un ejemplo de configuración:

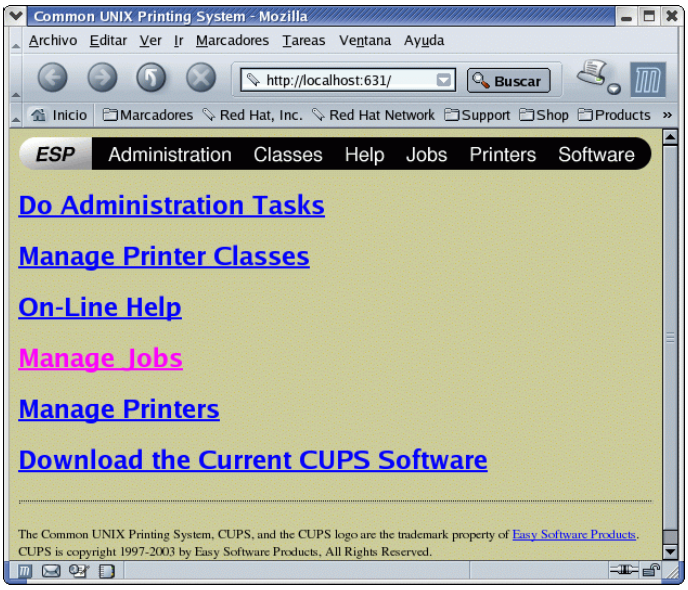
```
<Location />
  # Directorio raíz del servidor CUPS
  # Permitimos sólo acceso desde el bucle local y la red local.
  Order Deny,Allow
  Deny From All
  Allow From 127.0.0.1
  Allow From 172.26.0.0/255.255.255.0
  # Además permitimos la administración a la IP remota[20]
  Allow 150.214.5.11
</Location>
<Location /admin>
  # Configuremos el acceso a la interfaz Web.
  # Autentificación básica y sólo desde la red local
  AuthType Basic
  AuthClass System
  Order Deny,Allow
  Deny From All
  Allow From 127.0.0.1
  Allow From 172.26.0.0./255.255.255.0
</Location>
```

13.4. Interfaz Web

Para acceder a la interfaz web de configuración del programa, abrimos nuestro navegador web y escribimos:

```
http://localhost:631
```

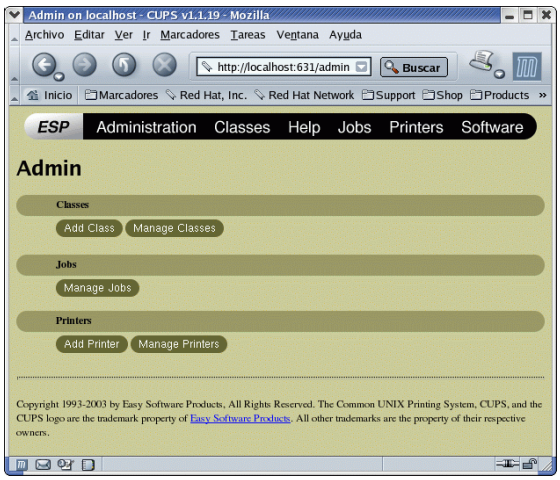
[20] Si mantenemos sólo la directiva
`Listen 127.0.0.1:631`
no podremos acceder desde otras máquinas, es necesario ajustarla añadiendo, por ejemplo:
`Listen 150.214.5.11:631`
o no limitando el acceso

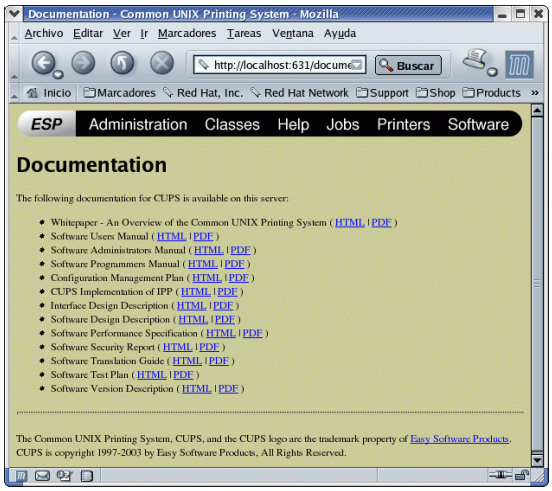


Para poder cambiar algo se nos pedirá mediante una ventana de autentificación la contraseña del root.

Las opciones disponibles son:

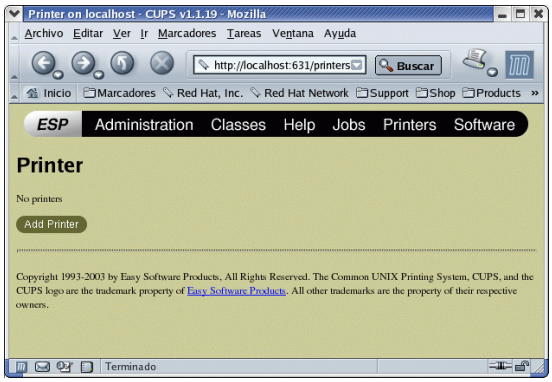
- Tareas administrativas: desde este apartado podemos gestionar las clases, los trabajos de impresión y las impresoras.
- Administrar clases de impresoras
- Ayuda en línea
- Administrar los trabajos de impresion
- Administrar las impresoras
- Acceder a la Web de CUPS por si deseamos bajarnos la última versión del programa.

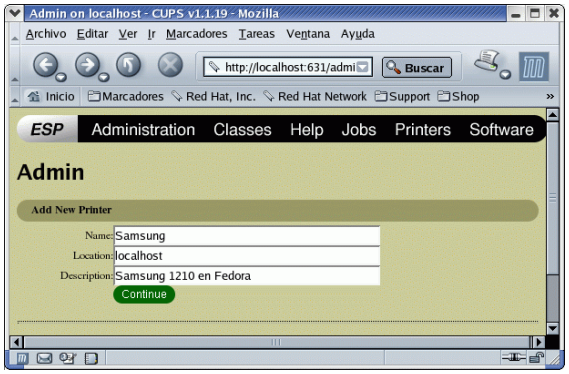




13.4.1. Añadir una impresora

Pulsamos sobre **Printers** y optamos por añadir una impresora (**Add Printer**). Introducimos el nombre, la localización y una breve descripción sobre la impresora que estamos configurando. Sólo es obligatorio introducir el nombre (es el único campo que no podremos modificar después).





A continuación hemos de optar por el interfaz al que está conectada. Si nuestra impresora es local, sólo hemos de optar por el puerto adecuado (serie, usb, paralelo). Algunas de las opciones son:

- Puerto paralelo
- Puertos USB
- Puertos serie
- Impresora en red compartida mediante el sistema LPD
- Impresora compartida mediante IPP (con otro CUPS o windows 2000)
- Impresora compartida mediante SMB (protocolo de red de windows)

Además de impresoras conectadas de forma local, CUPS soporta los protocolos socket, LPD ("*Line Printer Daemon*"), IPP y smb[21]. En CUPS a cada cola de impresión se le asocia un nombre y un dispositivo. La sintaxis URI con que se especifica cada dispositivo es por ejemplo:

- `parallel:/dev/lp0` para una impresora local conectada al puerto paralelo.
- `lpd://servidor/lp` para una impresora de nombre lp conectada a un "servidor" de impresión UNIX en el que corre un sistema LPD
- `ipp://servidor/impresora` o `ipp://servidor/printers/impresora` en este caso, nuestro servidor utiliza CUPS y deseamos imprimir en la impresora de nombre `impresora`
- `smb://servidor/impresora`, `smb://user:password@workgroup/servidor/impresora` o `smb://user:password@servidor/impresora` si usamos un servidor de impresión basado en Windows.

Podemos conocer los *back-ends* disponible es nuestra máquina listando `/usr/lib/cups/backend`[22]

Si optamos por seleccionar una impresora de red tendremos que introducir la ruta completa al dispositivo. CUPS nos pone el protocolo que seleccionemos (podemos modificarlo al cambiar la URI). Por ejemplo:

file:/tmp/impresora.prn Imprimirá en el archivo especificado

[21] impresión en una impresora compartida de Windows

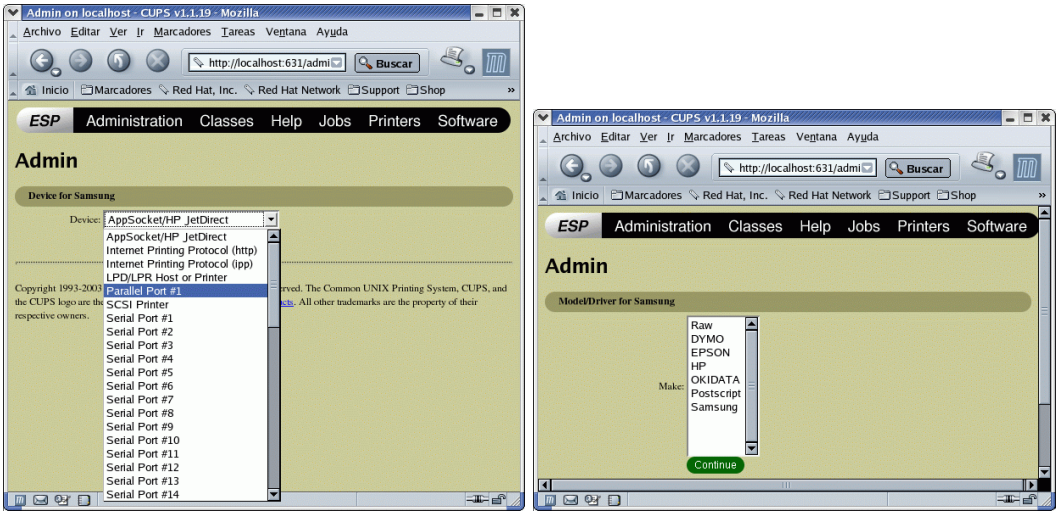
[22] O usar:

```
$/usr/sbin/lpinfo -v
```

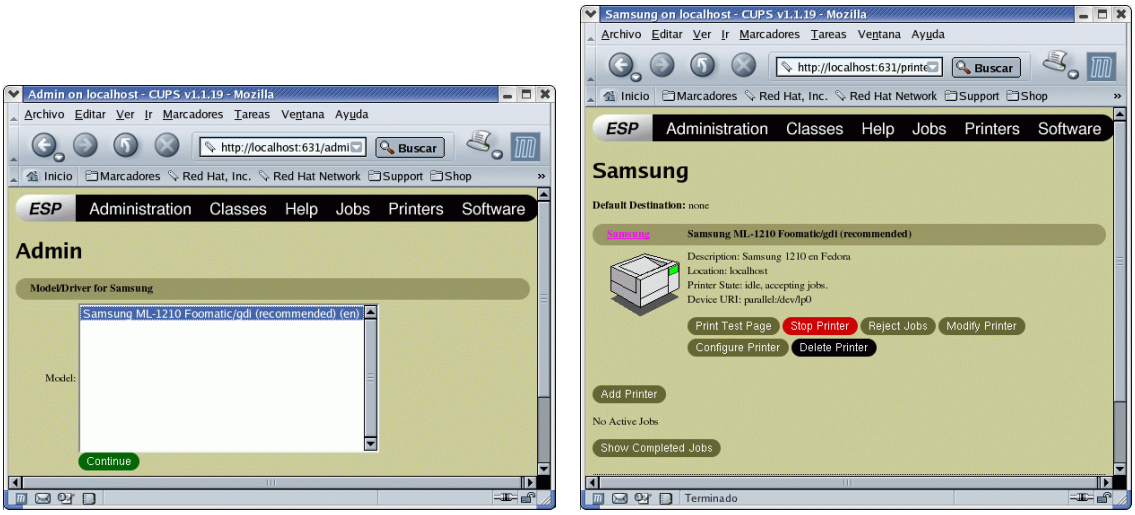
ipp://mileto.cica.es/printers/lp0 Imprimirá en la cola `lp0` de la máquina `mileto`

smb://murgi/tux/epson Imprimirá en una impresora compartida de una red windows con:

- grupo de trabajo `murgi`
- nombre del servidor `tux`
- recurso compartido `epson`



Llega el momento de seleccionar la marca y el modelo de impresora (filtro a usar para nuestra impresora, en el ejemplo capturado se trata de una Samsung ML1210). Una vez que todo está bien ya tenemos nuestra impresora lista para usar. Antes de dar por finalizada la configuración es conveniente imprimir una página de prueba (**Print Test Page**) y, en su caso, ajustar los parámetros de impresión (tipo de papel, resolución, ...)



Si accedemos a la impresora instalada, veremos que disponemos de las opciones:

Print test page imprimir una página de prueba

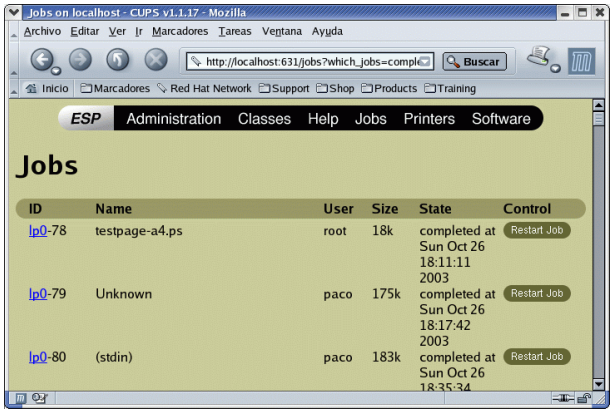
Stop printer para detener los trabajos en curso

Rejects Jobs rechazar trabajos de impresión

Modify Printer cambiar la configuración de la impresora

Configure Printer para ajustar los parámetros finales de impresión: tipo de papel, resolución, añadir un *banner*, ... Dependen del modelo de impresora.

Desde la sección **Jobs** podemos acceder a los trabajos pendientes de imprimir (**Show Active Jobs**) y los trabajos completados (**Show Completed Jobs**)



En esta ventana se nos informa de:

ID impresora responsable del trabajo

Name nombre del fichero impreso

User usuario que ha mandado el trabajo

Size tamaño del fichero

State estado en que se encuentra: activo, cancelado, en espera.

Control si el trabajo está completado podemos imprimirlo de nuevo pulsando sobre **Restart Jobs**.

Si el trabajo está aún activo, nos aparecen las opciones:

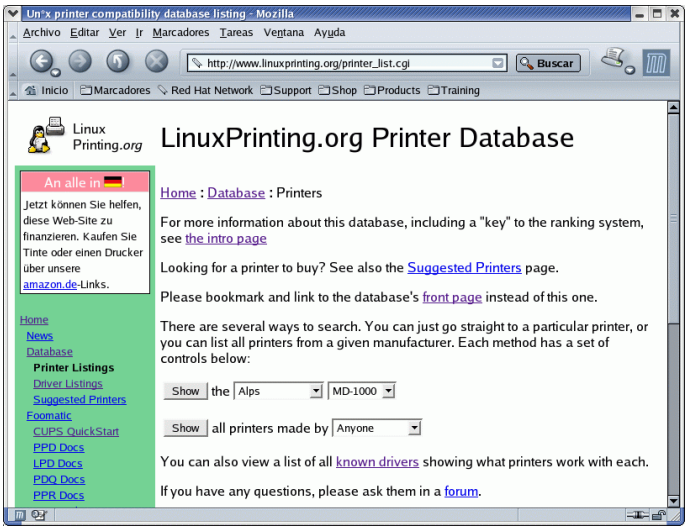
Hold Jobs para detener el trabajo de impresión

Cancel Jobs para cancelar el trabajo

➢ Si el filtro no está instalado

En este caso la página de obligada visita es

`http://www.linuxprinting.org/printer_list.cgi`



Veamos un par de ejemplos[23]::

↬ **Samsung ML1210** Tras consultar en la web anterior nos informan que:
Recommended driver: gdi (Home page, view PPD, download PPD)
Bajamos el driver PPD y lo ponemos en el lugar adecuado
`# cp Samsung-ML-1210-gdi.ppd /usr/share/cups/model/`
Tras esto, es fundamental reinicar el servicio, si no, no lee los cambios.

- Fedora

  ```
  #service cups restart
  ```

- Debian

  ```
  #/etc/init.d/cupsys restart
  ```

↬ **Epson C62** Para esta impresora (y tras revisar la página anterior) obtenemos que:
Recommended driver: gimp-print (Home page)
Instalamos el paquete adecuado `gimp-print-cups`
`#apt-get install gimp-print-cups`
y tras reinicar el servicio todo solucionado.

13.4.2. Añadir una clase

Una de las características que hacen de CUPS un gran servidor de impresión es la idea de clase. Una clase está formada por una serie de impresoras y es CUPS, cuando imprimimos sobre esa clase, el que se encarga de gestionar la carga: si una impresora está ocupada se envía el trabajo a otra de la misma clase, de esta forma podemos minimizar tiempos de espera.

Para crear una clase optaremos por pulsar sobre **Classes** y tras rellenar el nombre, localización y breve descripción pasamos a seleccionar las impresoras que van a constituir la clase.

[23] Se trata de dos ejemplos "ficticios" ya que el filtro para ambos modelos, en general, debería estar ya instalado.

Las posibilidades de administración de una clase son similares a las comentadas sobre impresoras.

13.5. Un poco de comandos

CUPS proporciona los comandos[24]

/usr/bin/cancel cancela los trabajos de impresión existentes

/usr/bin/disable para la impresora o clase pasada como argumento

/usr/bin/enable arranca la impresora o clase pasada como argumento

/usr/bin/lp imprime el fichero o altera un trabajo pendiente

/usr/bin/lpoptions muestra o establece las opciones de las impresoras

/usr/bin/lppasswd para añadir, cambiar o borrar contraseñas en el fichero de contraseñas de CUPS (`passwd.md5`)

/usr/bin/lpq muestra, para la impresora pasado como parámetro, el estado actual de la cola de impresión

/usr/bin/lpr imprime el fichero pasado como parámetro

/usr/bin/lprm cancela trabajos de la cola de impresión

/usr/bin/lpstat muestra información del estado de las clases, trabajos e impresoras actuales

/usr/sbin/accept indica al sistema de impresión que acepte trabajos para el destino indicado

/usr/sbin/cupsaddsmb exporta impresoras para usarlas con clientes Windows mediante SAMBA

/usr/sbin/cupsd demonio de impresión

/usr/sbin/lpadmin para configurar las impresoras y clases

[24] En general, en Fedora a algunos se les puede añadir el "apellido" `.cups` (en Debian no aparece). Así, para imprimir un trabajo de nombre `fichero.ps` escribiremos:

```
$lp fichero.ps
```

/usr/sbin/lpc permite controlar las impresoras y clases

/usr/sbin/lpinfo lista los dispositivos disponibles o los controladores conocidos por el servidor CUPS

/usr/sbin/lpmove mueve el trabajo especificado a otro destino

/usr/sbin/reject indica al sistema que rechace trabajos de impresión para el destino especificado

Para conocer las opciones de estos comandos os recomendamos usar las páginas man de cada uno de ellos.

El uso de `lpr`, `lprm` y `lpc` es inmediato.

Todo lo que hemos visto usando el interfaz Web (y más) se puede hacer con los comandos anteriores. Uno de los más importantes es:

13.5.1. lpadmin

permite añadir impresoras y colas de impresión
Su sintaxis básica es
```
#lpadmin -p mi_impresora -E dispositivo -m driver.ppd
```
con este comando añadimos la impresora de nombre `mi_impresora`

↪ Por ejemplo, para añadir la impresora Samsung de ejemplo, conectada al puerto parelelo, escribiremos

```
#lpadmin -p samsung -E -v parallel:/dev/lp0 -m Samsung-ML-1210-gdi.ppd
```

Para modificar una impresora escribiremos
```
#lpadmin -p mi_impresora -m driver1.ppd
```
si lo que deseamos es trabajar con otro filtro de impresión
Para poner mi_impresora como impresora por defecto
```
#lpadmin -d mi_impresora
```

Si queremos eliminar la impresora anterior usaremos
```
#lpadmin -x mi_impresora
```

Para añadir la impresora a la clase pasada de nombre `mi_clase`
```
#lpadmin -p mi_impresora -c clase
```

y, para eliminar la impresora de la clase `mi_clase`
```
#lpadmin -p mi_impresora -r mi_clase
```

Si lo que deseamos es eliminar una clase escribiremos
```
#lpadmin -x mi_clase
```

Si no queremos que dos usuarios de nombre `usuario1` y `usuario2`[25] puedan imprimir en la impresora pasada como parámetro escribiremos
```
#lpadmin -p mi_impresora -u deny:usuario1,usario2
```
y, para que `usuario1` pueda imprimir de nuevo
```
#lpadmin -p mi_impresora -u allow:usuario1
```

 `lpstat`

Para conocer el estado de la impresoras usaremos

```
$ lpstat -v
```

[25] Para conseguir esto mismo con un grupo escribiremos `@grupo`

```
device for epson: smb://MURGI/NOVO/epson
device for epsonfed: ipp://172.26.0.2:631/printers/epson
device for lp0: parallel:/dev/lp0
```

en esta salida de ejemplo podemos ver que en la máquina donde se ha ejecutado el comando se dispone de:

- Una impresora en red que trabaja sobre una máquina Windows
- Una impresora en red sobre una máquina Linux usando CUPS
- Una impresora local conectada al puerto paralelo

13.6. ➥ Para Practicar

Veamos un par de ejemplos sobre las posibilidades que nos brinda CUPS. En ambos sólo hemos puesto aquellas directivas necesarias para conseguir que todo funcione, el resto del fichero ha quedado igual en el fichero de configuración.

 Dos notas que nos pueden facilitar la realización de las prácticas:

- Si cambiamos el fichero que configuración de CUPS y deseamos que se active, antes hay que reiniciar el servicio.
- Para comprobar los cambios al trabajar con autentificación de usuarios hemos de reinciar el navegador web.

1. Con este ejemplo de configuración sólo permitimos imprimir en nuestra impresora local (`Samsung`) al usuario "matematicas". Lo hacemos en dos pasos:

   a) Modificamos el fichero de configuración de CUPS para que permita autentificación básica y que sea accesible desde la red local.

```
########
########␣Browsing␣Options
########

#Descomentamos␣para␣que␣se␣anuncie␣a␣la␣red␣local
BrowseAddress␣@LOCAL

<Location␣/>
————→Order␣Deny,Allow
————→Deny␣From␣All
————→Allow␣From␣127.0.0.1
————→#Permitimos␣el␣acceso␣a␣la␣red␣local
————→Allow␣From␣@LOCAL
</Location>

<Location␣/jobs>
#
#␣You␣may␣wish␣to␣limit␣access␣to␣job␣operations,␣either␣with␣Allow
#␣and␣Deny␣lines,␣or␣by␣requiring␣a␣username␣and␣password.
#
————→AuthType␣Basic
————→AuthClass␣User
</Location>

#ñaadimos␣la␣óseccin
```

```
<Location /printers/Samsung>
        →#Usamos óautenticacin ábsica
        →AuthType Basic
        →#Permitimos el acceso a todos los usuarios del sistema
        →AuthClass User

        →#Descomentamos para restringir el acceso a la red local
    192.168.1.0
        →#éTambin es posible hacerlo usando la ílnea comentada al
    final
        →Order Deny,Allow
        →Deny From All
        →Allow From 192.168.1.0/255.255.255.0
        →#Allow From @LOCAL
</Location>
```

Listado 13.1: Cups Matemáticas

Así cuando deseamos imprimir un trabajo, nos pedirá el nombre de usuario y la contraseña.

b) Sólo permitimos que imprima el usuario del sistema de nombre matematicas, para eso hemos de usar la directiva AllowUser en el fichero de configuración de las impresoras (`/etc/cups/printers.conf`). Para el modelo de ejemplo quedaría

```
# Printer configuration file for CUPS v1.1.21rc1
# Written by cupsd on Thu Mar  3 22:33:50 2005
<DefaultPrinter Samsung>
        →Info
        →Location
        →DeviceURI parallel:/dev/lp0
        →State Idle
        →Accepting Yes
        →JobSheets none none
        →QuotaPeriod 0
        →PageLimit 0
        →KLimit 0
        →#ñAadimos esta ílnea para permitir que solo pueda imprimir
    el usuario matematicas
        →AllowUser matematicas
</Printer>
```

Listado 13.2: Printers Cups Matemáticas

2. Con el segundo ejemplo, veamos algunas de las posibilidades de CUPS usando Internet. Se trata de conseguir que:

- Podamos configurar[26] la impresora del instituto desde nuestra casa (80.32.193.107) en la impresora (de nombre imprenta) que está en una máquina del instituto (IP 80.32.184.162).

```
#Si deseamos que sea accesible desde la IP especificada hay que
    permitirlo
Listen 127.0.0.1:631
#IP úpblica de la ámquina del centro
Listen 80.32.184.162:631

<Location /admin>
        →AuthType Basic
        →AuthClass System
```

[26] Tal cual está planteado tendremos que conocer la password del root de la máquina del instituto

```
        ##␣Restringimos␣el␣acceso
        Order␣Deny,Allow
        Deny␣From␣All
        Allow␣From␣127.0.0.1
        #Permite␣la␣óadministracin␣desde␣la␣IP␣de␣casa
        Allow␣From␣80.32.193.107
        #Encryption␣Required
</Location>

<Location␣/>
        Order␣Deny,Allow
        Deny␣From␣All
        Allow␣From␣127.0.0.1
        #Permite␣acceder␣desde␣la␣IP␣de␣casa
        Allow␣From␣80.32.193.107
</Location>
```

Listado 13.3: cups-2.1

- Imprimir desde nuestra casa en la impresora que hay conectada a la máquina del instituto. Para poder hacerlo, además de tener que adecuar el fichero de configuración de CUPS para que nos permita imprimir:

  1. Tendremos que permitir la posibilidad de imprimir

```
<Location␣/printers/imprenta>
        Order␣Deny,Allow
        Deny␣From␣All
        Allow␣From␣127.0.0.1
        AuthType␣None
        #Para␣que␣nos␣permita␣imprimir␣desde␣nuestra␣casa␣en␣esta␣
    impresora
        Allow␣From␣80.32.193.107
</Location>
```

Listado 13.4: cups-2.2

  2. Por último, añadiremos en nuestro sistema la impresora remota, por ejemplo con

     `ipp://80.32.184.162:631/printers/imprenta`

     y seleccionar después el filtro adecuado

Capítulo 14

Samba

> Samba es la idea de Andrew Tridgell, ... Unos cuantos años después, él lo expandió como su servidor SMB particular y comenzó a distribuirlo como producto por Internet bajo el nombre de servidor SMB. Sin embargo, Andrew no pudo mantener ese nombre -ya pertenecía como nombre de producto de otra compañía-, así que intentó lo siguiente para buscarle un nuevo nombre desde Unix:
>
> ```
> grep -i 's.*m.*b' /usr/dict/words
> ```
>
> y la respuesta fue:
>
> ```
> salmonberry samba sawtimber scramble
> ```
>
> De ésta manera nació el nombre de Samba.
>
> (*Usando Samba*, Robert Eckstein y otros)

14.1. ¿Qué es Samba?

La página principal de Samba es:
`http://www.samba.org`
Mediante Samba y a través del protocolo TCP/IP podemos compartir y utilizar recursos de sistemas de ficheros Unix e impresoras con otros sistemas operativos[1] (discos duros e impresoras) que "hablen" el protocolo SMB (*Session Message Block*, Bloque de mensajes de sesión). Samba es rápido y sencillo de configurar. Linux con Samba puede trabajar como servidor y como cliente. Como servidor ofrece recursos (discos e impresoras) para que los utilicen las máquinas windows. Como cliente utiliza los servicios ofrecidos por las máquinas windows[2]

Una noticia interesante que apareció hace poco en Internet exponía: "Segun los testeos realizados por IT Week Labs (`http://www.itweek.co.uk/News/1144312`), confirman que la nueva versión de Samba , es casi tres veces más rápida que su contraparte comercial de Microsoft. " Con Samba, además, disponemos también de servicios de dominios NT[3].

[1] Sistemas Windows 3.11, 9x, NT, 200x, XP y OS/2
[2] Os remitimos a la documentación comentada más adelante para esto
[3] Si proporciona servicios de archivo y de impresión, soporte de *Active Directory*, ..., es mejor como NT que un NT y es gratis, la pregunta es obvia (aunque Microsoft y su propaganda intente demostrar lo contrario).

Samba debe sus "orígenes" a Andrew Tridgell. Necesitaba poder compartir archivos desde el DOS a su servidor UNIX y consiguió el primer programa sobre 1992, que si bien funcionó dejaba bastante que desear. En 1994 y tras retomar el proyecto inicial pero ahora con la idea de interconectar en la misma red Windows y Linux apareció de forma "oficial" Samba. Con Samba disponemos de los servicios:

- Servicios de archivos y de impresión.
- Autentificación y autorización.
- Servicios de resolución de nombres.
- Servicios de exploración.

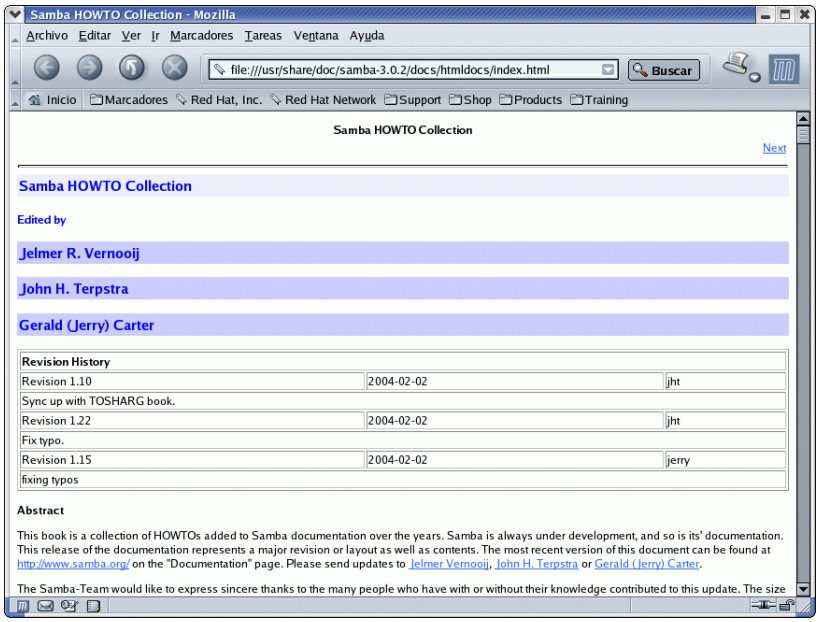
La versión de SAMBA con la que vamos a trabajar es la 3.0. Entre otras mejoras sobre versiones anteriores, destacamos:

- Soporte de *Active Directory*
- Soporte Unicode
- Nuevo sistema de autentificación
- Nuevos comandos net
- Mejor soporte de impresión para Win 2000/XP/2003 incluyendo la publicación de los atributos de impresora en el *Active Directory*

Para afinar mejor la configuración y ampliar sobre el tema os remitimos a

- la extensa documentación que acompaña al programa[4]

  `/usr/share/doc/samba-x.x.x`



[4] En Guadalinex hay que instalar el paquete `samba-doc`

- la traducción del libro de Oreilly *Usando SAMBA* (imprescindible) disponible con la documentación que acompaña al programa y traducido en[5]:

  `http://es.tldp.org/Manuales-LuCAS/USANDO-SAMBA/`

  `http://www.sobl.org/modules.php?name=Downloads`

- al cómo *Samba-Como*

  `http://lucas.hispalinux.es/COMO-INSFLUG/COMOs/Samba-Como/Samba-Como.html`

- un documento traducido por PEDRO P. FÁBREGA llamado *Configuración y Uso de Samba* que podéis conseguir en:

  `http://www.arrakis.es/~pfabrega`

  en él se explican todos los parámetros del archivo de configuración de SAMBA.

- a una página personal

  `http://teleline.terra.es/personal/garzones/garzalin.html`

- además, en Inglés, tenemos los Howtos (están más actualizados que los que hay en castellano):

  `http://www.tldp.org/HOWTO/Samba-Authenticated-Gateway-HOWTO.html`

  `http://www.tldp.org/HOWTO/SMB-HOWTO.html`

  `http://www.tldp.org/HOWTO/Windows-LAN-Server-HOWTO.html`

- y cómo no, a las manpages de los programas.

14.2. Instalación

14.2.1. Fedora

Para garantizarnos que disponemos de la última versión del programa ejecutemos[6]
`# apt-get install samba samba-common samba-client`

¿Qué contiene cada paquete?

- Paquete "principal" de la aplicación. Al instalarlo dispondremos de un servidor SMB:

  `samba-x.x.x.i386.rpm`

- Si está instalado dispondremos de varios clientes SMB que complementan el sistema de ficheros SMB. Permiten acceder a recursos compartidos SMB (archivos e impresoras)

  `samba-client-x.x.x.i386.rpm`

- Proporciona ficheros necesarios para los paquetes anteriores

  `samba-common-x.x.x.i386.rpm`

Si instalamos sólo los dos últimos ¿qué disponibilidad tenemos? Muchísima, podemos ser clientes de máquinas windows. Dispondríamos ya (entre otros ficheros) de las utilidadades:

[5] Una versión anterior

[6]

- De nuevo comentar que también se puede usar el comando `yum`.
- También podemos optar por usar el comando `rpm` instalando la versión que viene en los CDs de la distribución o bajarlos "a mano" de `http://us2.samba.org/samba/ftp/Binary_Packages/Fedora/RPMS/i386/core/3/`

```
$rpm -ql samba-client | grep bin/
/sbin/mount.cifs
/sbin/mount.smb
/sbin/mount.smbfs
/usr/bin/findsmb
/usr/bin/nmblookup
/usr/bin/rpcclient
/usr/bin/smbcacls
/usr/bin/smbclient
/usr/bin/smbmnt
/usr/bin/smbmount
/usr/bin/smbprint
/usr/bin/smbspool
/usr/bin/smbtar
/usr/bin/smbtree
/usr/bin/smbumount

$rpm -ql samba-common | grep bin/
/usr/bin/net
/usr/bin/ntlm_auth
/usr/bin/pdbedit
/usr/bin/profiles
/usr/bin/smbcquotas
/usr/bin/smbpasswd
/usr/bin/testparm
/usr/bin/testprns
/usr/bin/wbinfo
/usr/sbin/winbindd
```

y del fichero de configuración de Samba
`/etc/samba/smb.conf`

14.2.2. Debian

Si bien se instala por defecto, lo mejor es actualizar los paquetes a la última versión[7]:
`# apt-get install samba samba-common smbclient samba-doc smbfs`

14.2.3. Programas

Una vez instalados los programas, tenemos a nuestra disposición las utilidades[8]:

smbd demonio SMB, se encarga de los servicios de archivos, de impresión y autentificación y autorización

nmbd demonio de servidor de nombres NetBIOS.

winbindd demonio para resolver nombres con servidores NT.

Además de estos demonios, en los paquetes que componen el programa Samba tenemos entre otros:

findsmb nos muestra información sobre las máquinas SMB.

net utilidad similar a la del Windows o DOS

nmblookup se usa para consultar nombres de NetBIOS y mapearlos a direcciones IP.

smbclient cliente tipo ftp .

smbmount para montar sistemas compartidos SMB en nuestra máquina Linux.

smbumount para desmontar un sistema de archivos SMB ya montado.

---

[7]

- Los dos últimos paquetes no son "indispensables", se trata de la documentación de SAMBA y de poder disponer de los comandos `mount` y `umount` para el sistema de ficheros `smb`.
- También podemos bajarlos de
  `http://us2.samba.org/samba/ftp/Binary_Packages/Debian/samba3/dists/stable/main/binary-i386/`
  y usar `dpkg`.
- Si deseamos disponer del demonio `winbindd` habrá que instalarlo:

  ```
  #apt-get install winbind
  ```

[8] No están todas las utilidades que contienen los paquetes que componen samba.

smbadduser para añadir usuarios.

smbpasswd para cambiar la contraseña de acceso SMB tanto local como remota.

smbprint smbclient reducido que permite imprimir en los recursos de impresión compartidos SMB.

smbstatus utilidad para mostrar las conexiones SMB activas.

smbtar para hacer copia de seguridad de los sistemas de archivos compartidos SMB en una unidad de cinta de nuestra máquina Linux.

smbtree un buscador en modo texto de máquinas que hablan el protocolo SMB

swat utilidad para configurar SAMBA usando un navegador[9].

testparm revisa/prueba los archivos de configuración de SAMBA.

testprns para revisar el nombre de impresora para usarlo con SAMBA.

14.3. Configuración

Una vez instalados los paquetes estamos casi listos para funcionar, ya que los demonios que requiere se ponen en marcha por defecto al arrancar el sistema operativo[10]. Antes de activarlos tendremos que configurar la máquina Linux y la máquina Windows.

Partiremos de una red privada con un grupo de trabajo THALES con la siguiente configuración:

Sistema Operativo	Nombre	IP
Linux	eco	172.26.0.2
Windows 98	bag	172.26.0.11
Windows XP	compa	172.26.0.12

El proceso consiste:

14.3.1. Configuración de las máquinas Güindows

Para trabajar con Samba tendremos que tener cargados los protocolos TCP/IP, por tanto, en Red comprobaremos que tenemos instalados[11] esos protocolos

[9] Hay que instalarlo

[10] Si deseamos activarlos sin reiniciar el sistema escribiremos:

Fedora `#/etc/rc.d/init.d/smb start`

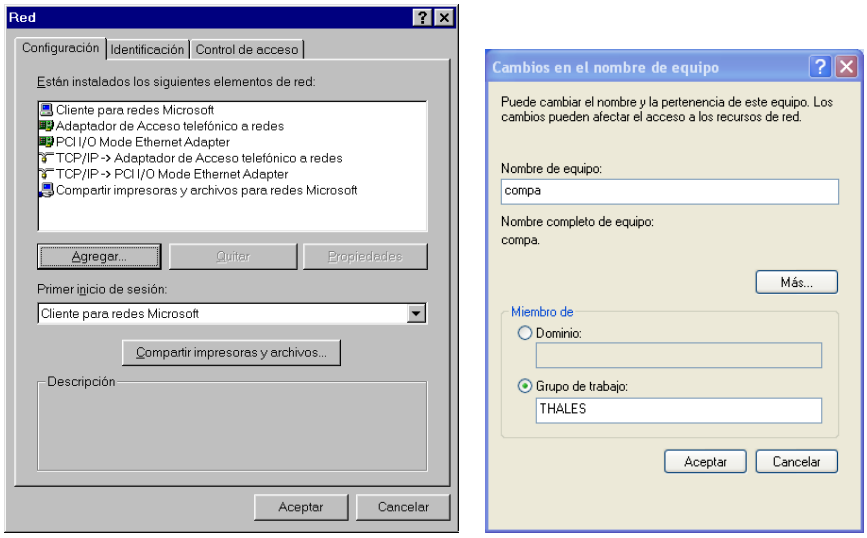
Debian `#/etc/init.d/samba start`

Si optamos por cambiar las opciones de arranque podemos usar las herramientas gráficas:

Fedora `#system-config-services`

Debian `#services-admin`

[11] Notar que las capturas son un poco viejas, pero hemos preferido mantenarlas porque en Windows 95 no se instalaba por defecto el protocolo TCP/IP. No es así en los Windows posteriores.



y asignaremos una dirección IP a nuestras máquinas (por ejemplo 172.26.0.11), en máscara de Red pondremos 255.255.255.0

Asignaremos un nombre a nuestra máquina (BAG) y pondremos el grupo de trabajo en el caso de que no estuviese ya definido (THALES). Repetiremos este proceso con la máquina COMPA.

Problemas con las contraseñas.

Si todos nuestros equipos con Windows son posteriores a Win95 OSR2 (de esa premisa partimos en la configuración de ejemplo), no tendremos que hacer nada de esto y lo mejor es trabajar con contraseñas encriptadas. Incluso si no estamos seguros lo mejor es dejar este tema aparcado. Sólo en el caso de que tengamos una máquina con un Windows antiguo con la que no conseguimos acceder al servidor SMB y sin embargo sí podemos usar el resto de servicios es cuando deberíamos plantearnos si el problema puede estar aquí. Para saber cómo conseguirlo véase en

Fedora `/usr/share/doc/samba-x.x.x/docs/Registry`

Debian `/usr/share/doc/samba-doc/registry`

y la página **??**.

14.3.2. Configuración de la máquina Linux

Análisis del archivo de configuración de Samba

El archivo de configuración de Samba es `/etc/samba/smb.conf`[12]. Mediante este archivo podemos controlar la gran cantidad de opciones disponibles del programa, aunque tratar a fondo todas y cada una de ellas está más allá del objetivo de esta entrega.

Nos centraremos para su estudio en el fichero de Fedora[13], trasladar lo aquí expuesto a máquinas con Debian no supone ningun dificultad.

Se divide en varias secciones, cada una de ellas determinada por un nombre entre corchetes: `[impresoras]`, `[global]`, etc. En cada sección encontramos una serie de parámetros compuestos por pares de clave/valor. Los comentarios comienzan con punto y coma (`;`) o mediante una almohadilla

[12] Antes de tocar este archivo, como viene siendo habitual, deberíamos hacer una copia por si acaso.
[13] Es más compelto y se presta más a su estudio que el que se instala por defecto con Guadalinex

(#).[14] Ejemplifiquemos esto con una parte de la sección `[global]`:

```
[global]
# La almohadilla indica que estamos en un comentario
   remote announce = 172.26.0.255 172.26.2.44
# El siguiente parámetro está marcado como comentario y
# el segundo no, es decir, está activo
;   local master = no
   os level = 33
```

Si modificamos el fichero de configuración tendremos que reiniciar el servidor[15] con:

Fedora `#/etc/rc.d/init.d/smb restart`

Debian `#/etc/init.d/samba restart`

Para comenzar a trabajar y conocer las posibilidades que nos ofrece el programa sólo vamos a modificar las opciones más usuales de este fichero.

Sección [global] En esta sección configuraremos parámetros para todo el servidor SAMBA así como algunos valores predeterminados a las otras secciones. Veamos algunas de las opciones más usuales[16].

Comencemos por ajustar el grupo de trabajo[17] en el que nos encontramos. Por ejemplo, si nuestro grupo de trabajo es THALES, escribiremos:

`workgroup=THALES`

Con el parámetro

`server string = Samba Server`

indicamos el nombre que identificará al servidor cuando lo consultan los clientes SAMBA. Con la directiva

`netbios name = bag`

establecemos el nombre NetBIOS de la máquina.

- Si no se porporciona el nombre del grupo de trabajo tomará como grupo predeterminado `WORKGROUP`.
- Si no establecemos el nombre NetBIOS de la máquina tomará el que se obtenga de ejecutar el comando `hostname`

La línea que sigue aparece marcada como comentario. Si la activamos con el valor

`hosts allow = 172.26.0. 127.`

conseguimos que el acceso al servidor de Samba esté restringido a los hosts o redes especificados. En este caso el acceso está limitado a la red 172.26.0.0/24 y al host local. Si deseamos limitar el acceso a una red de clase B escribiríamos 172.26. y si es de clase A, 172.

Con `hosts deny` podemos negar el acceso a determinadas máquinas o subredes (en caso de duda "gana" `hosts allow`). Por ejemplo, con

[14] Normalmente apareceren marcados los parámetros como comentarios utilizando el punto y coma, y se deja la almohadilla para comentarios normales.

[15] Es deseable actualizar el fichero `/etc/hosts` con las direcciones IP y el nombre de cada máquina a la que vamos a acceder. De esta forma con sólo escribir el nombre del equipo al que queremos acceder, el servidor buscará en ese fichero el nombre del equipo y dirección IP correspondiente.

[16] Para conocer todas las opciones véase el libro *Usando Samba* ya comentado.

[17] Deberá estar limitado como máximo a nueve caracteres, sin espacios y todos en mayúsculas.

```
hosts allow = 172.26. EXCEPT 172.26.0.
```

permitimos el acceso a la red de clase B 172.26. pero denegamos el acceso a la subred de clase C 172.26.0.

Por defecto el archivo `smb.conf` permite que estén disponibles todas las impresoras definidas en `/etc/printcap` como recursos compartidos.

```
# si se quiere cargar automáticamente la lista de impresoras en lugar
# de configurarlas por separado, será necesario esto
 printcap name = /etc/printcap
 load printers = yes

# No será necesario especificar el tipo de sistema de impresión a menos
# que no sea estándar. Los sistemas con soporte en la actualidad incluyen:
# bsd, sysv, pip, Iprng, aix, hpux, qnx
;printing = cups

# Así le decimos a cups que los datos han sido tratados
 cups options = raw
```

Para poder usar CUPS como servidor de impresión hemos de introducir aquí algunos cambios. Para que las distintas modificaciones no queden "dispersas" por la entrega hemos optado por concentarlos todos en 14.4.2 en la página 251

Si se quiere especificar una cuenta de usuario *guest* (invitado) para el acceso anónimo a los servicios en un servidor Samba descomentaremos esta línea. Esto no es imprescindible, por omisión se permite el acceso de un usuario *guest* mediante la cuenta *nobody*[18].

```
# Quitar la marca de comentario aquí si se desea una cuenta de usuario guest,
# se debe añadir esto a /etc/passwd, de no ser así, se utiliza el usuario
# nobody
; guest account = pcguest
```

La líneas que siguen nos indican el lugar en dónde se almacerán los logs del sitema SMB así como el tamaño máximo (en Kb) que pueden tener[19]:

```
 log file = /var/log/samba/%m.log
 max log size = 50
```

Si deseamos que se use un sólo fichero escribiremos `log file = /var/log/samba/smbd.log`

Con la entrada que sigue nos garantizamos que sólo los usuarios de la máquina Linux tienen acceso vía SMB

```
# Modo de seguridad. La mayoría querrá nivel de seguridad de usuario.
# Ver security_level.txt para más detalles.
 security=user
# Sólo se debe descomentar esta directiva si security = server
; password server = <NT-Server-Name>
```

Si en vez de `user` escribimos `share` podrán acceder al servidor usuarios no registrados en el sistema. Es mejor si se desea un acceso menos restringido al servicio de impresión o crear una zona pública limitada por la seguridad del recurso.

Con las líneas que siguen (están comentadas por defecto y no deberían descomentarse salvo que tengamos problemas) establecemos el número de caracteres (no distinguen entre mayúsculas y minúsculas) que comprobaremos de los nombres de usuario así como de su contraseña.

```
# El nivel de password permite que concuerden _n_ caracteres de la contraseña
# para todas las combinaciones de mayúsculas y minúsculas.
```

[18] Sobre la forma de dar de alta un usuario véase la orden `smbpassword` 14.4.2 en la página 250

[19] Con un valor de 0 no ponemos ningún límite en cuanto al tamaño de los ficheros.

```
; password level = 8
; username level = 8
```

Aunque las líneas que siguen estén comentadas, no importa ya que SAMBA trabaja con contraseñas encriptadas por defecto: es la forma de pasar las contraseñas de los Windows 95 OSR2 y siguientes.

```
# Si se quiere usar encriptación de la password, consúltese, por favor
# ENCRYPTION.txt, Win95.txt y WinNT.txt en la documentación de Samba.
# NO activar esta opción a menos que se hayan consultado dichos documentos
;encrypt passwords = yes
;smb passwd file = /etc/smbpasswd
```

Permite a Samba actualizar el fichero de contraseñas de Linux cuando un usuario cambia su contraseña encriptada (fichero `/etc/samba/smbpasswd`).

```
# Lo que sigue se necesita para permitir cambiar las contraseñas desde Windows
# y que se actualicen las de Linux.
# NOTE: Úselo con 'encrypt passwords' y 'smb passwd file'.
# NOTE2: No necesita esto para permitir a las estaciones de trabajo cambiar solamente
# las contraseñas encriptadas SMB. Permite que las conraseñas Unix
# se mantengan sincronizadas con las password SMB.
;unix password sync = Yes
;passwd program = /usr/bin/passwd %u
;passwd chat = *New*password* %n\n *Retype*new*password* %n\n ...
```

Permite especificar un fichero que contiene un mapa de nombres de usuarios de los clientes del servidor[20]

```
# Los usuarios de Unix pueden mapear a distintos nombres de usuario SMB
; username map = /etc/samba/smbusers
```

Si la descomentamos, copiamos el fichero objetivo en el actual fichero de configuración a partir de este punto. Si el fichero referenciado no existe se ignora esta directiva. La varibale `%m` almacena el nombre NetBios del cliente

```
# Using the following line enables you to customise your configuration
# on a per machine basis. The %m gets replaced with the netbios name
# of the machine that is connecting
; include = /etc/samba/smb.conf.%m
```

Con `socket options`[21] podemos poner las opciones de *socket* para usarlas cuando hable con el cliente. Se usa para ajustar Samba a bajo nivel y conseguir mejorar las prestaciones de la red local: "Puesto que cada red es diferente (cableado, interruptores, ruido, etc), no hay una fórmula mágica que funcione para todo el mundo. Como consecuencia, si quiere un ajuste fino del rendimiento de Samba para su red específica, tendrá que hacer algunos experimentos. Para los puristas (y un gran remedio contra el insomnio), pueden documentarse sobre sockets en la página `man socket`" (*The Unofficial Samba HOWTO*[22]).

```
# Mucha gente encontrará que esta opción mejora el rendimiento del servidor
# Para más detalles, vea speed.txt y las páginas del manual
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
```

[20] Cada línea de este fichero tiene de formato:
`usuario Unix= uno o más nombres de usuario cliente separados por espacios.`
Por ejemplo:
`thales = PepeLinux`
`mileto = Griego`
Véase la sección 6.2.3.1 del Libro *Usando Samba*

[21] Para entender mejor esto, una buena frase de Linus Torvalds:
He intentado obtener alguna documentación fuera de Digital sobre esto, pero hasta donde puedo decir incluso_ellos_no la tienen;-)

[22] Para conocer más sobre la directiva que sigue se puede consultar este documento.

Con `interfaces` configuramos las direcciones de red a las cuales reconoce el servidor Samba. Si nuestra máquina tiene varias tarjetas de red tendremos que ponerlas aquí, si no lo hacemos sólo reconocerá el primer interfaz de red al arrancar y trabajará sólo con esa subred.

```
# Configura Samba para multiples interfaces de red
# Si se tienen varias tarjetas de red hay que listarlas aquí
#Véase las páginas man para más detalles
; interfaces = 192.168.12.2/24 192.168.13.2/24
```

`browse sync` lista los servidores Samba para sincronizar sus listas de visualización con los demás visualizadores maestros en otras subredes. Si se descomenta la línea de ejemplo, el servidor Samba contactará con la máquina de IP 192.168.3.25 para sincronizar listas de visualización. Con 192.168.5.255 forzamos a Samba a hacer un *broadcast* de peticiones para determinar las direcciones IP de los visualizadores maestros locales en la red 192.168.5.0 para después realizar la sincronización.

Con `remote announce` hacemos que Samba porporcione listas de visualización al visualizador maestro local de una red externa. El comentario anterior se puede extender a este caso para entender la difencencia de escribir 192.168.1.255 y 192.168.2.44 en la línea de ejemplo comentada.

```
# Para sincronizar las listas de visualización
# y los anuncios de peticiones hacia o desde los visualizadores maestros:
# desde una máuquina concreta o hacia una subred completa
; remote browse sync = 192.168.3.25 192.168.5.255
# El host mandará anuncios a todas las subredes locales especificadas aquí
; remote announce = 192.168.1.255 192.168.2.44

# Browser Control Options:
# ponga local master en no si no quiere que samba sea
# un visualizador maestro en su red. Si no, se aplicarán las reglas normales de elección
; local master = no
# OS Level determina la prioridad de este servidor en las elecciones
# del visualizador maestro. Por defecto debería tener un vaor razonable
; os level = 33
# Domain Master especifica que Samba sea el Visualizar Maestro de Dominio. Esto
# le permite a Samba comparar listas de visualización entre subredes
# no lo use si ya tiene un controlador de dominio de Windows NT haciendo esto
; domain master = yes
# Preferred Master hace que Samba establezca el bit de maestro preferido en la elección
de visualizador maestro
# y le da una posibilidad ligeramente mayor de ganar en la elección
; preferred master = yes
```

Podemos conseguir que Linux autentifique a los clientes Windows en la red. Por defecto, estas líneas están comentadas:

```
# Actívese esto si se desea que Samba sea un servidor de inicio de sesión de
# dominio para estaciones de trabajo de Windows95.
; domain logons = yes

# Si se activan los inicios de sesión de dominio puede ser necesario un
# script de inicio de sesión por máquina o por usuario ejecútese un archivo
# específico de procesamiento por lotes de inicio de sesión por máquina
;  logon script =%m.bat

# ejecútese un archivo específico (de procesamiento por lotes de inicio de
# sesión por cada usuario
;  logon script =%U.bat
```

```
# Dónde almacenar perfiles itinerantes (sólo para Win95 and WinNT)[23]
#%L se sustituye por el nombre del servidor netbios,%U es el nombre de usuario
# debe descomentar [Profiles] que aparece más abajo
; logon path = \\%L\Profiles\%U
```

La directiva `name resolve order` especifica el orden de los servicios que usará SAMBA para resolver nombres. Si descomentamos la línea no sigue el método por defecto[24]: primero opta por un servidor Wims, después el fichero `lmhost` y por último usa *broadcasting* para determinar la dirección de un nombre NetBIOS.

```
; name resolve order = wins lmhosts bcast

# Windows Internet Name Serving Support Section:
# WINS Support - Le dice al componente NMBD de Samba que habilite su servidor WINS
; wins support = yes
# WINS Server - Le dice a su componente NMBD que sea un cliente WIMS
# Nota: Samba puede ser un servidor o un cliente WIMS, pero no ambos
; wins server = w.x.y.z
# WINS Proxy - Permite a SAMBA responder a las peticiones de resolución de nombres
# en nombre de un cliente, para que funcione debe de haber al menos
#un servidor WIMS en la red. Por defecto está en No
; wins proxy = yes
# DNS Proxy - si Samba debe intentar o no resolver nombres NetBIOS
# vía DNS nslookups. Por defecto en las versiones 1.9.17 es sí,
# y ha cambiado en la versión 1.9.18 a no.
dns proxy = no
#
```

Sección [homes] Con esta sección podemos controlar de qué forma accederán los clientes al directorio principal de usuario en el servidor Linux. La configuración aquí establecida permite que los parámetros sean válidos para todos lo usuarios y no hay que especificar una configuración para cada usuario por separado.

```
comment = Home Directories
browseable = no
writable = yes
valid users =%S
create mode = 0664
directory mode = 0775
```

Analicemos cada una de las líneas anteriores[25]:

comment cadena de identificación que se muestra a los clientes que examinan el servidor Samba.

browseable al establecerlo a `no` conseguimos que el explorador de Windows no nos muestre los `$HOME` de otros usuarios del sistema Linux. Si no somos ningún usuario registrado del sistema no veríamos ningún `$HOME`. Sin embargo, si nos conectamos como un usuario del sistema Linux aparecerá una carpeta de nombre ese usuario a la que podremos acceder con nuestra contraseña. Si lo establecemos a `yes` aparecerán todas las carpetas de los `$HOME` de usuario, aunque no podríamos acceder a ellas salvo que la validación sea la correcta (nombre de usuario y contraseña)

[23] Véase el libro *Usando Samba*, sección 6.7.1

[24] `lmhosts`, después los métodos de resolución Linux estándar (`/etc/hosts`, *DNS*, y *NIS*), a continuación interroga a un servidor WINS, y por último usa *broadcasting*

[25] Las tres últimas no aparacen en el fichero de configuración por defecto.

writable (sinónimo de **writeable**) permite que un usuario pueda crear y modificar archivos en su directorio `$HOME` cuando inicia una conexion SAMBA. Podemos conseguir esto mismo sustituyendo esta línea por

```
read only = no
```

```
write ok = yes
```

valid users lista de usuarios que pueden conectarse a un recurso (en este caso, la variable `%S` contiene el nombre actual del recurso).

create mode determinamos que los permisos de archivo para todos los archivos creados en el directorio compartido sean 0664.

directory mode los permisos para todos los directorios creados en el recurso compartido serán 0775.

Secciones [netlogon] y [Profiles]

Por defecto están comentadas. Samba soporta la ejecución de script de entrada que permiten configurar las opciones de red cuando los usuarios se conectan. También permite ([**Profiles**]) que cada usuario almacene su perfil, accesible vía red.

Para poder usarlas tenemos que configurar las directivas adecuadas de la sección [**Global**]. Para conocer mejor este tema se puede consultar el capítulo 6 (Usuarios, seguridad y dominios), sección Scripts de Entrada, del libro *Usando Samba*

La recomendación de *The Unofficial Samba HOWTO* sobre la posibilidad de activar la sección [**Profiles**] es: *Si desea habilitar roaming profiles para Windows 2000/XP haga los cambios siguientes en su archivo smb.conf. Nota: ¡es absolutamente desaconsejable a menos que sepa lo que está haciendo (prevenga quebraderos de cabeza)!*. Así que para no tener que tomar una pastilla mejor lo dejamos como está.

Sección [printers]

Con SAMBA podemos configurar de dos formas distintas la forma en que nuestras impresoras están disponibles para la red:

1. Creando una sección específica para compartir en `/etc/printcap` para cada impresora que se quiera compartir. No es el método que vamos a usar.

2. Usar la sección especial `[printers]` para compartir todas las impresoras definidas en el archivo `/etc/printcap`.

Con esta sección definimos cómo se controlan los servicios de impresión en el caso de que no haya entradas específicas en el archivo de configuración de SAMBA: si disponemos de un servicio, SAMBA buscará en primer lugar un servicio con ese nombre, si no hay ninguno con ese nombre se usa esta sección para permitir al usuario conectarse con cualquier impresora definida en `/etc/printcap`.

Con la sección `[printers]` que se lista a continuación damos a compartir todas las impresoras del sistema y además permitimos imprimir a cualquiera, excepto a la cuenta *guest*.

```
[printers]
  comment = All Printers
  path = /var/spool/samba
  browseable = no
  # Set public = yes para permitir al usuario guest imprimir
  # es equivalente a guest ok = yes
  guest ok = no
  writable = no
```

```
printable = yes
```

Los parámetros `comment` y `browseable` ya se han visto en la sección `[homes]`.

Con el parámetro `path` determinamos en qué directorio temporal[26] se copian los archivos antes de imprimirlos, en este caso `/var/spool/samba`.

Con el parámetro `printable` en `yes` permitimos la escritura en el fichero de *spool* de impresión (si los permisos lo permiten). Con el parámetro `writable` a `no` restringimos a que sólo se permita la escritura de trabajos de impresión en el directorio de *spool* definido con el parámetro `path`. Es decir, definimos el servicio como no "escribible" pero sí como "imprimible".


Para poder usar CUPS hay que cambiar está sección, véase 14.4.2 en la página 251

Secciones personalizadas Usando las secciones personalizadas podemos compartir impresoras o directorios de una manera no genérica. La idea que subyace en estas secciones consiste en poder compartir directorios para grupos de usuarios o permitir que determinados directorios sean de acceso público. A su vez, si tenemos varias impresoras conectadas a nuestra máquina y no queremos darlas todas a compartir, podemos usar esta sección para dar a compartir sólo una impresora en concreto.

Con el ejemplo que se lista a continuación permitimos acceso al servicio `trabajos` a los usuarios del grupo llamado `clase`:

```
[trabajos]
  comment = Directorio compartido de la clase de informática
  path = /home/trabajos
  browseable = yes
  writable = yes
  printable = no
  valid users = @clase
```

Las líneas de este ejemplo significan:

- damos a compartir el directorio `/home/trabajos`
- La identificación de este servicio es: `Directorio compartido de la clase de informática`
- Al estar `browseable` en `yes`, se mostrará la carpeta del recurso compartido en el explorador de Windows siempre que accedamos al servidor SMB.
- Permitimos que se pueda escribir en él.
- Con `printable = no` indicamos que no es un servicio de impresión.
- Con el parámetro `valid users = @clase` restringimos el acceso a este directorio a miembros del grupo `clase`.

Veamos cómo compartir una impresora dedicada en la que sólo puede imprimir el usuario `CURSO-LINUX`:

```
[Impresora]
  comment = Impresora a compartir
  # con print ok = yes se consigue el mismo efecto
```

[26] Los permisos de este directorio están configurados para permitir la lectura y la escritura, lo mismo que el directorio `/tmp`.

```
drwxrwxrwt 2 root root 4096 jun 9 20:05 /var/spool/samba
```

```
    # que con la línea que sigue
    printable= yes
    printer = lp
    path = /var/tmp
    public = no
    writable = no
    valid users = cursolinux
```

Para finalizar, ¿por qué no dar a compartir nuestra unidad de CD a los usuarios `THALES` y `MILETO` de nuestra máquina:

```
[Cdrom]
   #En Guadalinex será /cdrom
   path =/mnt/cdrom
   writable = no
   printable = no
    valid users = thales, mileto
    public = no
```

Cuando terminemos de configurar nuestro sistema podemos usar:
`$ testparm`
y comprobar que todo está perfectamente.

En Guadalinex

El fichero que se instala en un sistema actualizado de Guadalinex 2004 es de la forma

```
[global]
  netbios name = G2004_1108897514
  server string = Guadalinex 2004
  workgroup = GUADALINEX
  wins support = no
  encrypt passwords = true

# Do something sensible when Samba crashes: mail the admin a backtrace
# panic action = /usr/share/samba/panic-action %d

[compartido]
  path = /home/compartido
  comment = Directorio compartido en Guadalinex 2004
  writeable = no
  guest ok = yes
  guest only = yes
  browseable = yes
```

Listado 14.1: /etc/samba/smb.conf

No debería presentar problema comprender su significado y deberíamos adecuarlo a nuestros intereses.

Respecto a lo estudiado sólo hay que comentar varias cuestiones:

- Si instalamos varios Guadalinex y no actualizamos SAMBA tendremos que ajustar el nombre NetBIOS ya que por defecto pone en todos ellos el mismo. Si actualizamos SAMBA este problema desaparece.
- En Guadalinex han creado un directorio de uso compartido y visible por todos (`browseable = yes`) que no está exento de riesgos de seguridad ya que:

- no es necesario nombre de usuario ni contraseña para acceder al recurso (`guest ok = yes`), y
- aunque la conexión sea autentificada, en este recurso se accede como anónimo (`guest only = yes`)
- al final de esta sección (14.4.2 en la página 253) hemos añadido un fichero de configuración básico de ejemplo que recoje los aspectos más importantes tratados en este tema, permite que los usuarios accedan a sus `$HOME` desde máquinas Windows y que puedan imprimir desde los windows sobre la máquina Linux.

14.3.3. Swat

Vamos a comentar una herramienta que permite configurar Samba usando el navegador Web, se trata de SWAT (*Samba Web Administration Tool*). SWAT se incluye como parte del paquete estándar de Samba. La idea de este programa (hay más con esta misma filosofía) consiste en facilitar la configuración del servidor.

Tiene una "pega", y es que cuando se usa escribe un archivo de configuración sin comentarios.

Instalación

Fedora `# apt-get install samba-swat`

Antes de poder usar esta utilidad tenemos que configurar nuestro sistema para que permita acceder a ella, para eso hemos de decirle a `xinetd` que nos permita trabajar con él. En el fichero `/etc/xinetd.d/swat` sustituiremos la línea

```
disable = yes
```

por

```
disable = no
```

Después, tendremos que reiniciar `xinetd` para activar los cambios:

```
# /etc/rc.d/init.d/xinetd restart
```

Debian: `# apt-get install swat`

Una vez instalado, hemos de descomentar la línea adecuada (una que comienza por `swat`) del fichero `/etc/inetd.conf` y releer la nueva configuración

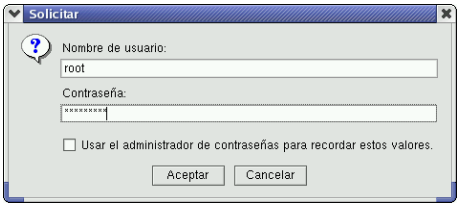
```
#/etc/init.d/inet reload
```

Uso

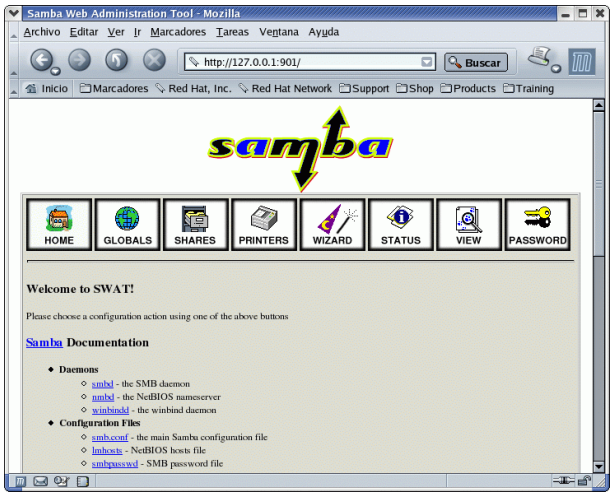
Para usar el programa SWAT tendremos que iniciar un navegador y nos conectaremos a la URL

```
http://localhost:901/
```

Tras la ventana que nos pide los datos del root



accederemos a la página principal de la aplicación:



Los iconos de la parte superior de la pantalla nos permiten acceder a diferentes páginas de SWAT:

página principal, en ella entre otras cosas tenemos enlaces a la documentación del paquete Samba. Entre otros el libro de Oreilly (pero en Inglés) comentado en esta misma sección.

desde aquí podemos manipular la sección globals del archivo `/etc/samba/smb.conf`. Podemos modificar los valores de los distintos parámetros, obtener ayuda sobre ellos y/o mantener el valor predeterminado. Para grabar los cambios pulsaremos sobre [**Commit Changes**].

usando esta página podremos añadir, modificar o borrar recursos compartidos. Por defecto la pantalla inicial muestra sólo los parámetros de uso más frecuente del archivo `smb.conf`. Si pulsamos sobre [**Advanced View**] tendremos la posibilidad de configurar alguno de los parámetros menos usados.

para configurar las impresoras.

desde esta página podemos limpiar el archivo `smb.conf` de todos los comentarios y valores por defecto.

desde aquí podemos comprobar el estado de Samba. Además de ver cómo están las "cosas" podemos arrancar y parar los demonios de Samba y ver las conexiones activas de nuestro servidor Samba.

para examinar el contenido del archivo `smb.conf`. Si deseamos ver todas las variables disponibles y sus valores hay que pulsar en **Full View**.

con este enlace aparecerá la pantalla mediante la cual podremos cambiar cuentas de usuarios locales y cuentas del controlador del dominio primario.

14.4. A "bailar" la Samba

14.4.1. Acceder desde una máquina Linux a una Windows

Como es evidente sólo podremos acceder a aquellos recursos autorizados en la máquina Windows.

Si optamos sólo por usar esta posibilidad no necesitamos tener instalado el paquete "principal" de la aplicación[27]. Trabajando de esta forma Samba no comparte ningún recurso con otro sistema, se limita a acceder a los recursos compartidos en los servidores de recursos la red. Si trabajamos sólo como cliente no tenenos que tener activos los demonios `smbd` o `nmbd` aunque sí debemos ajustar a nuestro grupo de trabajo el archivo `smb.conf`, la única línea necesaria en ese fichero sería:

```
workgroup=THALES
```

Acceder en modo gráfico es fácil, sólo hemos de pulsar sobre el escritorio en **Equipo→Red**[28]

y acceder a la **Red de Windows**. Pero, además de la posibilidad de acceder en modo gráfico, veamos algunas de las posibilidades de trabajo en modo texto:

sbmclient

Una de las utilidades más interesantes de las que acompañan a Samba es `smbclient`. Con él podemos acceder desde Linux a los recursos compartidos en máquinas Windows con métodos que incluirían FTP, NFS y los "comandos r", como rcp.

`smbclient` nos permite disponer de un interfaz similar a un FTP, por tanto, es una utilidad cuyo objetivo es accesos temporales a un recurso compartido.[29]

Veamos algunos ejemplos sobre su uso. Partiremos de la base de que estamos conectados en red, que nuestra máquina Linux se denomina ECO y que podemos acceder a varias máquinas con Windos (máquinas BAG y COMPA).

Antes de nada y como no sabemos qué máquinas están a nuestra disposición ejecutemos:

```
# nmblookup -d 2 thales
added interface ip=172.26.0.2 bcast=172.26.0.255 nmask=255.255.255.0
querying thales on 172.26.0.255
Got a positive name query response from 172.26.0.2 ( 172.26.0.2 )
Got a positive name query response from 172.26.0.11 ( 172.26.0.11 )
Got a positive name query response from 172.26.0.12 ( 172.26.0.12 )
172.26.0.2 thales<00>
172.26.0.11 thales<00>
172.26.0.12 thales<00>
```

O bien

```
$ findsmb[30]
IP ADDR         NETBIOS NAME   WORKGROUP/OS/VERSION
------------------------------------------------------------------------
172.26.0.2     ECO            +[THALES] [Unix] [Samba 3.0.11]
172.26.0.11    BAG             [THALES]
117.26.0.12    COMPA           [THALES]
```

Vemos qué máquinas del grupo de trabajo especificado responden positivamente. Pero el mejor se deja para el final, usamos ahora:

[27] Sólo hay que instalar `samba-client` y `samba-common`.
[28] Equivale a abrir **Nautilus** y escribir `network://`
[29] Si lo que deseamos es mantener una conexión "permanente" es mejor usar `smbmount`.
[30] Sólo para Fedora. Si no tenemos instalados y activos los demonios del paquete samba puede que nos dé algunos errores.

➥ **Para practicar:** comprobar que el comando que mejor nos informa de los equipos y recursos compartidos es:

```
$smbtree
```

■

Vemos que la máquina de IP 172.26.0.11 está encendida y, al ejecutar el comando anterior sabemos ya qué recursos tiene compartidos. Otra forma de listar los recursos compartidos es con el comando `smbclient` pasándole el parámetro `-L`. Por ejemplo, si nuestra máquina Windows se llama `bag` una posible salida es:

```
$ smbclient -L bag
added interface ip=172.26.0.2 bcast=172.26.0.255 nmask=255.255.255.0
Password:
     Sharename   Type    Comment
     ---------   ----    -------
      ERASE      Disk
      ADMIN$     Disk
      PRINTER$   Disk
      EPSON      Printer
        D        Disk
        C        Disk
      IPC$       IPC      Comunicación remota entre procesos

        Server      Comment
      ---------    -------

       Workgroup    Master
      ---------    -------
```

nos muestra una lista con los recursos disponibles, las máquinas que comparten recursos y los grupos de trabajo. Comentar que la máquina Linux verá a la máquina Windows aunque el nombre del grupo de trabajo no sea el mismo.

Si usamos[31]:

```
$ smbclient -L fedora -U cursolinux
added interface ip=172.26.0.2 bcast=172.26.0.255 nmask=255.255.255.0

Anonymous login successful
Domain=[THALES] OS=[Unix] Server=[Samba 3.0.11]

        Sharename      Type      Comment
        ---------      ----      -------
        IPC$           IPC       IPC Service (Samba Server)

        Server               Comment
        ---------            -------
```

[31] Mejor si le decimos que el nombre de usuario y las password están vacías:

```
$ smbclient -U%
```

```
 ECO                Samba Server

Workgroup           Master
---------           -------
THALES               SECRE
```

veremos la lista de los recursos compartidos disponibles en el servidor SMB FEDORA para el usuario *cursolinux*.

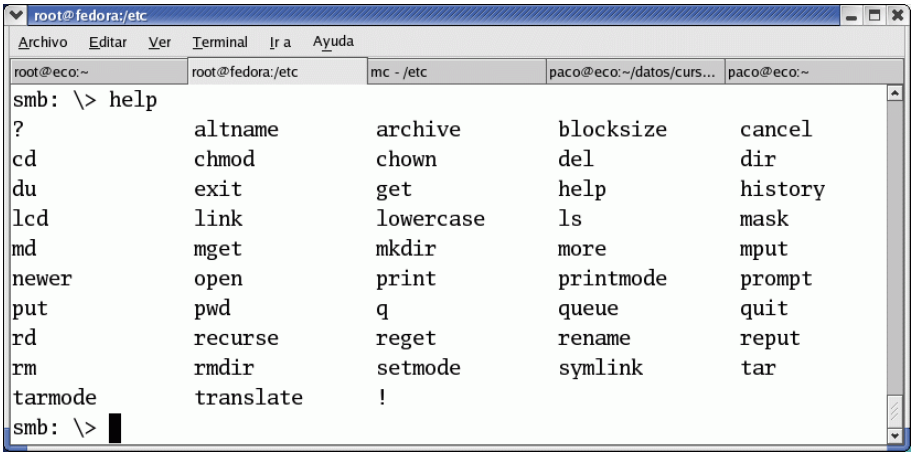
Con salida de `smbclient -L bag` hemos podido comprobar qué recursos compartidos tenemos a nuestra disposición en esa máquina, accedamos a alguno de ellos, para eso escribimos[32]:

```
$ smbclient //bag/c
```

si deseamos acceder al disco C. Tras introducir la contraseña veremos:

```
smb: \>
```

Aquí podemos introducir comandos, para saber cuáles tenemos disponibles podemos usar la orden `help`[33]



Para ampliar sobre el significado de cada uno de ellos sólo debemos usar de nuevo `help` seguido del comando del que queremos ayuda:

```
smb: \>help get
   HELP get:
          <remote name>[local name] get a file
```

Comentemos algunos de los más usuales:

? [comando] muestra ayuda sobre ese comando o lista de comandos

help [comando] igual que ?

! [comando de shell] ejecuta un comando de la shell

cd [directorio] cambia el directorio remoto

lcd [directorio] cambia el directorio local

md [directorio] crea un directorio

mkdir [directorio] igual que md

rd [directorio] borra el directorio

rmdir [directorio] igual que rm

del [archivos] borra el archivo

dir [archivos] lista los archivos

ls [archivos] igual que dir

get [rarchivo] [larchivo] copia el archivo remoto (`rarchivo`) en el archivo de nombre `larchivo`

[32] Si el nombre del recurso compartido contiene espacios (por ejemplo "Mis Documentos") tendremos que escribir: `smbclient //bag/Mis\ Documentos`

[33] Son muy parecidos a los del ftp.

mget [**archivos**] copia los archivos que que coinciden con el nombre especificado (normalmente se usan comodines)

newer [**archivo**] sólo tomará los archivos posteriores al especificado.

put [**larchivo**] [**rarchivo**] copia desde la máquina local el archivo `larchivo` en la máquina remota con el nombre `rarchive`

mput [**archivos**] copia en el servidor los archivos de la máquina local que coinciden con el nombre especificado (normalmente se usan comodines)

printmode [**modo**] determina el modo de impresión (`text` o `graphics`)

print [**archivo**] imprime el archivo especificado en la máquina remota

queue muestra la cola de impresión.

exit sale del programa

quit igual que exit

Por ejemplo, si tras hacer un listado de la máquina remota:

```
smb: \>ls
```

comprobamos que hay un archivo de nombre `curso.txt`, podemos bajarlo a la máquina local usando:

```
smb: \>get curso.txt
```

para salir

```
smb: \>exit
```

smbmount y smbumount

Si deseamos tener la posibilidad de montar un sistema de archivos compartido en el árbol del sistema de archivos de Linux tenemos que usar `smbmount`[34]

Para montar algún recurso de la máquina Windows usaremos el comando:

```
# smbmount //máquina_windows/recurso destino_montaje
```

Veamos su utilidad con un ejemplo. Recordemos que en la máquina BAG disponíamos del recurso `C`, montémoslo en nuestro sistema de archivos. Para eso creemos un directorio destino de montaje:

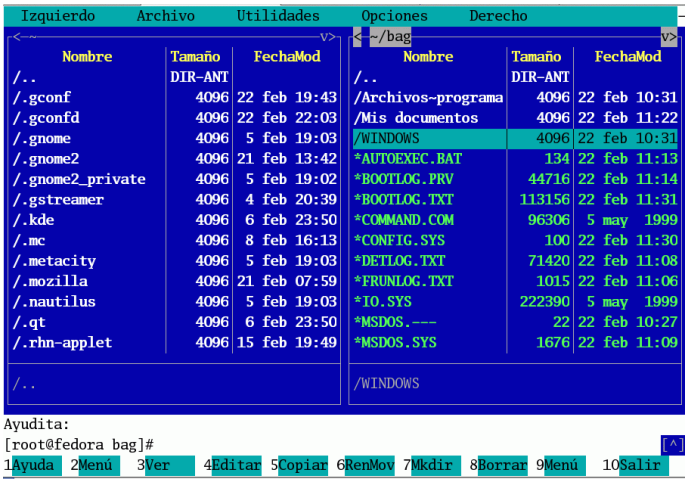
```
# mkdir /mnt/bag
```

ya sí, usemos ahora

```
# smbmount //bag/C /mnt/bag
Password:
```

Al usar este comando, estamos consiguiendo que el recurso compartido sea montado en `/mnt/bag` y que lo veamos como cualquier otra parte del sistema de archivos Linux.

[34] Tanto `smbmount` como `smbumount` las podrán usar los usuarios "normales" si les activamos el bit setuid. Para ampliar sobre su uso os remitimos a las `manpages` de ambos programas.



- Podemos acceder a máquinas Windows sin necesidad de usar el paquete Samba. Esto es posible gracias al soporte del sistema de archivos `smbfs` del núcleo. Si nuestro núcleo está compilado con esta opción (los de RedHat lo están) podemos montar recursos compartidos de máquinas Windows sin tener activos los demonios del paquete Samba y sólo necesitamos el programa `mount`. Si optamos por esta opción, para montar el recurso C de la máquina BAG escribiremos:

  ```
  #mount -t smbfs //bag/c /mnt/bag
  ```

- Debemos tener cuidado porque los archivos de texto de Linux y de Windows son diferentes. Si creamos archivos de texto en un recurso montado vía SAMBA desde Linux, éste utilizará el formato de fin de línea del sistema Linux y después tendremos problemas para leerlos desde windows.

Para desmontarlo usaremos el comando `smbumount`

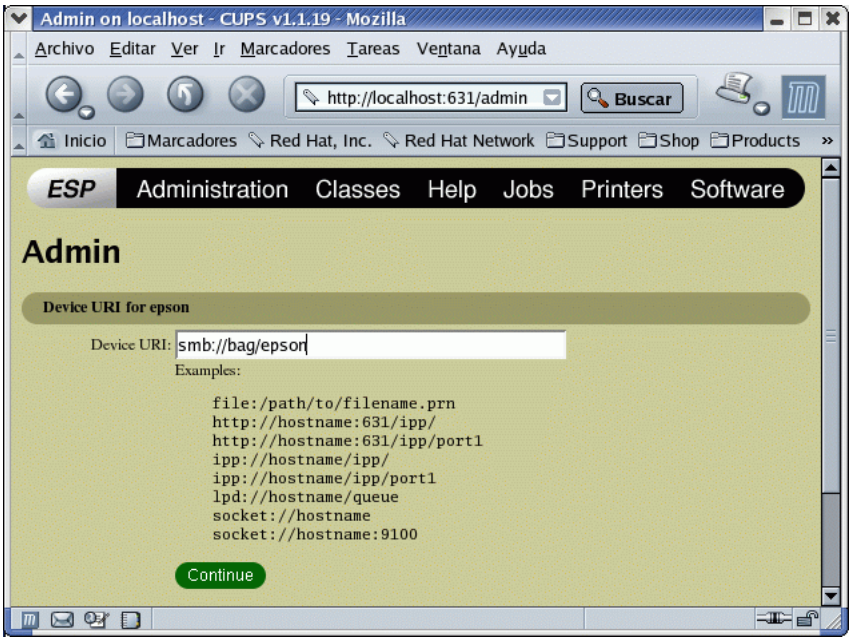
```
#smbumount /mnt/bag
```

o sólo `umount`

```
# umount /mnt/bag
```

Agregar una impresora. Veamos cómo imprimir vía red usando una impresora conectada a una máquina Windows (notar que previo debe de estar dada a compartir). Supongamos que disponemos de la impresora `EPSON` en la máquina BAG, añadámosla a Linux. Iniciamos CUPS, optamos por añadir una impresora (véase 14.4.2 en la página 251) y en Device URI escribimos los datos adecuados a nuestro sistema[35]:

[35] Si no existe el enlace simbólico lo podemos crear con:
`#ln -s /usr/bin/smbspool /usr/libs/cups/backend/smb`



- Si la máquina está en el mismo grupo de trabajo y no se requieren usuario y password:
  `smb://servidor/recursocompartido`
- Si la máquina está en el mismo grupo de trabajo y se requieren usuario y password:
  `smb://nombreusuario:password@servidor/recursocompartido`
- Si la máquina está en otro grupo de trabajo y no se requieren usuario y password:
  `smb://grupotrabajo/servidor/recursocompartido`
- Si la máquina está en otro grupo de trabajo y se requieren usuario y password:
  `smb://nombreusuario:password@grupotrabajo/servidor/recursocompartido`

Una vez especificada la URI correctamente sólo nos falta seleccionar el filtro adecuado.

14.4.2. Acceder desde Windows a la máquina Linux

Un papel más interesante para Linux y Samba en una red Windows es el de servidor de recursos. Para conseguir esto debemos tener instalado el paquete `samba` y tener correctamente configurado el fichero `/etc/samba/smb.conf`.

Para acceder (sin hacer más cambios que los ya comentados[36]) desde una máquina Windows a una Linux sólo lo podremos hacer si somos usuarios registrados en el sistema Linux. Además, en Linux debemos dar de alta a ese usuario y definir con qué contraseña puede acceder a nuestro sistema. Si queremos dar de alta al usuario *cursolinux* de la máquina Linux (con igual nombre en la máquina Windows) usaremos:

```
#smbpasswd -a cursolinux
ENTER password for cursolinux
New SMB password:
Retype new SMB password:
Added user cursolinux.
```

[36]Puede haber también directorios públicos, para los cuales no hace falta que el usuario esté dado de alta como usuario del linux. A los usuarios registrados en Linux además les muestra su directorio `home`.

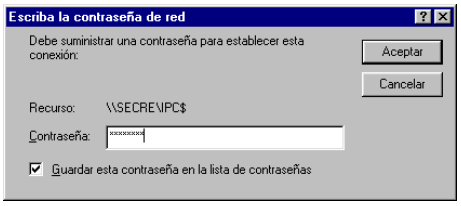
También, si existe un NT que valida los usuarios en la red, se puede poner ese NT como *password server* y es el que valida los usuarios.

Podremos cambiar la contraseña después usando el comando `smbpasswd` (sin el parámetro `-a`). Si lo que deseamos es borrar un usuario le pasaremos como parámetro `-x` seguido del nombre de ese usuario.

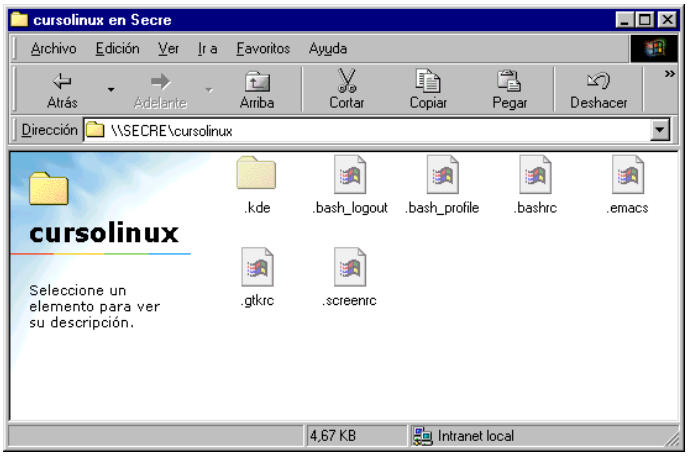
Si nuestro usuario `cursolinux` desea tener otro alias en el sistema tenemos que usar el fichero `/etc/samba/smbusers`. Por ejemplo:

```
root = administrator admin
cursolinux = thales mileto
```

En la máquina Windows 9x tendremos que entrar **como ese usuario** (si se trata de un XP esto no es necesario) y si en el inicio no ponemos la contraseña, ésta se nos pedirá cuando intentemos acceder a los servicios de Red:



Una vez validada la contraseña (interesa desmarcar la casilla de guardar contraseña ya que si no, nos arriesgamos a que nos fastidien el Linux desde Windows), los directorios a los que tengamos acceso en nuestro sistema Linux se nos mostrarán como carpetas de Windows y podremos trabajar con ellas de la forma habitual.



Es importante resaltar que si el grupo de trabajo no está bien configurado en el fichero `/etc/smb.conf` no veremos a la máquina Linux cuando accedamos a la red.

Agregar una impresora.

Vamos a partir de la base de que los *drivers* se instalan de lado del cliente (en la máquina Windows). Si se opta por usar el servidor SAMBA como depósito de los *drivers*, véase la documentación de SAMBA.

Para poder usar una impresora configurada con CUPS desde una máquina Windows hay que realizar una serie de cambios en los ficheros de configuración de samba y de CUPS. Comencemos por `/etc/smb.conf`, tenemos que añadir/modificar

```
load printers = yes
printing = cups
printcap name = cups
```

en la sección [**Global**]. Además, la sección [**Printers**] ha de quedar como sigue

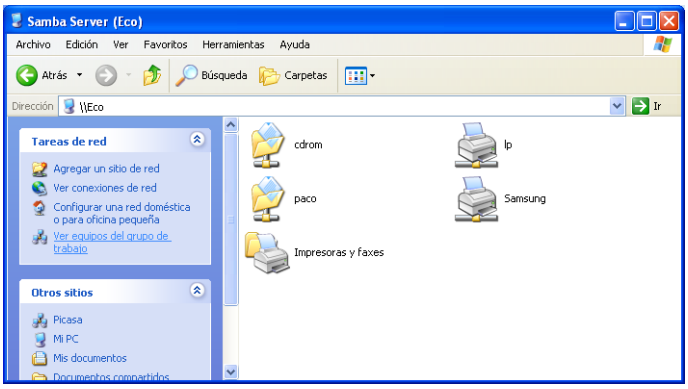
```
[printers]
comment = All Printers
path = /var/spool/samba
browseable = no
public=yes
guest ok = yes
writable = no
printable = yes
```

En Guadalinex no existe el directorio `/var/spool/samba` así que tenemos dos opciones:

- Usar `/var/tmp`
- Crearlo con los permisos adecuados

```
#mkdir /var/spool/samba
#chmod 1777 /var/spool/samba
```

En cuanto a los ficheros de configuración de CUPS, hemos de permitir que los trabajos de impresión se manden en "bruto", para eso, descomentaremos las líneas:
`application/octet-stream`
del fichero `/etc/cups/mime.types`, y
`application/octet-stream  application/vnd.cups-raw  0  -`
del fichero `/etc/cups/mime.convs`
Una vez en el Windows y tras acceder como un usuario registrado de la máquina Linux, instalaremos la impresora sin más dificultad, ya que es igual que si la red fuese sólo de equipos Windows.



O sea que: **se puede montar una unidad de Red en Windows usando Linux, instalar programas sobre esa unidad, cortar, copiar, pegar, imprimir, etc sin que el cliente Windows se entere; usando la seguridad, precio y potencia de un Linux.**

➥ Para practicar

Vamos a partir de que en la máquina Linux hay un usuario genérico de nombre INVITADO, ajustarlo a vuestro caso particular.

1. Instalar el servidor samba y conseguir que:
   a) El usuario INVITADO pueda acceder desde Windows al equipo Linux.
   b) El acceso esté limitado a la red local.
   c) Haya un recurso compartido público de sólo lectura (`/home/samba`) que sea accesible por todos los equipos de la red. Para eso:
   ```
   # mkdir /home/samba
   # chmod 755 /home/samba
   ```
   y en `/etc/samba/smb.conf`:
   ```
   security=share
   ```
   y además
   ```
   [public]
     comment = Directorio Público
     path = /home/samba
     public =yes
     writable = no
     printable = no
   ```
2. Construir una sección personalizada de nombre `[Practicas]` para el fichero `smb.conf` de manera que el directorio `/home/practicas` sea un recurso compartido de sólo lectura para el grupo `clase`. Además, deberá poder verse (la carpeta que da acceso a él) desde el navegador de windows por todos los usuarios del sistema si bien no podrán acceder a él.
3. En esta práctica vamos a conseguir que un CD se monte (y se desmonte) de forma automática cuando se accede a él desde un cliente Windows, además de ser un recurso accesible para toda la red. Sólo comentaremos las directivas nuevas:

```
[cdrom]
browseable = yes
#Activamos el soporte de bloqueos oportunistas por la-
do del cliente
oplocks = yes
guest ok = yes
#Hay que adecuar a nuestro sistema el punto de montaje
#En Guadalinex será /cdrom
path = /mnt/cdrom
#Comando a ejecutar antes de conextarse al recurso.
root preexec = /bin/mount -t iso9660 /dev/cdrom /mnt/cdrom
#Comando ejecutado al desconectarse del recurso
root postexec = /bin/umount /dev/cdrom
```

smb.conf de ejemplo

```
[global]
    workgroup = MYGROUP
    server string = Samba Server
    netbios name = Linux
    ; hosts allow = 192.168.1. 127.

    load printers = yes
    printing = cups
    cups options = raw
    printcap name= cups

[homes]
```

```
   comment = Directorios de usuario
   browseable = no
   writable = yes

[printers]
   comment = All Printers
   path = /var/spool/samba
   browseable = no
   guest ok = no
   writable = no
   printable = yes

[compartido]
  path = /home/compartido
  comment = Directorio compartido en Guadalinex 2004
  writeable = no
  guest ok = yes
  guest only = yes
  browseable = yes
```

Listado 14.2: /etc/samba/smb.conf

Capítulo 15

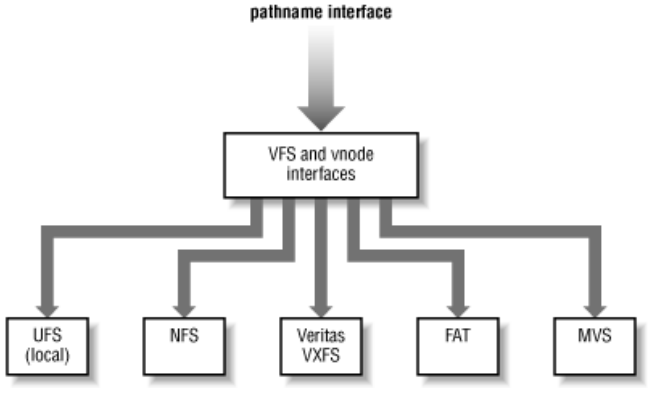
Servicio de compartición de ficheros NFS

> Cuando trabajas con Linux estás ante un sistema operativo orientado al trabajo con redes de ordenadores. ¿Qué nos empuja a poder afirmarlo tan categóricamente? Ya te darás cuenta poco a poco. (*Manual Avanzado de Linux* de Raúl Montero Rivero, Ed. Anaya)

Un Sistema de Ficheros en Red (NFS[1]) es un método de compartir archivos entre máquinas de una red, de tal forma que tenemos la impresión de trabajar en nuestro disco duro local, cuando en realidad están en otro lugar de la red. En los sistemas Unix era la forma tradicional de compartir ficheros en red, pero SAMBA ha ido ocupando gran parte de su terreno, debido a la necesidad de comunicarse con máquinas Windows. La tendencia es a que surjan sistemas de ficheros accesibles a través de la red, como WebDAV o el reciente ZFS.

Un *servidor NFS* es el que ofrece uno o varios de sus sistemas de ficheros para que otros sistemas los puedan utilizar. El *cliente NFS* es el que monta un sistema de ficheros de un sistema remoto sobre su sistema de ficheros local, accediendo a él como si estuviera en sus discos.

El sistema de ficheros remoto se integra en la jerarquía local y las aplicaciones accederán a él de forma transparente, sin darse cuenta de que están en una ubicación remota. Esa independencia le hace acceder a distintos sistemas de ficheros, ya sean ext3, FAT, NTFS, UFS o VXFS de la misma manera, a través de una entrada en el árbol de directorios.



15.1. Servidor NFS

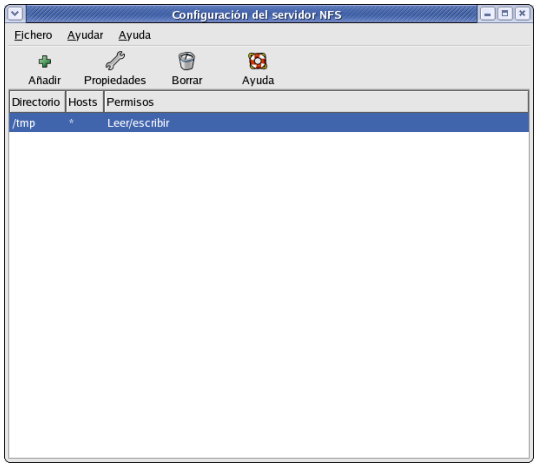
Compartir archivos desde un servidor NFS es conocido como exportar directorios. Para disponer de los servicios NFS tenemos que tener activos (en Fedora) los demonios `nfsd` y `mountd` que forman

[1] *Network File System*

parte del paquete `nfs-utils`[2].

La herramienta de configuración del servidor NFS se puede usar para configurar un sistema como servidor NFS.

Para usar la Herramienta de configuración del servidor NFS, desde el entorno gráfico, debe tener el paquete RPM `system-config-nfs` instalado. Para iniciar la aplicación, seleccione Botón de menú principal (→ **Configuración del sistema** → **Configuración de servidores** → **Servidor NFS**, o escriba el comando `system-config-nfs`.



Para añadir una partición NFS, pulse el botón Añadir. Aparecerá un cuadro de diálogo con tres pestañas: **Básico**, **Opciones generales** y **Acceso al usuario**.

La pestaña Básico necesita que le aportemos la siguiente información:

Directorio Especifique el directorio a compartir, por ejemplo `/tmp`.

Host(s) Especifique el o los hosts con los que compartir el directorio.

Permisos básicos. Especifique si el directorio deberá tener permisos de sólo lectura o de lectura y escritura.

Las opciones de las siguientes pestañas normalmente no serán utilizadas. La siguiente figura muestra la pestaña de *Acceso al usuario.*

[2]En Debian

```
#apt-get install nfs-common portmap nfs-kernel-server
```

Después de **Aceptar** la opción de añadir, modificar o eliminar un directorio compartido[3] mediante NFS desde la lista, los cambios tendrán efecto inmediatamente. El demonio del servidor es reiniciado por la interfaz gráfica y el archivo de configuración antiguo es guardado como `/etc/exports.bak`. La nueva configuración es escrita a `/etc/exports`. También podemos modificar el archivo de forma manual.

15.1.1. Fichero /etc/exports

El archivo `/etc/exports` controla qué directorios exporta el servidor NFS. Su formato es como se muestra a continuación:

```
# more /etc/exports
/tmp *.midominio.org(rw,sync)
/usr/local thales(ro) mileto(ro)
/ pitagoras(rw,no_root_squash)
```

Pasemos a comentarlo. Se permite montar el directorio `/tmp` a los hosts de `midominio.org` con permisos de lectura/escritura. El directorio `/usr/local` lo podrán montar las máquinas thales y mileto con permisos de sólo lectura y el directorio raíz, se podrá montar desde la máquina pitágoras con permisos de lectura y escritura. No muy recomendale esto último.

Formato del nombre de host

El nombre de host puede ser de alguna de las siguientes maneras:

Máquina única Nombre de dominio completamente cualificado (thales.cica.es), nombre del host que puede ser resuelto por el servidor (thales, sabiendo que se encuentra en el dominio cica.es) o dirección IP.

Series de máquinas especificadas con comodines. Se usa el caracter * o ? para especificar una cadena de caracteres que coincida. Por ejemplo, 192.168.100.* especifica cualquier dirección IP que comience con 192.168.100. Cuando se usan comodines en nombres de dominio completos, los puntos (.) no son incluidos en el comodín. Por ejemplo, *.cica.es incluye thales.cica.es pero no incluye sirio.sistemas.cica.es.

Redes IP. Se usa el formato a.b.c.d/z, donde a.b.c.d es la red y z es el número de bits en la máscara de red (por ejemplo 192.168.0.0/24). Otro formato aceptable es a.b.c.d/netmask, donde a.b.c.d es la red y netmask es la máscara de red (por ejemplo, 192.168.100.8/255.255.255.0).

Flags

Sin ser exhaustivos, comentaremos algunos valores de flags que podemos tener tras el nombre de host.

ro El sistema de ficheros se montará en modo de sólo lectura.

rw El sistema de ficheros se montará en modo de lectura/escritura.

[3] En inglés *share*

root_squash El superusuario del sistema cliente (el que monta el sistema de ficheros) no tendrá privilegios especiales sobre el sistema de ficheros que se monte.

no_root_squash Lo contrario del anterior, el supersuario sigue siendo el "jefe" incluso en los ficheros remotos.

Para ver el estado del demonio NFS desde Fedora utilizamos el siguiente comando:

```
/sbin/service nfs status
```

Lo arrancamos como de costumbre:

```
/sbin/service nfs start
```

En Debian, para reiniciar el servicio usaremos:

```
/etc/init.d/nfs-kernel-server restart
```

15.1.2. RPC y portmap

NFS se apoya en las llamadas de procedimientos remotos (RPC[4]) para funcionar. Se requiere del demonio `portmap` para enlazar las peticiones RPC a los servicios concretos. Los procesos RPC notifican a `portmap` cuándo comienzan, indicando el número de puerto en el que están esperando y el número de programas RPC que esperan servir. En definitiva, `portmap` es un demonio que se encarga de controlar puertos y de asignar programas RPC a dichos puertos.

El sistema cliente contacta con el `portmap` del servidor con un número de programa RPC particular. Entonces `portmap` redirecciona al cliente al número del puerto apropiado para que se comunique con el servicio adecuado.

Como `portmap` proporciona la coordinación entre servicios RPC y los números de puertos usados para comunicarlos, es útil poder visualizar el estado de los servicios RPC actuales usando `portmap` cuando estamos resolviendo algún problema. El comando `rpcinfo` muestra cada servicio basado en RPC con su número de puerto, número de programa RPC, versión y tipo de protocolo (TCP o UDP).

Para asegurarnos de que los servicios NFS basados en RPC están activos para `portmap`, podemos utilizar el comando `rpcinfo -p`:

```
# rpcinfo -p
programa vers proto puerto
100000 2 tcp 111 portmapper
100000 2 udp 111 portmapper
100024 1 udp 1024 status
100024 1 tcp 1026 status
391002 2 tcp 1027 sgi_fam
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100003 2 tcp 2049 nfs
100003 3 tcp 2049 nfs
100021 1 udp 1046 nlockmgr
100021 3 udp 1046 nlockmgr
100021 4 udp 1046 nlockmgr
100021 1 tcp 1856 nlockmgr
100021 3 tcp 1856 nlockmgr
100021 4 tcp 1856 nlockmgr
100011 1 udp 784 rquotad
100011 2 udp 784 rquotad
100011 1 tcp 787 rquotad
100011 2 tcp 787 rquotad
```

[4] *Remote Procedure Call*

```
100005 1 udp 792 mountd
100005 1 tcp 795 mountd
100005 2 udp 792 mountd
100005 2 tcp 795 mountd
100005 3 udp 792 mountd
100005 3 tcp 795 mountd
```

15.2. Cliente NFS

El comando mount es el utilizado para montar directorios de NFS compartidos desde otra máquina, al igual que se montaría un sistema de ficheros local:

```
#mount thales:/tmp /mnt/tmp
```

Donde `thales:/tmp` significa el directorio `/tmp` de la máquina thales que montaremos bajo el directorio `/mnt/tmp` de la máquina local. El directorio `/mnt/tmp` de la máquina local debe existir y tener los permisos adecuados para el usuario que intenta montarlo.

Una vez hayamos ejecutado el comando `mount` (siempre que tengamos los permisos adecuados en el servidor thales), podremos teclear `#ls /mnt/tmp` y obtener un listado de los archivos que se encuentran en el directorio `/tmp` de la máquina thales.

Para que un cliente pueda montar sistemas de fichero remotos mediante NFS, debe soportar este tipo de sistema de ficheros en el kernel. Podemos comprobarlo mediante:

```
$cat /proc/filesystems
nodev rootfs
nodev bdev
nodev proc
nodev sockfs
nodev tmpfs
nodev shm
nodev pipefs
ext2
nodev ramfs
iso9660
nodev devpts
ext3
nodev usbdevfs
nodev usbfs
nodev autofs
vfat
nodev nfs
```

Las siguientes órdenes nos muestran cómo se montaría un sistema de ficheros NFS.

```
#mount -t nfs linux:/tmp /mnt/tmp
```

La apariencia del sistema de ficheros montado junto a los sistemas locales.

```
[root@linux images]# df
S.ficheros Bloques de 1K Usado Dispon Uso% Montado en
/dev/hda2 2276952 2048220 113064 95% /
none 62988 0 62988 0% /dev/shm
linux:/tmp 2276952 2048220 113064 95% /mnt/tmp
```

Desmontamos el sistema de ficheros

```
[root@linux images]# umount /mnt/tmp
```

15.2.1. Montar sistemas de archivos NFS usando /etc/fstab

Un método alternativo para montar datos compartidos mediante NFS es añadir una línea en el archivo `/etc/fstab`. La línea debe incluir el nombre del servidor NFS, el directorio que el servidor está exportando y el directorio de nuestra máquina local donde queremos montar el sistema de

archivos. Recordad que debéis tener permisos de superusuario para poder modificar el archivo `/etc/fstab`.

La sintaxis general de esta línea del archivo `/etc/fstab` es la siguiente:

```
server:/tmp /mnt/tmp nfs rsize=8192,timeo=14,intr
```

Los valores de opciones que podemos utilizar son:

`rsize=n,wsize=n` Especifican el tamaño del datagrama[5] utilizado por los clientes en las peticiones de lectura y escritura, respectivamente.

`timeo=n` Especifica el límite en décimas de segundo que el cliente esperará que una petición se complete. Lo que ocurre después de un timeout (agotamiento del tiempo) depende de si hemos utilizado la opción `hard` o la opción `soft`.

`hard` Es la opción por defecto. El cliente, si no puede contactar con el servidor, presenta un mensaje por pantalla y continúa intentando indefinidamente la operación.

`soft` Causa que al ocurrir un timeout, la operación falle con un error de entrada/salida y no se reintente más.

`intr` Permite mandar señales de interrupción a la llamada NFS. Es útil para abortar la operación cuando el servidor no responde.

[5] Sí, NFS utiliza UDP, por eso son datagramas. Las nuevas versiones empiezan a incorporar TCP.

Capítulo 16

Servicio de Proxy-caché

> -Como me quieres bien, Sancho, hablas desa manera -dijo don Quijote-; y, como no estás experimentado en las cosas del mundo, todas las cosas que tienen algo de dificultad te parecen imposibles; pero andará el tiempo, como otra vez he dicho, y yo te contaré algunas de las que allá abajo he visto, que te harán creer las que aquí he contado, cuya verdad ni admite réplica ni disputa.(*El ingenioso hidalgo Don Quijote de la Mancha*. Miguel de Cervantes Saavedra).

16.1. ¿Qué es un proxy caché?

Existen dos métodos para que los usuarios de nuestra red naveguen por internet. El primer método implica que los usuarios tengan sus ordenadores conectados directamente a internet. Los navegadores solicitan directamente las páginas a los servidores web remotos. Este método es el utilizado cuando navegamos por internet desde un ordenador que está conectado por módem, cable-módem o ADSL directamente.

El otro método implica un almacenamiento (caché) de las páginas que se visitan. Siempre que un usuario quiera visitar una página web, el navegador se conectará a un servidor de caché que le suministrará la página (caso de tenerla almacenada) o la solicitará al servidor web remoto (caso de no disponer de ella).

En caso de optar por la segunda opción, puede reducirse considerablemente el tiempo de descarga de una página. Las páginas que se encuentran en caché no requieren acceso al servidor remoto y las páginas que no están en caché no introducen un tiempo extra en la descarga, al disponer el servidor de todo el ancho de banda para descargar las páginas que no están en caché.

16.2. Squid, un proxy caché para Linux

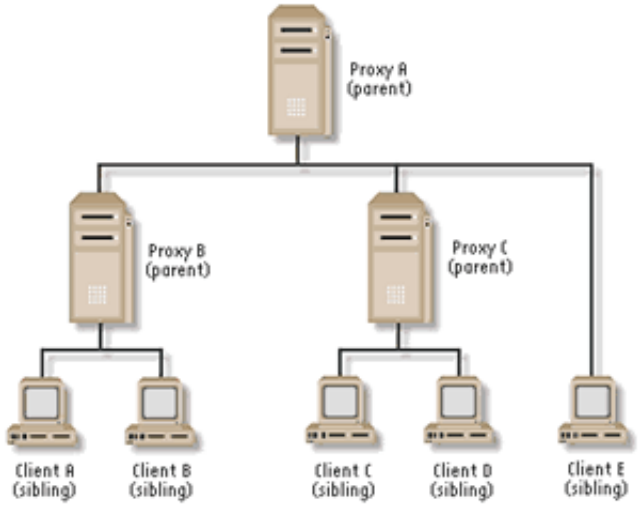
16.2.1. Visión general

Como alternativa al software comercial existente, apareció Squid. Su funcionamiento se basa en guardar las peticiones que hacen los usuarios a servidores web remotos. Cuando un usuario quiere acceder a una página la solicita a Squid, que se encarga de acceder al servidor web remoto. Una vez obtenida, la reenvía al usuario, guardando una copia. En el caso que otro usuario solicite de nuevo esa página, únicamente tendrá que recuperarla de su disco local y servirla.

Otra función que realiza Squid es la de proporcionar un servicio de proxy a ordenadores que necesiten acceder a internet a través de algún tipo de cortafuegos. Por eso es común denominar a Squid como un proxy caché, al unir las dos funcionalidades que presenta.

Squid puede almacenar datos de los protocolos HTTP, FTP, Gopher y DNS. El tener un servidor de caché especializado puede reducir considerablemente el uso que se haga del ancho de banda disponible. En lugar de descargar páginas repetidamente, se comprobará si la página del

Figura 16.1: Jerarquía de proxy



servidor remoto es más nueva que la que tiene almacenada en disco. De no ser así, no se molestará en descargarla.

16.2.2. Conceptos sobre cachés

Los servidores que actúan de proxy-caché se pueden configurar de varias formas. La forma más simple es un solo servidor proxy-caché en la red en el que todos los ordenadores pertenecientes a esa red accederán a este servidor, que será el que almacenará todos los datos. Cuando un usuario solicita al servidor una página, éste comprueba si fue actualizada desde que fue almacenada. Si tiene la versión actualizada ahorra al usuario final la descarga de la misma proporcionándosela directamente.

Otro método de configurar la salida a internet de una red de ordenadores es creando una jerarquía de servidores proxy-caché. Los servidores en un nivel superior a un servidor son denominados padres (*parent*) y los que se encuentran al mismo nivel son hermanos o iguales (*sibblings*, *neighbor* o *peer*).

Cuando Squid obtiene una petición de un cliente, comprueba si el objeto solicitado (página, gráfico o fichero) está en el disco del servidor. Si está, comprueba que el objeto no está caducado y procede a enviarlo al cliente. Si, por el contrario, el objeto no está o ha caducado, comprueba que otras cachés (padres o hermanas) lo tengan. Lo hace a su vez enviando paquetes UDP a esas máquinas con la URL.

La otra caché comprueba, a continuación, si tiene dicho objeto en el disco duro y envía un mensaje indicando si lo posee o no. La máquina original espera las respuestas y después decide si debe obtener el objeto de la otra caché o debe ir directamente a por él.

Cuando existe una máquina hermana el servidor le solicitará la información que no tiene y en caso de no tenerla estos servidores accederá directamente al servidor web remoto. En caso que existan también servidores padre en la configuración, la información que no tengan los hermanos la solicitará al servidor padre, que a su vez la solicitará directamente al servidor web remoto.

16.2.3. Instalación

Si no estuviera instalado, lo instalamos a partir del paquete RPM o DEB correspondiente.

Fedora: `apt-get install squid`

y una vez instalado optar porque se inicie en el arranque

```
# ntsysv
```

Guadalinex: `apt-get install squid`

Los ficheros y directorios más importantes son:

- En el directorio `/etc/squid` se guardan los ficheros de configuración. Específicamente en el fichero `squid.conf` se encuentra la mayor parte de ella.
- Una parte importante de ficheros se encuentran en `/usr/lib/squid`, pero no tendremos que preocuparnos de ellos por ahora.
- La documentación se encuentra en `/usr/share/doc/squid-x.x.x/`
- En `/var/spool/squid` se van a encontrar las páginas "cacheadas", es decir, las traídas desde Internet y que se almacenan para la próxima vez que las solicite alguien y no hayan cambiado.
- En `/var/log/squid` se guardan los accesos de nuestros usuarios a Internet a través del proxy, así como los posibles errores que hayan ocurrido.

16.3. Configuración de Squid

16.3.1. Configuración básica

El archivo de configuración que utiliza Squid es `squid.conf`, como hemos dicho se encuentra situado en la ruta `/etc/squid/squid.conf`, pudiendo encontrarse en otras localizaciones dependiendo de la instalación. Este archivo está ampliamente comentado por lo que no lo analizaremos de forma detallada, sino que haremos un rápido recorrido por el fichero de configuración centrándonos en los aspectos que consideremos más importantes.

Una de las primeras directivas de configuración que aparece es:

```
http_port 3128
```

Indica el puerto en el que va a estar escuchando Squid.

A continuación podemos ver el parámetro que indica el puerto en el que Squid escucha las peticiones ICP[1].

```
icp_port 3130
```

Como ya hemos descrito, Squid es un proxy-caché y pueden existir elementos que no queramos almacenar. Esto puede conseguirse a través del fichero de configuración. Con la siguiente línea no almacenaríamos en caché ningún objeto que se encuentre en la ruta `cgi-bin`:

```
acl QUERY urlpath_regex cgi-bin \?
no_cache deny QUERY
```

Posteriormente veremos con más detalle la sintaxis y usos de la directiva `acl` para la creación de clases.

Es posible también definir parámetros de uso de memoria. Si queremos definir la cantidad de memoria RAM que deseamos asignar a las funciones de Squid con un valor de 8 Mb[2]

```
cache_mem 8 MB
```

[1] Protocolo utilizado para comunicaciones entre distintos cachés

[2] Con las configuraciones actuales y en función del uso de nuestra máquina, en general, debemos optar por un valor mayor.

Es posible también definir cuando se empieza a eliminar archivos de la caché. Cuando la caché llega al total del porcentaje de `cache_swap_high` Squid comienza a eliminar los elementos almacenados menos utilizados hasta que llega al total del porcentaje `cache_swap_low`.

```
cache_swap_low 90
cache_swap_high 95
```

Otra alternativa para regular el caché es configurarlo para que no almacene archivos que tengan un tamaño mayor que el indicado:

```
maximum_object_size 4096 KB
```

Ya hemos comentado que el caché de Squid es un espacio en disco reservado para almacenar los distintos objetos que se piden a través del proxy. Será necesario definir el lugar donde se va a almacenar el caché[3].

```
cache_dir ufs /var/spool/squid 100 16 256
```

El formato genérico de esta directiva es:

```
cache_dir tipo directorio Mbytes L1 L2 [options]
```

- `tipo`. Tipo de sistema de almacenamiento a utilizar (ufs es el único que está definido por defecto en la instalación).
- `directorio`. Ruta del directorio que se va a utilizar para guardar los datos del caché.
- `Mbytes`. Cantidad de espacio en disco que se va a utilizar para el caché. Si queremos que utilice el disco entero es recomendable poner aquí un 20 % menos del tamaño.
- `L1`. Número de subdirectorios de primer nivel que serán creados bajo directorio.
- `L2`. Número de subdirectorios de segundo nivel que serán creados bajo cada subdirectorio de primer nivel.

Una consideración a tener en cuenta es que el contenido de este directorio va a cambiar con frecuencia, siendo recomendable colocarlo en una partición separada por varias razones:

- La caché podría sobrepasar al resto del sistema de archivos o de la partición que comparte con otros procesos.
- Cuanto más cambie un sistema de archivos, mayores son también las posibilidades de que se encuentre dañado. Mantener la caché en una partición limita la parte de su sistema completo de archivos que resulta dañado.

También es posible configurar la localización de los archivos de log así como la información general de la caché[4]:

```
cache_access_log /var/log/squid/access.log
cache_log /var/log/squid/cache.log
cache_store_log /var/log/squid/store.log
```

[3] El valor de 100MB es escaso para los discos duros actuales, un valor de 1GB puede ser mejor si nuestro disco lo permite.

[4] Los ficheros de contabilidad que deja, pueden ser monitorizados para impedir accesos a Internet no deseados. Un ejemplo real es el de un organismo que manda semanalmente a los usuarios de Internet un fichero con los accesos en ese periodo. El usuario se siente controlado y es más responsable con sus accesos.

Una herramienta sencilla de configurar y muy útil para obtener estadísticas sobre las páginas visitadas es `sarg`, la podéis conseguir de `http://web.onda.com.br/orso/sarg.html`. La analizaremos mejor al final de este capítulo.

16.3.2. Configuración de jerarquía de caché

De los anteriores conceptos sobre jerarquía de caché podemos deducir que el empleo de este tipo de infraestructura puede ser complicado y a veces puede no merecer la pena. Dependiendo del tipo de instalación y del tráfico que manejemos puede ser interesante su consideración.

Como ya hemos comentado, si configuramos el caché para tener sólo hermanos, dicha caché enviará las peticiones UDP a la lista de hermanos y si no poseen dicho objeto, Squid se conectará directamente al servidor web remoto.

En el caso de realizar una configuración del caché para tener un padre, significa que si dicha caché no tiene el objeto, y ninguna de las hermanas lo posee, abrirá una conexión TCP a la caché padre para que ésta obtenga el objeto. Al ser una conexión TCP, esta caché padre posiblemente recorrerá la lista de cachés padres y hermanas para buscar el objeto (sólo les envía una petición UDP para comprobar si la poseen en el disco duro). Lo que complica las cosas es tener múltiples cachés padres y hermanas.

Obviamente, si sólo un padre tiene el objeto, lo descargará de allí, pero si ninguna lo posee, su caché dará la petición a la máquina que respondió más rápida, suponiendo que es la máquina menos saturada o que posee una conexión más directa.

Existe otra opción que es balancear las cachés, repartiendo la carga. Incluso se puede especificar que use una caché padre para descargar las peticiones que no haya podido obtener.

Veamos ahora cómo pueden configurarse estas opciones en Squid, lo que nos aclarará un poco los conceptos.

```
cache_host cache.mordor.com parent 3128 3130
```

En este ejemplo sólo existe una caché a la que Squid va a preguntar. Sin embargo, podemos ajustar un poco más la configuración de forma que se conectará al caché padre para todas las peticiones, a diferencia del ejemplo anterior que necesitaba saber si estaba o no activa.

```
cache_host cache.mordor.com parent 3128 3130 no-query default
```

Otra característica muy útil de este sistema es la capacidad de cachés hermanas. Supongamos que no desea gastar mucho dinero en una sola caché, y desea balancear la carga entre varias máquinas, pero no duplicando los objetos en cada máquina. Es posible configurar estas máquinas para que hablen entre ellas mediante las señales `proxy-only`, como este ejemplo:

Configuración en caché 1

```
cache_host cache2.gondor.com sibling 3128 3130 proxy-only
```

Configuración en caché 2

```
cache_host cache1.rohan.com sibling 3128 3130 proxy-only
```

En este caso, si una petición se dirige a la caché 1 y no se encuentra en el disco, esta caché enviará una petición ICP a la caché 2 (3, 4, etc) y la descargará de allí, pero no la salvará en su disco duro, tan solo la descargará de la otra caché cuando la necesite de nuevo. Este sistema duplicará la capacidad del disco duro sin necesidad de gastar grandes cantidades de dinero en sistemas raid para soportar muchos gigas.

16.3.3. Control de acceso

Otro aspecto importante en la configuración de Squid son las listas de control de acceso o ACL[5]. Una de sus principales funciones es la de permitir o denegar el acceso a la caché, aunque no se queda aquí. Las ACL pueden usarse también para definir las jerarquías de caché.

[5] *Access Control List*

El procedimiento que se sigue es definir las distintas ACL y posteriormente se permite o deniega el acceso a una determinada función de la caché. La opción de configuración encargada es `http_access`, que permite o deniega al navegador web el acceso a Squid.

Es importante tener en cuenta que Squid lee las directivas de arriba a abajo para determinar qué regla aplicar.

Veamos un primer ejemplo de uso. Supongamos que disponemos de una red de clase C (con direcciones IP dentro de la red 172.26.0.0) y que solo queremos permitir el acceso a internet a través de Squid a estas máquinas.

```
acl hostpermitidos src 172.26.0.0/255.255.255.0
acl all src 0.0.0.0/0.0.0.0
http_access allow hostpermitidos
http_access deny all
```

Las dos primeras líneas de este ejemplo crean las ACL `hostpermitidos` y `all`. El formato general de esta directiva es:

```
acl nombreACL tipoACL cadena ...
acl nombreACL tipoACL "fichero" ...
```

donde:

- `nombreACL` es el nombre que corresponde a esta definición ACL
- `tipoACL` es el tipo de elemento contenido en esta definición
- `cadena` o `fichero` es el argumento apropiado al `tipoACL`, pudiendo haber más de un argumento en la lista

Cuando utilicemos un fichero como origen de los datos de la ACL deberá contener un elemento por línea. Algunos de los tipos de ACL que podemos utilizar en esta directiva son:

- Direcciones IP de los clientes. Especifica la dirección IP local, dirección de red o rango de direcciones a buscar

  ```
  acl nombreACL src dirIP/máscara
  acl nombreACL src dirIP1-dirIP2/máscara
  ```

- Dirección IP de la URL destino. Especifica la dirección IP de la máquina remota, dirección de red o rango de dirección a buscar

  ```
  acl nombreACL dst dirIP/máscara
  acl nombreACL dst dirIP1-dirIP2/máscara
  ```

- Dominio de la máquina cliente. Especifica el `host.dominio.extensión` o bien el `dominio.extensión` a buscar

  ```
  acl nombreACL srcdomain nombreDominio
  ```

- Dominio de la máquina destino. Especifica el `host.dominio.extensión` o bien el `dominio.extensión` a buscar

  ```
  acl nombreACL dstdomain nombreDominio
  ```

- Expresión regular que concuerda con el nombre del cliente

  ```
  acl nombreACL srcdom_regex [-i] expresión
  ```

- Expresión regular que concuerda con el nombre del servidor

```
acl nombreACL dstdom_regex [-i] expresión
```

- Control por día y hora. Especifica la información de tiempo a buscar

```
acl nombreACL time [día] [h1:m1-h2:m2]
M - Lunes
T - Martes
W - Miércoles
H - Jueves
F - Viernes
A - Sábado
S - Domingo
h1:m1 < h2:m2
```

- Expresión regular que concuerda con la URL completa

```
acl aclname url_regex [-i] ^http://expresión
```

- Expresión regular que concuerda con la ruta de la URL

```
acl aclname urlpath_regex [-i] \.gif$
```

La primera de las opciones permite decidir en qué ACL se encuentra la dirección IP del usuario. También podemos decidir sobre aspectos como el tiempo actual o el sitio al que se dirigen. Para más información sobre estos y otros tipos de ACL recomendamos acceder al fichero `/etc/squid/squid.conf`.

Una vez definidas las ACL entra en juego la directiva `http_access`, que se utiliza para permitir o denegar el acceso de una determinada ACL, siempre y cuando el cliente utilice el método HTTP para solicitar el objeto al servidor web remoto.

Hay que tener en cuenta que la lectura se realiza de arriba hacia abajo, y se para en la primera coincidencia para decidir si permitir o denegar la petición. En el ejemplo anterior, Squid verá que la primera línea `http_access` se cumple, y procederá a aplicarla. En este caso, permitirá el acceso y ejecutará la petición.

Pero tengamos en cuenta el siguiente ejemplo:

```
acl hostpermitidos src 172.26.0.0/255.255.255.0
acl all src 0.0.0.0/0.0.0.0
http_access deny all
http_access allow hostpermitidos
```

En este caso no funcionará, ya que Squid aplicará la primera coincidencia (la primera línea) y denegará el acceso.

Las ACL son especialmente útiles cuando queremos prohibir el acceso a una lista de sitios inapropiados (enlaces a páginas web con contenido pornográfico, ...). Squid no está optimizado para gestionar una larga lista de sitios, pero puede gestionar un número concreto de sitios sin problemas.

```
acl adultos dstdomain playboy.com sex.com
acl hostpermitidos src 172.26.0.0/255.255.255.0
acl all src 0.0.0.0/0.0.0.0
http_access deny adultos
http_access allow hostpermitidos
http_access deny all
```

En este caso, las direcciones que serán consideradas como inadecuadas son las que tienen los dominios `playboy.com` o `sex.com`. Estas URL tendrán filtrado el acceso y no será posible acceder a ellas, tal como indica la directiva `http_access deny adultos`. Si se piden otras URL, Squid pasará a evaluar la siguientes directivas. Por tanto, si el cliente se conecta dentro del rango permitido se cursará la petición. De lo contrario, la petición será rechazada.

La configuración que viene por defecto es denegar todos los accesos, mediante:

```
http_access deny all
```

Obviamente deberemos al menos incluir algún grupo que pueda acceder, porque si no, nuestro proxy sería innecesario.

Una última observación, este método considera exclusivamente los dominios. Para evitar conexiones especificando la IP de la máquina, utilizaremos la directiva `dst acl`.

Suponiendo que ya hemos definido nuestra política de acceso, arrancamos el servicio

```
#/etc/init.d/squid start
```

La primera vez que lo ejecutemos tardará un ratito porque tiene que construir sus índices para el almacenamiento de páginas.

```
root@guadalinex:/usr/lib/squid# /etc/init.d/squid start
Starting proxy server: Creating squid spool directory structure
2005/02/15 14:18:23| Creating Swap Directories
squid.
root@guadalinex:/usr/lib/squid#
```

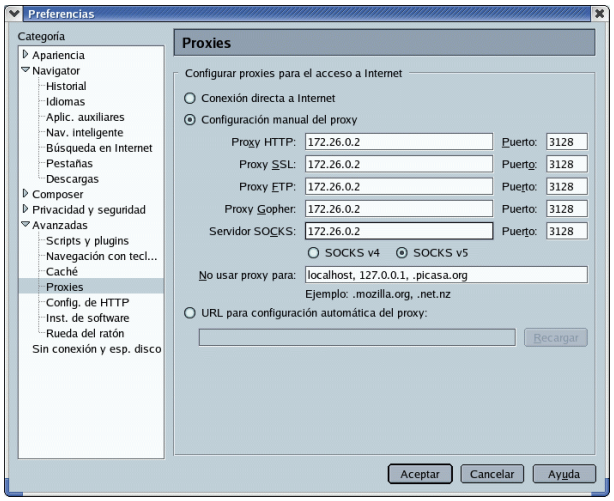
En función de la versión de squid puede que nos aparezca un error al ponerlo en marcha, en general se debe a que hay que configurar correctamente la directiva `visible_hostname` (aparece comentado por defecto).

16.4. Configuración de los clientes

El cliente lo configuraremos de la siguiente forma:

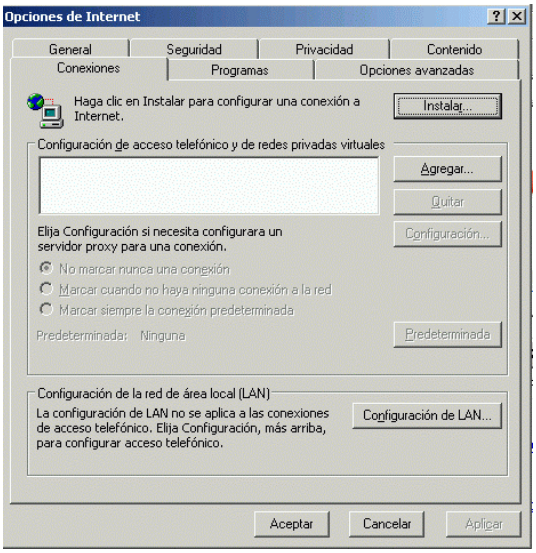
- Si es Mozilla, seleccionamos **Editar→Preferencias→Avanzadas→Proxy→Configuración Manual del Proxy,** pulsamos **Ver** y en los distintos protocolos ponemos el host[6] 176.26.0.2 por el puerto 3128 que es por el que escucha el squid peticiones de sus clientes. Podemos ponerlo en todos los protocolos menos en el socks.

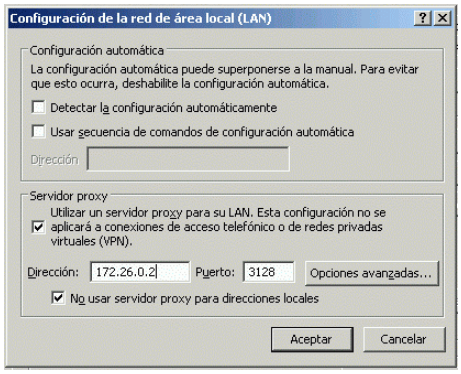
[6]Suponemos que el proxy-caché está a la escucha en la máquina de IP 176.26.0.2



Veremos que nuestros accesos a Internet son mucho más rápidos y el control que podemos llegar a tener es muy importante.

- Si es Explorer optaremos por seleccionar **Configuración de LAN...** y una vez allí marcar la IP de la máquina con el proxy squid, el puerto (por defecto 3128) y mejor si no usamos el proxy para direcciones locales





- Si trabajamos en modo consola y deseamos definir la variable de entorno[7] `http_proxy`, podemos usar:

  ```
  export http_proxy="http://ip_proxy:3128"
  ```

  Para ver que todo está bien:

  ```
  lynx http://www.iesmurgi.org
  ```

  Es útil, por ejemplo, para actualizar un sistema com `yum` o `apt-get` que accede a internet a través de un proxy

[7] Si añadimos la variable a algún script de arranque se tomará como valor por defecto. Desde GNOME o KDE también podemos configurarla.

16.5. Acceso a internet autenticado contra ldap

16.5.1. Métodos de autenticación de Squid

Squid tiene un potente conjunto de utilidades que permiten la autenticación del proxy. Mediante la autenticación, las peticiones HTTP de los clientes contienen una cabecera que incluye las credenciales de autenticación. Estas credenciales suelen consistir en una pareja usuario/clave. Squid decodifica esta información y posteriormente realiza una consulta a un proceso de autenticación externo, el cual se encarga de la verificación de las credenciales.

Squid soporta varias técnicas de autenticación:

- Basic
- Digest
- NTLM

Nos centraremos en la autenticación básica. Aunque es una técnica insegura debido a que la pareja usuario/clave va a viajar en claro por la red, nos servirá de aproximación al resto de métodos, algo más complicados de implementar.

Ya vimos en un apartado anterior como configurar ldap y aprovecharemos ahora esta base de datos de usuario para la autenticación. Utilizaremos los programas de ayuda que proporciona Squid, más concretamente `/usr/lib/squid/ldap_auth`[8].

Las directivas de Squid encargadas de definir los parámetros del mecanismo de autenticación son `auth_param` y `proxy_auth`. Es muy importante el orden en que se ponen estas directivas en el fichero de configuración.

Será necesario definir al menos un método de autenticación con `auth_param` antes que ninguna ACL definida con `proxy_auth` haga referencia al mismo. En caso contrario Squid mostrará un mensaje de error e ignorará la ACL definida con `proxy_auth`, aunque la ejecución de Squid seguirá su curso. La directiva `proxy_auth` tomará los nombres de usuario como valores por defecto, aunque la mayoría de las veces únicamente se utilizará en su definición `REQUIRED`:

```
auth_param ...
acl autenticacion proxy_auth REQUIRED
```

En este caso, cualquier petición con credenciales válidas verifica la ACL. En el caso que necesitemos un control más fino es cuando usaremos los nombres de usuario:

```
auth_param ...
acl autenticacion1 proxy_auth usuario1 usuario2
acl autenticacion2 proxy_auth usuario3 usuario4 usuario5
```

Anteriormente configuramos nuestro propio LDAP y ahora es un buen momento para utilizarlo[9]. La configuración del módulo de autenticación dentro de Squid quedaría:

```
auth_param basic program /usr/lib/squid/ldap_auth -b ou=personas,dc=
    midominio,dc=org -D cn=Manager,dc=midominio,dc=org -W /etc/ldap.secret -
    f uid=%s -v 3 192.168.0.50
auth_param basic children 5
auth_param basic realm Squid proxy-caching web server
auth_param basic credentialsttl 2 hours
```

[8] En versiones anteriores de Squid este módulo se llamaba `squid_ldap_auth`. Cuando ejecutamos `ldap_auth` sin ningún argumento nos muestra una ayuda en la que aún se hace referencia a este programa.

[9] Utilizaremos el árbol LDAP creado para la autenticación mediante módulos PAM, siendo el `uid` el atributo que contendrá el identificador de usuario.

El fichero `/etc/ldap.secret` contendrá la contraseña que se estableció para el usuario `Manager` al crear nuestro LDAP[10]. El resto de opciones pueden obtenerse de la ejecución de `ldap_auth` sin ningún argumento:

```
Usage: squid_ldap_auth -b basedn [options] [ldap_server_name[:port]]...
        -b basedn (REQUIRED)     base dn under which to search
        -f filter                search filter to locate user DN
        -u userattr              username DN attribute
        -s base|one|sub          search scope
        -D binddn                DN to bind as to perform searches
        -w bindpasswd            password for binddn
        -W secretfile            read password for binddn from file
            secretfile
        -H URI                   LDAPURI (defaults to ldap://localhost)
        -h server                LDAP server (defaults to localhost)
        -p port                  LDAP server port
        -P                       persistent LDAP connection
        -c timeout               connect timeout
        -t timelimit             search time limit
        -R                       do not follow referrals
        -a never|always|search|find
                                 when to dereference aliases
        -v 2|3                   LDAP version
        -Z                       TLS encrypt the LDAP connection, requires
            LDAP version 3
        If no search filter is specified, then the dn <userattr>=user,basedn
        will be used (same as specifying a search filter of '<userattr>=',
        but quicker as as there is no need to search for the user DN)
        If you need to bind as a user to perform searches then use the
        -D binddn -w bindpasswd or -D binddn -W secretfile options
```

Una vez modificada la configuración de Squid es necesario reiniciar el servicio para que los cambios tengan efecto con `/etc/init.d/squid restart`[11].

Existe una forma de comprobar la correcta configuración del módulo mediante la ejecución del mismo desde la línea de comandos:

```
root@guadalinex:~# /usr/lib/squid/ldap_auth -b ou=personas,dc=midominio,dc=
    org -D cn=Manager,dc=midominio,dc=org -W /etc/ldap.secret -f uid=%s -v 3
     192.168.0.50
jose.fernandez hola
OK
jose.fernandez holaa
ERR
```

En el ejemplo anterior estamos comprobando la configuración usando al usuario `jose.fernandez` con clave `hola`. En caso de introducir mal la clave o que el usuario no exista la salida será `ERR`, por el contrario, en una autenticación correcta obtenemos `OK`.

Ya tenemos correctamente configurado el módulo, pero aún no hemos dicho como queremos utilizarlo. Ahora entra en juego la definición de la ACL siguiendo las indicaciones previas:

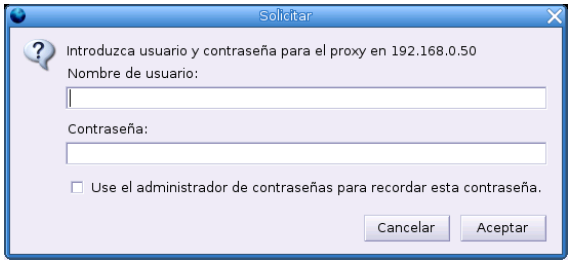
```
acl ldap proxy_auth REQUIRED
...
http_access allow ldap
```

[10] Hay ocasiones en que la opción -W no funciona correctamente por lo que no obtendremos la clave almacenada en él. Bastará con sustituir este parámetro `-W <fichero_clave_manager>` por `-w <clave_manager>`.

[11] Aunque el tiempo necesario es mínimo, si no queremos parar la instancia de Squid que se encuentra activa podemos utilizar `squid -k reconfigure` que actualiza la configuración sin parar Squid.

Con esta restricción cada vez que accedamos a una página de internet aparecerá la siguiente pantalla en la que deberemos introducir nuestro nombre de usuario y clave, según lo hayamos definido en LDAP.

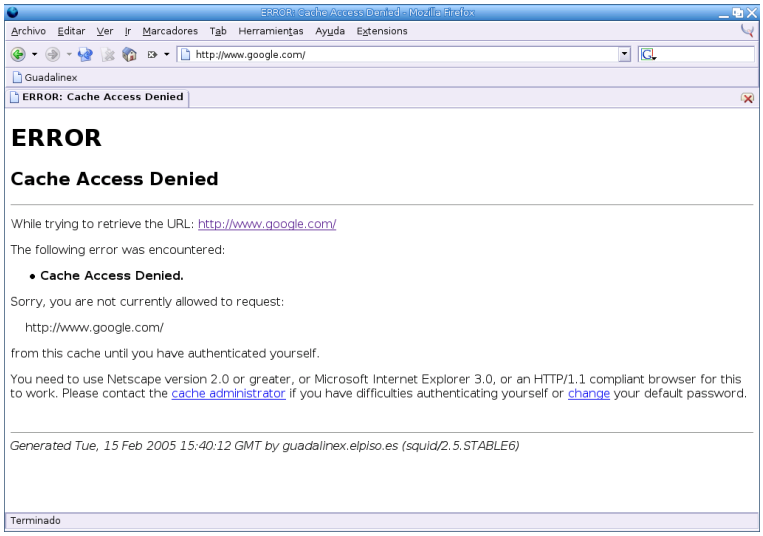
Figura 16.2: Autenticación de Squid



A partir de este momento, el proceso de autenticación se realizará únicamente para las nuevas instancias del navegador pero no para las que se creen a partir de la actual.

Cuando falle la autenticación se mostrará un mensaje de error informándonos al respecto.

Figura 16.3: Error en la autenticación



16.5.2. Analizador de logs SARG

El objetivo de restringir el acceso a internet es controlar que no se realice un uso ilegítimo del mismo. Mediante la configuración que acabamos de ver será necesario autenticarse antes de acceder a internet. Pero ¿cómo sabemos que el uso que se hace de este recurso es el correcto? Vamos ver qué información se está almacenando en los ficheros de log de Squid, más concretamente en `/var/log/squid/access.log`:

```
1108489147.817    471 192.168.0.50 TCP_MISS/200 3322 GET http://sarg.
    sourceforge.net/sarg.README.txt hugo.santander DIRECT/66.35.250.209 text
    /plain
```

Este log ya lo vimos anteriormente, pero ahora refleja un nuevo dato, el nombre de usuario que se autenticó. A partir de él podemos saber a qué direcciones han accedido los usuarios autenticados.

Podríamos generar a mano informes sobre el uso del acceso a internet a través de nuestro proxy. Sería necesario crear un script que formatease el fichero de log de Squid y generase un resumen. La herramienta Sarg nos permite obtener esta información.

Sarg es una herramienta que nos permite conocer las páginas que nuestros usuarios han visitado, así como otra información referente a la navegación de los mismos a través de Squid. Se obtiene con un paquete separado que puede descargarse de `http://sarg.sourceforge.net/sarg.php` o instalarse a partir de las utilidades que tienen Guadalinex y Fedora para instalar nuevos paquetes:

```
# apt-get install sarg
```

En el caso de Guadalinex los ficheros de configuración se sitúan en `/etc/squid`. El fichero de configuración más importante es `sarg.conf`.

```
# sarg.conf
#
# TAG:  language
#       Available languages:
#               Bulgarian_windows1251
#               Catalan
#               Czech
#               Dutch
#               English
#               French
#               German
#               Hungarian
#               Indonesian
#               Italian
#               Japanese
#               Latvian
#               Polish
#               Portuguese
#               Romanian
#               Russian_koi8
#               Russian_windows1251
#               Serbian
#               Spanish
#               Turkish
#
language English
# TAG:  access_log file
#       Where is the access.log file
#       sarg -l file
#
#access_log /usr/local/squid/var/logs/access.log
access_log /var/log/squid/access.log
# TAG:  title
#       Especify the title for html page.
#
title "Squid User Access Reports"
# TAG:  font_face
#       Especify the font for html page.
#
font_face Arial
# TAG:  header_color
#       Especify the header color
#
header_color darkblue
# TAG:  header_bgcolor
```

```
#       Especify the header bgcolor
#
header_bgcolor blanchedalmond
# TAG:  font_size
# TAG:  font_size
#       Especify the font size
#
header_font_size -1
# TAG:  background_color
# TAG:  background_color
#       Html page background color
#
background_color white
# TAG:  text_color
#       Html page text color
#
text_color black
# TAG:  text_bgcolor
#       Html page text background color
#
text_bgcolor beige
# TAG:  title_color
#       Html page title color
#
title_color green
# TAG:  logo_image
#       Html page logo.
#
#logo_image none
# TAG:  logo_text
#       Html page logo text.
#
#logo_text ""
# TAG:  logo_text_color
#       Html page logo texti color.
#
#logo_text_color black
# TAG:  logo_image_size
#       Html page logo image size.
#       width height
#
#image_size 80 45
# TAG:  background_image
#       Html page background image
#
#background_image none
# TAG:  password
#       User password file used by authentication
#       If used here, reports will be generated only for that users.
#
#password none
# TAG:  temporary_dir
#       Temporary directory name for work files
#       sarg -w dir
#
temporary_dir /tmp
# TAG:  output_dir
#       The reports will be saved in that directory
```

```
#       sarg -o dir
#
#output_dir /var/www/html/squid-reports
output_dir /var/www/squid-reports
# TAG:  output_email
#       Email address to send the reports. If you use this tag, no html
    reports will be generated.
#       sarg -e email
#
#output_email root@localhost
# TAG:  resolve_ip yes/no
#       Convert ip address to dns name
#       sarg -n
resolve_ip
# TAG:  user_ip yes/no
#       Use Ip Address instead userid in reports.
#       sarg -p
user_ip no
# TAG:  topuser_sort_field field normal/reverse
#       Sort field for the Topuser Report.
#       Allowed fields: USER CONNECT BYTES TIME
#
topuser_sort_field BYTES reverse
# TAG:  user_sort_field field normal/reverse
#       Sort field for the User Report.
#       Allowed fields: SITE CONNECT BYTES TIME
#
user_sort_field BYTES reverse
# TAG:  exclude_users file
#       users within the file will be excluded from reports.
#       you can use indexonly to have only index.html file.
#
exclude_users /etc/squid/sarg.users
# TAG:  exclude_hosts file
#       Hosts, domains or subnets will be excluded from reports.
#
#       Eg.: 192.168.10.10 - exclude ip address only
#            192.168.10.0  - exclude full C class
#            s1.acme.foo   - exclude hostname only
#            acme.foo      - exclude full domain name
#
exclude_hosts /etc/squid/sarg.hosts
# TAG:  useragent_log file
#       Put here where is useragent.log to nable useragent report.
#
#useragent_log none
# TAG:  date_format
#       Date format in reports: e (Europe=dd/mm/yy), u (USA=mm/dd/yy), w (
    Weekly=yy.ww)
date_format u
# TAG:  per_user_limit file MB
#       Save userid on file if download exceed n MB.
#
#       This option can be used to disable user access if user exceed a
    download limit.
#per_user_limit none
# TAG: lastlog n
#      How many reports files must be keept in reports directory.
```

```
#       The oldest report file will be automatically removed.
#       0 - no limit.
#
lastlog 0
# TAG: remove_temp_files yes
#       Remove temporary files: geral, usuarios, top, periodo from root
    report directory.
#
remove_temp_files yes
# TAG: index yes|no|only
#       Generate the main index.html.
#       only - generate only the main index.html
#
index yes
# TAG: overwrite_report yes|no
#       yes - if report date already exist then will be overwrited.
#        no - if report date already exist then will be renamed to filename.n
    , filename.n+1
#
overwrite_report yes
# TAG: records_without_userid ignore|ip|everybody
#       What can I do with records without user id (no authentication) in
    access.log file ?
#
#       ignore - This record will be ignored.
#           ip - Use ip address instead. (default)
#   everybody - Use "everybody" instead.
#
records_without_userid ip
# TAG: use_comma no|yes
#       Use comma instead point in reports.
#       Eg.: use_comma yes => 23,450,110
#            use_comma no  => 23.450.110
#
use_comma yes
# TAG: mail_utility mail|mailx
#       Mail command to use to send reports via SMTP
#
mail_utility mailx
# TAG: topsites_num n
#       How many sites in topsites report.
#
topsites_num 100
# TAG: topsites_sort_order CONNECT|BYTES A|D
#       Sort for topsites report, where A=Ascendent, D=Descendent
#
topsites_sort_order CONNECT D
# TAG: index_sort_order A/D
#       Sort for index.html, where A=Ascendent, D=Descendent
#
index_sort_order D
# TAG: exclude_codes file
#       Ignore records with these codes. Eg.: NONE/400
#
exclude_codes /etc/squid/sarg.exclude_codes
# TAG: replace_index string
#       Replace "index.html" in the main index file with this string
#       If null "index.html" is used
```

```
#
#replace_index <?php echo str_replace(".", "_", $REMOTE_ADDR); echo ".html";
    ?>
# TAG: max_elapsed milliseconds
#      If elapsed time is recorded in log is greater than max_elapsed use 0
    for elapsed time.
#      Use 0 for no checking
#
#max_elapsed 0
# 8 Hours
max_elapsed 28800000
# TAG: report_type type
#      What kind of reports to generate.
#      topsites            - shows the site, connect and bytes
#      sites_users         - shows which users were accessing a site
#      users_sites         - shows sites accessed by the user
#      date_time           - shows the amount of bytes used by day and hour
#      denied              - shows all denied sites with full URL
#      auth_failures       - shows autentication failures
#      site_user_time_date - shows sites, dates, times and bytes
#
#      Eg.: report_type topsites denied
#
report_type topsites sites_users users_sites date_time denied auth_failures
    site_user_time_date
# TAG: usertab filename
#      You can change the "userid" or the "ip address" to be a real user
    name on the rpeorts.
#      Table syntax:
#               userid name   or   ip address name
#      Eg:
#               SirIsaac Isaac Newton
#               vinci Leonardo da Vinci
#               192.168.10.1 Karol Wojtyla
#
#      Each line must be terminated with '\n'
#
usertab /etc/squid/sarg.usertab
# TAG: long_url yes|no
#      If yes, the full url is showed in report.
#      If no, only the site will be showed
#
#      YES option generate very big sort files and reports.
#
long_url no
# TAG: date_time_by bytes|elap
#      Date/Time reports will use bytes or elapsed time?
#
date_time_by bytes
# TAG: charset name
#      ISO 8859 is a full series of 10 standardized multilingual single-byte
    coded (8bit)
#      graphic character sets for writing in alphabetic languages
#      You can use the following charsets:
#               Latin1          - West European
#               Latin2          - East European
#               Latin3          - South European
#               Latin4          - North European
```

```
#                Cyrillic
#                Arabic
#                Greek
#                Hebrew
#                Latin5          - Turkish
#                Latin6
#                Windows-1251
#                Koi8-r
#
charset Latin1
# TAG: user_invalid_char "&/"
#      Records that contain invalid characters in userid will be ignored by
    Sarg.
#
#user_invalid_char "&/"
# TAG: privacy yes|no
#      privacy_string "***.***.***.***"
#      privacy_string_color blue
#      In some countries the sysadm cannot see the visited sites by a
    restrictive law.
#      Using privacy yes the visited url will be changes by privacy_string
    and the link
#      will be removed from reports.
#
#privacy no
#privacy_string "***.***.***.***"
#privacy_string_color blue
# TAG: include_users "user1:user2:...:usern"
#      Reports will be generated only for listed users.
#
#include_users none
# TAG: exclude_string "string1:string2:...:stringn"
#      Records from access.log file that contain one of listed strings will
    be ignored.
#
#exclude_string none
# TAG: show_successful_message yes|no
#      Shows "Successful report generated on dir" at end of process.
#
show_successful_message no
# TAG: show_read_statistics yes|no
#      Shows some reading statistics.
#
show_read_statistics no
# TAG: topuser_fields
#      Which fields must be in Topuser report.
#
topuser_fields NUM DATE_TIME USERID CONNECT BYTES %BYTES IN-CACHE-OUT
    USED_TIME MILISEC %TIME TOTAL AVERAGE
# TAG: user_report_fields
#      Which fields must be in User report.
#
user_report_fields CONNECT BYTES %BYTES IN-CACHE-OUT USED_TIME MILISEC %TIME
     TOTAL AVERAGE
# TAG: topuser_num n
#      How many users in topsites report. 0 = no limit
#
topuser_num 0
```

```
# TAG: site_user_time_date_type list|table
#      generate reports for site_user_time_date in list or table format
#
site_user_time_date_type table
# TAG: datafile file
#      Save the report results in a file to populate some database
#
#datafile none
# TAG: datafile_delimiter ";"
#      ascii character to use as a field separator in datafile
#
#datafile_delimiter ";"
# TAG: datafile_fields all
#      Which data fields must be in datafile
#      user;date;time;url;connect;bytes;in_cache;out_cache;elapsed
#
#datafile_fields user;date;time;url;connect;bytes;in_cache;out_cache;elapsed
# TAG: weekdays
#      The weekdays to take account ( Sunday->0, Saturday->6 )
# Example:
#weekdays 1-3,5
# Default:
#weekdays 0-6
# TAG: hours
#      The hours to take account
# Example:
#hours 7-12,14,16,18-20
# Default:
#hours 0-23
# TAG: squidguard_log_path file
#      Generate reports from SquidGuard logs.
#
#squidguard_log_path none
# TAG: show_sarg_info yes|no
#      shows sarg information and site patch on each report bottom
#
#show_sarg_info yes
# TAG: parsed_output_log directory
#      Saves the processed log in a sarg format after parsing the squid log
    file.
#      This is a way to dump all of the data structures out, after parsing
    from
#      the logs (presumably this data will be much smaller than the log
    files themselves),
#      and pull them back in for later processing and merging with data from
     previous logs.
#
#parsed_output_log none
# TAG parsed_output_log_compress /bin/gzip|/usr/bin/bzip2|nocompress
#      sarg logs compress util
#
#parsed_output_log_compress /bin/gzip
# TAG displayed_values bytes|abbreviation
#      how the values will be displayed in reports.
#      eg. bytes        -  209.526
#          abbreviation -  210K
#
displayed_values bytes
```

No entraremos a describir los parámetros de configuración de Sarg ya que el fichero de configuración generado por defecto es suficiente para comenzar a utilizar la herramienta. Únicamente indicar que algunos de estos parámetros pueden ser definidos también en tiempo de ejecución. De esta manera cuando ejecutemos Sarg podemos modificar el comportamiento definido en `sarg.conf` a partir de parámetros de ejecución. Para más información puede consultarse la página del manual referente a `sarg` (`man sarg`).

La configuración por defecto establece la localización de los ficheros de log de Squid así como el lugar donde queremos almacenar los informes generados.

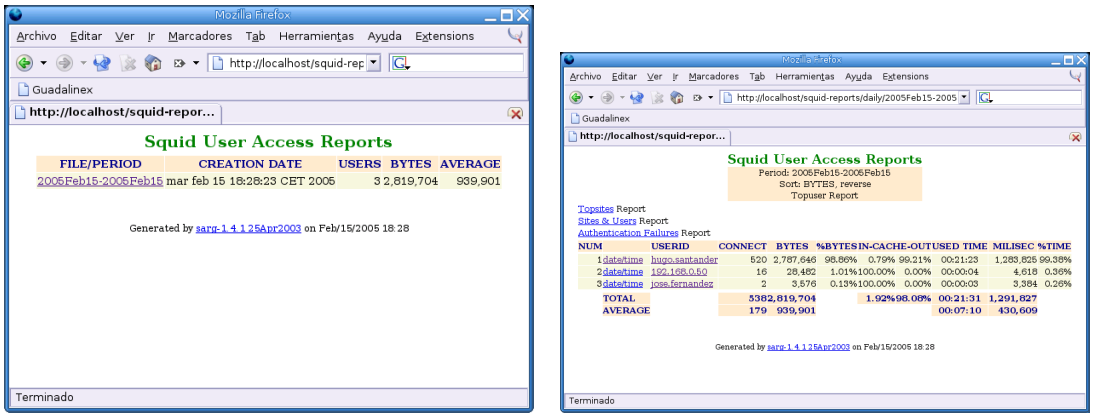
```
# TAG:   access_log file
#        Where is the access.log file
#        sarg -l file
#
access_log /var/log/squid/access.log
....
# TAG:   output_dir
#        The reports will be saved in that directory
#        sarg -o dir
#
output_dir /var/www/squid-reports
```

Estos parámetros de configuración pueden modificarse desde línea de comandos:

```
root@guadalinex:/etc/squid# /usr/bin/sarg -l /var/log/squid/access.log -o /
    var/www/squid-reports/daily
```

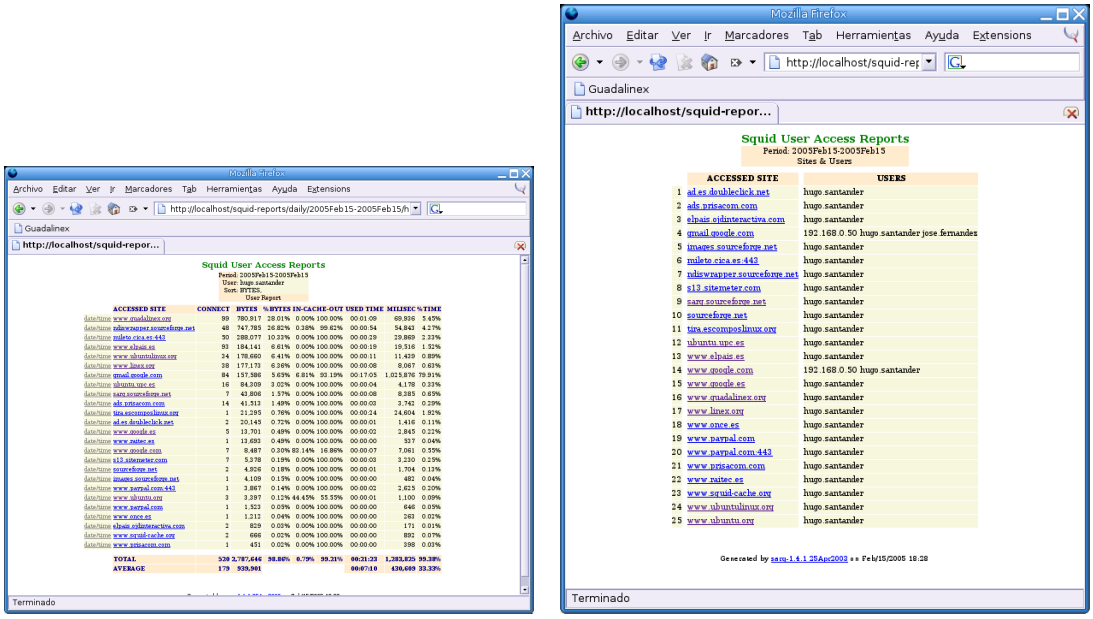
¿Y eso es todo? Pues vamos a ver que informes se han generado:

Figura 16.4: Página principal y listado de informes disponible



Puede obtenerse aún más detalle, mostrando un informe detallado de los sitios que ha accedido cada usuario o un listado de sitios con los usuarios que han accedido a los mismos:

Figura 16.5: Informes por usuario y por sitio visitado



16.6. ➥ Para practicar

16.6.1. Castellanizar los errores de Squid

1.

   a) Instalar y configurar `squid` en el equipo local para que no permita el acceso a más máquinas que la nuestra y que además limite el acceso a todas las páginas en cuya URL aparezcan las cadenas: `apa` y `sex`.

   b) Castellanizar los errores: sólo hay que modifcar el enlace simbólico `/etc/squid/errors` para que apunte a `/usr/share/squid/errors/Spanish`. Después hacer que el demonio relea la configuración.

   ```
   # ll /etc/squid/errors
   lrwxrwxrwx 1 root root 31 nov 9 14:49 errors ->/usr/share/squid/errors/Spanish
   ```

   c) Comprobar que funciona configurando `mozilla` tal cual aparece en los apuntes.

16.6.2. Limitar ancho de banda para determinadas extensiones

Se trata de optar porque nuestros clientes no consuman todo el ancho de banda[12] disponible para bajarse determinados programas. Para eso se usan las *delay pools*. Tendremos que añadir al final (casi) del fichero de configuración de `squid` una serie de líneas. Las directrices que vamos a usar se pueden resumir en:

- Un par de ACL que nos van a permitir:
  1. Definir el ancho de banda para nuestra máquina
  2. Limitar el ancho de banda a partir de la extensión al resto de ordenadores

[12] Para conocer más sobre este tema Limitar el ancho de banda COMO `http://mural.uv.es/~joferna/doc/Limitar-ancho-de-banda-COMO/html/`

Podemos optar por no poner la primera regla y limitar también a nuestro ordenador.

```
acl regla_primera src 127.0.0.1/255.255.255.255
acl regla_segunda url_regex -i .exe .mp3 .zip .avi .mpeg .rar
```

- Definimos el nº de reglas de demora

  ```
  delay_pools 2
  ```

- Podemos definir tres tipos de *delay pools*, pero sólo vamos a trabajar con el segundo tipo, eso lo indicamos con

  ```
  delay_class número tipo
  ```

  Cada tipo permite definirle una serie de parámetros, a las del tipo dos

  ```
  delay_parameters número global máquina
  ```

  donde lo único a comentar es el significado de `global` y `máquina`. Ambos de la forma `caudal_bytes/máximo_bytes` con el significado:

  - `caudal_bytes` es el número de bytes por segundo de tasa de transferencia mantenida una vez que la descarga sobrepasa `máximo_bytes`[13]. Si establecemos los valores `-1/-1` el significado es sin límite.
  - El primer par de números (`global`) establece los valores para toda la red, mientras que con `máquina` lo hacemos para una IP concreta. Por ejemplo con

    ```
    delay_parameters 2 10000/20000 5000/15000
    ```

    establecemos:

    `10000/20000` una vez que los archivos sean mayores de 20000 bytes, las descargas (para toda la red) usan como máximo 10000 bytes

    `5000/15000` para cada IP concreta, si el archivo sobrepasa los 15000 bytes, proseguirá la descarga a 5000 bytes por segundo

  Con `delay_access` establecemos qué *delay pools* gestiona la petición.

- Todo junto quedaría

```
#Listas ACL y número de delay pools
acl regla_primera src 127.0.0.1/255.255.255.255
acl regla_segunda url_regex -i .exe .mp3 .zip .avi .mpeg .rar
delay_pools 2

#Pool 1 de tipo 2
#Permitimos que desde nuestra máquina no haya límite de bajada
delay_class 1 2
delay_parameters 1 -1/-1 -1/-1
delay_access 1 allow regla_primera

#Pool 2 de tipo 2
#Al resto de máquinas le límitamos la bajada de ficheros extensión
#exe .mp3 .zip .avi .mpeg .rar .vqf
#Establecemos un límite para toda la red a 10Kb/s
```

[13] Máximo guardado para toda la red o máquina en particular.

```
#pero cada máquina con un límite de 5Kb/s
delay_class 2 2
delay_parameters 2 10000/15000 5000/15000
delay_access 2 allow regla_segunda
```

16.6.3. Proxy transparente

Si deseamos disponer de un servicio de proxy en el que los clientes no tengan que modificar nada en la configuración del navegador necesitamos montar un proxy transparente (como el de Telefónica). En este caso, los usuarios no necesitan modificar nada en la configuración del navegador de sus equipos.

- Máquina servidor: Para disponer de esta funcionalidad descomentaremos en el fichero de configuración de squid las líneas[14]

  ```
  httpd_accel_host virtual
  httpd_accel_port 80
  httpd_accel_with_proxy on
  httpd_accel_uses_host_header on
  ```

  Además, hay que configurar bien las reglas de filtrado de paquetes (supondremos que el interfaz de red conectado a Internet es `eth1` y el de la red local `eth0`). Un posible script de configuración puede ser

  ```
  echo 1 >/proc/sys/net/ipv4/ip_forward
  iptables --flush
  iptables --table nat --flush
  #Activamos el NAT con enmascaramiento
  iptables --table nat --append POSTROUTING -s 172.26.0.0/24 --out-interface eth1 -j MAS-
  QUERADE
  iptables --append FORWARD -s 172.26.0.0/24 --in-interface eth0 -j ACCEPT
  # Hacer que squid responda llamadas http
  iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j REDIRECT --to-port 3128
  #Política para la red local de todo permitido
  iptables -A INPUT -s 172.26.0.0/24 -j ACCEPT
  ```

- Máquinas clientes: Es necesario poner la puerta de enlace y las IPs de los servidores de nombres (si optamos por usar un servicio de DHCP se puede hacer automáticamente). Pero ya no hay que optar por configurar nada en los navegadores.

16.7. DansGuardian

Si queremos proteger a los menores del contenido peligros que existe en Internet o la política de nuestra organización restringe la visita a determinados sitios, tenemos dos opciones. Una es aplicar reglas en squid para prohibir determinados sitios[15], pero esta opción se convertirá en una tarea draconiana. O sólo permitimos el acceso a unos pocos sitios, o la lista de los sitios prohibidos será siempre una lista inacabada. Una mejor solución es analizar el contenido de las páginas que se sirven y si no cumplen con nuestra norma de lo que es un sitio seguro, no dejarla pasar. Esta es la tarea de un filtro de contenidos, y DansGuardian es uno de ellos.

[14] Funciona pero sin https ni ftp.
[15] Como playboy.com, por ejemplo.

16.7.1. Funcionamiento

DansGuardian funcionará entre el usuario y el servidor proxy-cache (por defecto squid). El usuario realizará las peticiones a dansGuardian que se las envía al proxy si no están filtradas y a las devueltas por el proxy, les analizará el contenido para aplicar de nuevo las restricciones.

Es por esta manera de funcionar, que si queremos un filtrado de contenidos fiable debemos configurar las acl de squid para que sólo permita conexiones desde dansguardian, por ejemplo limitando a la misma máquina, si ambos se ejecutan en ella.

Para realizar la configuración inicial modificaremos lo necesario en `dansguardian.conf` y `dansguardianfx.conf` donde `x` será el grupo de filtrado.

DansGuardian revisa primero las listas de excepciones, después comprobará las listas grises y por último las banned.

Comprueba si el usuario y la máquina están permitidos y el filtro a aplicar. Comprueba el dominio, la url y posteriormente el contenido. A la hora de filtrar el contenido con las palabras clave, podremos cambiar esas expresiones o podremos bloquear la página. Para bloquear la página a la hora de definir las listas de palabras se les añade un peso, de tal forma que si esa cadena de texto aparece, se suma ese peso. Si la palabra pertenece a la lista de excepciones se le quita ese peso de forma que al final del análisis de la página obtendremos un peso total que no debe sobrepasar el límite que se le haya fijado. El límite variará según el tipo de usuario y se establece por grupo de filtrado (por ejemplo para los niños se recomienda un límite de 50).

Siempre tendremos que tener en cuenta el compromiso entre el filtrado de contenidos y el tiempo de proceso para la página.

Esta herramienta permite un filtrado eficiente pero para ello tendremos que probar, ajustar y personalizar la configuración hasta obtener los resultados requeridos.

16.7.2. Instalación

Para la instalación de dansguardian, según la distribución que estemos utilizando se realizará de forma inmediata.

Mediante :

```
#apt-get install dansguardian
```

o bien

```
# rpm -Uhv dansguardian-version-sistema.rpm
```

A la hora de instalar dansguardian, debemos tener en cuenta que para un funcionamiento correcto tendremos que tener instalado squid como proxy-cache.

Para iniciar dansguardian lo haremos como otro servicio más mediante el script

```
#/etc/init.d/dansguardian [start/stop]
```

Por defecto se inicia la ejecución en el puerto 8080 y con el puerto 3128 de squid.

16.8. Configuración

En el directorio de configuración `/etc/dansguardian` nos encontramos los diferentes ficheros que permiten afinar la configuración inicial.

En el fichero `dansguardian.conf` encontramos los siguiente parámetros más importantes:

`reportinglevel=3` $\longmapsto$ el valor por defecto es el más completo

`languagedir = '/etc/dansguardian/languages'` $\longmapsto$ directorio donde se situan las diferentes plantillas para los lenguajes

`language = 'mxspanish'` ⟼ nombre del fichero de idioma

`logfileformat = 3` ⟼ nos puede interesar que esté en formato de squid para unificar las estadísticas

`filterip =` ⟼ dirección ip en la que escucha

`filterport = 8080` ⟼ puerto en el que escucha

`proxyip = 127.0.0.1` ⟼ dirección de la máquina donde esté squid

`proxyport = 3128` ⟼ puerto en el que escucha el proxy

`urlcachenumber = 1000` y `urlcacheage = 900` ⟼ para caché de páginas permitidas

`preservecase = 0` ⟼ para no distinguir mayúsculas y minúsculas, todo se pasa a minúsculas antes de analizarlo

El resto de parámetros hacen referencia a datos del proceso y al resto de ficheros de configuración.

En el fichero `dansguardianf1.conf` se definen el resto de ficheros, el peso de la página a filtrar y la posibilidad de definir una palabra de paso para desabilitar el filtrado temporalmente.

A continuación, describimos el resto de ficheros que nos van a permitir realizar el filtrado y control de contenidos. Tendremos que ir definiendo y afinando los diferentes tipos de filtrado teniendo en cuenta el retraso que puede introducirse en la navegación web del usuario. Es por esto, que tendremos que llegar a un compromiso del tipo de filtrado y la rapidez del mismo.

bannedextensionlist ⟼ extensiones de ficheros no deseados

bannediplist ⟼ ip de los clientes para denegarles el acceso

bannedmimetypelist ⟼ tipos de contenidos no permitidos

bannedphraselist ⟼ palabras o frases para filtrar o ficheros donde se hayan definido previamente dichas palabras mediante la directiva .Include, podemos encontrar ejemplos en `/etc/dansguardian/phraselists`

bannedregexpurllist ⟼ expresiones que queremos filtrar cuando aparecen en la url

bannedsitelist ⟼ podemos incluir los dominios para filtrar, usar la directiva .Include, filtrar todo y permitir sólo las excepciones o forzar a que se filtre por ip

bannedurllist ⟼ bloquean una parte de un dominio mediante la definición de su url

banneduserlist ⟼ usuarios de la autenticación del proxy a los que se quiere bloquear el acceso

contentregexplist ⟼ mediante el formato "badword"->"expresion" podemos sustituir determinadas expresiones por otras

filtergroupslist ⟼ permiten asociar a usuarios grupos de filtros

greysitelist ⟼similares a las "banned" pero las sobreescriben

greyurllist ⟼ similares a las "banned" pero las sobrescriben

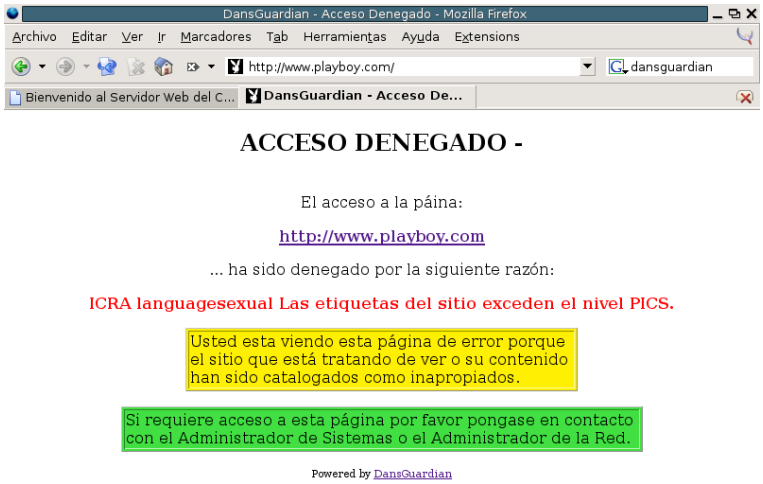
exceptioniplist, exceptionphraselist, exceptionsitelist, exceptionurllist, exceptionuserlist ⟼ En los ficheros "exception" usando la misma forma de definición y la directiva .Include podremos definir los elementos que queremos asegurarnos que no sean filtrados.

weightedphraselist ⟼ instrucciones para definir la lista de filtrado <palabra><peso>

A partir de aquí debemos comenzar a utilizar las múltiples opciones de filtrado hasta obtener un resultado óptimo.

Para facilitar la gestión, existe un módulo para versiones superiores de 2.4.x para la herramienta webmin.

Si configuramos nuestro navegador para que acceda al filtro de contenido (por ejemplo en la máquina guadalinex.midominio.org, en el puerto 8080), e intentamos acceder a una página no recomendada a menores, obtenemos lo siguiente:



Existe también un conjunto de herramientas de testeo y ampliación (antivirus, x-forwarded,...) Para profundizar más, podéis visitar `http://dansguardian.org`

Prácticas

TIPO I

E2-I-1

Realizar correctamente todas las preguntas del cuestionario que para esta entrega se ha habilitado en Moodle.

1. Queremos autorizar el acceso a internet exclusivamente a la red 172.26.0.x prohibiendo el acceso a las direcciones del dominio chicasychicosdesnudos.com. ¿Qué claúsulas de configuración de Squid serán necesarias?:

   a) acl adultos dstdomain `www.chicasychicosdesnudos.com`
   acl hostpermitidos src 172.26.0.0/255.255.255.0
   acl all src 0.0.0.0/0.0.0.0
   http_access deny adultos
   http_access allow hostpermitidos
   http_access deny all

   b) acl adultos dstdomain `www.chicasychicosdesnudos.com`
   acl hostpermitidos src 172.26.0.0/255.255.255.0
   acl all src 0.0.0.0/0.0.0.0
   http_access deny all

   c) acl adultos dstdomain `chicasychicosdesnudos.com`
   acl hostpermitidos src 172.26.0.0/255.255.255.0
   acl all src 0.0.0.0/0.0.0.0
   http_access allow hostpermitidos
   http_access deny all

   d) acl adultos dstdomain `chicasychicosdesnudos.com`
   acl hostpermitidos src 172.26.0.0/255.255.255.0
   acl all src 0.0.0.0/0.0.0.0
   http_access deny adultos
   http_access allow hostpermitidos
   http_access deny all

2. En caso de olvidar la frase de paso con la que ciframos la clave privada en OpenSSH, podemos cambiarla a una nueva mediante el uso de la siguiente línea de comando:

   a) `ssh-keygen -t dsa -p`

   b) `ssh-changepass`

   c) `ssh-keygen -t dsa -N "nueva frase de paso"`

d) `ssh-keygen -t dsa -p "nueva frase de paso"`

3. Si utilizamos el siguiente comando para añadir entradas a nuestro ldap sin soporte de tls ni sasl:
   `ldapadd -D 'cn=admin,dc=thales,dc=org' -w secret -f fichero.ldif`
   ¿Qué sucederá?:

   a) Se incluirán todas las entradas contenidas en el fichero.ldif
   b) Se pruducirá un error al no poder autenticarnos
   c) Se producirá un error por usar -w en lugar de -W
   d) Se incluirán sólo las entradas correctas

4. En los sistemas de criptografía asimétrica o de clave pública, podemos afirmar que:

   a) Lo cifrado con una clave sólo se puede descifrar con esa misma clave
   b) Existen dos claves, una pública y otra privada
   c) La clave pública debe permanecer oculta
   d) El algoritmo más utilizado es el 3DES

5. El contenido del fichero `/etc/hosts.allow` es:
   `sshd: *.cica.es`
   y de `/etc/hosts.deny` es:
   `smtp: ALL`
   Podemos afirmar que:

   a) Podemos dormir tranquilos porque nuestra máquina está protegida frente a los crackers
   b) La recogida de correo electrónico desde los clientes mediante POP está restringida
   c) Sólo podremos acceder por ssh desde el dominio cica.es
   d) Podremos acceder por ssh desde cualquier máquina

6. Tenemos la siguiente línea en el fichero `/etc/inetd.conf`
   `telnet stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.telnetd`
   ¿Qué respuesta es cierta?

   a) El acceso al servicio telnet está controlado por el sistema tcp-wrappers
   b) El servicio telnet corresponde al puerto 25
   c) El usuario root podrá conectarse siempre al servicio telnet, que para eso es el superusuario
   d) Es un servicio tcp, no orientado a conexión

7. La configuración de la utilidad de análisis de ficheros de logs de Squid SARG puede realizarse:

   a) Únicamente en el fichero de configuración sarg.conf
   b) Únicamente en el fichero de configuración squid.conf
   c) A partir del fichero de configuración sarg.conf o desde la línea de comandos indicando los parámetros que se modifican.
   d) A partir del fichero de configuración sarg.conf y de squid.conf

8. Para utilizar autenticación mediante ldap, la contraseña del usuario:

   a) La contraseña del usuario irá cifrada, anteponiendo {algoritmo} o sin cifrar, en el atributo userPassword de la clase posixAccount

   b) La contraseña del usuario debe estar en el fichero `/etc/passwd` del sistema operativo local

   c) La contraseña del usuario debe estar en el fichero `/etc/password` del servidor ldap

   d) La contraseña debe almacenarse en texto claro

9. Deseamos dar de alta al usuario mileto de la máquina linux para que pueda acceder a su $HOME de usuario desde la red Windows usando SAMBA. ¿Qué comando hemos de usar?

   a) `#smbpasswd mileto`

   b) `#smbpasswd -a mileto`

   c) `#smbadduser -u mileto`

   d) `#smbadduser mileto`

   e) Ninguna de las anteriores

10. Hemos añadido una sección personalizada al fichero de configuración de SAMBA de nombre [trabajos]. Queremos limitar el acceso a ese recurso a los miembros del grupo "dpto" de la máquina linux, ¿qué directiva tenemos que añadir?

    a) `valid users = @dpto`

    b) `valid users = dpto`

    c) `valid group = dpto`

    d) `valid group = @dpto`

    e) Ninguna de las anteriores

E2-I-2 DNS

El sistema de resolución de nombres se encarga de mantener servicios esenciales en Internet. Averiguar lo siguiente:

- Para el servicio `www.juntadeandalucia.es`, obtener su dirección IP y el nombre al que le corresponde el registro de tipo A para esa dirección IP.
- Nombres y direcciones IP de los intercambiadores de correo y servidores de nombres para el dominio juntadeandalucia.es

El resultado de la ejecución de los comandos y las explicaciones deben estar en un fichero de nombre `E2-I-2.txt`

Tipo II

E2-II-1 Cups:

Tenemos un servidor de impresión de IP 192.168.0.1 al que se conectan las impresoras de nombres: *color* y *laser*. El servidor sale a internet con una IP pública (80.32.123.200), las máquinas de la Red Local son todas Guadalinex, y deseamos que el servicio de impresión cumpla una serie de condiciones. Esta práctica consiste en generar un fichero o ficheros de configuración de cups que permitan:

- Imprimir desde la red local a cualquier máquina y sin autentificación sobre la impresora *color*.

  - Imprimir en la impresora *laser* sólo desde la dirección local (127.0.0.1) y la IP remota 80.32.134.123 a cualquier usuario del sistema.
  - Que, además, nos permita gestionar todo el servidor de impresión desde la IP anterior (80.32.134.123)
  - No permita imprimir al usuario: "muchacara" en la impresora *laser*.

    NOTA: Para esta última cuestión véase[16] el apartado b) del ejemplo del **Para Practicar** en la página ?? y téngase en cuenta que, en este caso, la directiva a usar es `DenyUser`

    El resultado de la práctica debéis mandarlo en un fichero de nombre `E2-II-1.txt`

E2-II-2 DansGuardian:

Instalar el sistema de control de contenidos DansGuardian. Configurar un navegador para que se conecte mediante él a Internet. Intentar un acceso a la dirección `www.telecinco.es`, seleccionar Crónicas marcianas en el panel de la izquierda (o directamente ir a `www.cronicasmarcianas.telecinco.es`).

Intentar también acceder a `www.interviu.es`. El resultado de las pantallas que presenta el navegador y las explicaciones deben mandarse en formato OpenOffice en un fichero de nombre `E2-II-2.sxw`

[16] Y en general para toda la práctica

Bibliografía

[1] Squid: The Definitive Guide. Duane Wessels. Editorial O'Reilly

[2] Sarg: Squid Analysis Report Generator. http://sarg.sourceforge.net/

[3] Guía de Administración de Redes con Linux. Olaf Kirch y Terry Dawson. Proyecto LuCAS, traducción al español. `http://es.tldp.org/Manuales-LuCAS/GARL2/garl2`

[4] Introduction to Linux A Hands on Guide. Machtelt Garrels. `http://tille.soti.org/training/tldp`

[5] TCP/IP Network Administration, 3rd Edition. Craig Hunt

[6] SSH, The Secure Shell: The Definitive Guide. by Daniel J. Barrett and Richard E. Silverman

[7] Managing NFS and NIS, 2nd Edition By Hal Stern, Mike Eisler and Ricardo Labiaga

[8] DNS and Bind, 4th Edition By Paul Albitz and Cricket Liu

[9] Usando SAMBA, Robert Eckstein, David Coliier-Brown, Peter Kelly

Parte III

Servidor Web y Correo electrónico

Capítulo 17

Servidor Web Apache

> Apache es un producto fantástico. Hace todo lo que se quiere que haga, y nada de lo que no se quiere. Es rápido, fiable y barato. ¿Qué más se podría pedir de una unidad de software?
>
> Apache puede ser todo esto porque es *open source*. (*Servidor Apache*, Rick Bowen & ken Coar)

17.1. Servidor Web (apache).

El servidor Apache es un servidor web HTTP de software libre que funciona en diversos sistemas operativos (Unix/Linux, Windows, MacOS, etc.). El objetivo del proyecto es proporcionar un servidor HTTP seguro, eficiente, extensible y que cumpla con los estándares[1].

Apache es el servidor web más usado en Internet desde Abril de 1996. Las últimas estadísticas, de febrero de 2005 proporcionadas por Netcraft Web Server Survey, muestran que más del 68 % de los sitios web de Internet utilizan Apache. Apache es un proyecto de la Apache Software Foundation. (www.apache.org).

Veamos algunas pinceladas de su historia. Poco después del nacimiento de la Web en el CERN, un grupo de personas del Centro Nacional de Actividades de Supercomputación (*National Center for Supercomputing Activities*, NCSA), de la Universidad de Illinois, creó un servidor web (HTTPd NCSA) que fue el más usado en la web hasta mediados de 1994.

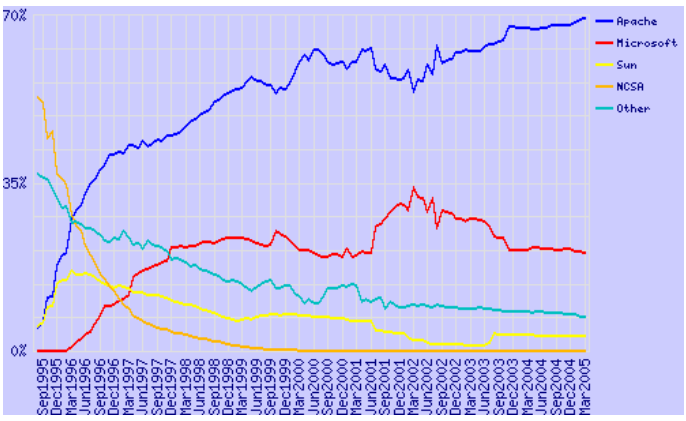


Figura 17.1: Estadísticas uso Apache

Su principal desarrollador (Rob McCool) abandonó poco después el NCSA y el proyecto. Sin embargo, bastantes personas siguieron trabajando con el servidor HTTPd NCSA y así fueron surgiendo diversos parches para el código fuente.

Fue entonces cuando un grupo de desarrolladores (ocho personas en principio) comenzaron a trabajar sobre HTTPd y los parches que habían ido mejorándolo: surgía el proyecto Apache (`http://www.apache.org/`).

[1] Y parece que hasta ahora lo ha conseguido.

[2] La última versión estable es la 2.0.53. Para saber las nuevas funcionalidades de Apache 2.0 `http://httpd.apache.org/docs-2.0/es/new_features_2_0.html`. Todavía siguen utilizándose las versiones 1.3 (1.3.33 es la última) sobre todo por compatibilidad con módulos de terceros, aunque se recomiendan las versiones 2.0 porque incorporan muchas nuevas funcionalidades.

La primera versión oficial, Apache 0.6.2, se lanzó en abril de 1995 (el nombre proviene de "A PAtCHy" release, ya que en principio era una versión parcheada del HTTPd 1.3 NCSA). El 1 de diciembre de 1995 se hizo pública la versión 1.0[2]

En 1998 se llegó a un acuerdo con IBM que permitió conseguir que Apache funcionara también bajo Windows, convirtiéndose en una excelente alternativa a IIS (*Microsoft Internet Information Server*).

Apache es el servidor Web (protocolo HTTP) más utilizado en el mundo actualmente. Se encuentra muy por encima de sus competidores, ya sean gratuitos o comerciales[3]. Por supuesto, es el más utilizado en sistemas Linux.

En su forma más simple, un servidor web transmite páginas en formato HTML a los navegadores cliente (Firefox, Netscape, Internet Explorer, Opera, Lynx...) utilizando el protocolo HTTP. Pero un servidor web hoy día puede hacer mucho más, ya sea por sus propios medios o mediante su integración con otros programas o módulos. Prácticamente todos los programas y aplicaciones informáticas tienden a que su forma de presentación para el cliente sea en formato web.

Existen varias formas en las que Apache puede proveernos contenidos:

- **Páginas estáticas** Es el modo básico y más primitivo, pero que en un gran número de casos es lo único que se necesita: transferir ficheros HTML, imágenes... Puede que con un servidor Linux de bajas prestaciones (incluso un 486) consigamos estupendos resultados, si es sólo esto lo que necesitamos.

- **Contenido dinámico** La información cambia constantemente, y un medio para mantener nuestras páginas actualizadas, es generarlas dinámicamente desde una base de datos, ficheros u otras fuentes de datos.

Apache posee muchas facilidades para generar este tipo de contenido.

1. **Soporte del protocolo HTTP 1.1**. Además mantiene la compatibilidad con el HTPP 1.0.

2. **Scripts CGI** y **FastCGI**. CGI viene de *common gateway interface*. Los scripts CGI son programas externos que se llaman desde el propio servidor cuando una página lo requiere. El CGI recibe información del servidor web y genera como salida una página web dinámica para el cliente. El script puede realizarse en cualquier lenguaje de programación siempre que siga las reglas del interfaz CGI. El problema es que es un proceso lento, al tenerse que lanzar un proceso externo al servidor web por cada petición. Perl es uno de los lenguajes más utilizados para ello, aunque también se utilizan scripts de una shell Unix/Linux.

3. **Host virtuales**. Permite atender varios sitios Web en dominios distintos, desde la misma máquina.

4. **Autenticación HTTP**. Permite restringir recursos a determinados usuarios o grupos (distintos de los del sistema).

5. **Intérpretes incluidos en Apache.** Tienen la ventaja sobre los cgi de que están incluidos en el propio Apache y no hay que lanzar un nuevo proceso por cada petición. Los módulos más utilizados son PHP y `mod_perl`.

6. Soporte de **SSI**[4] (*Server Side Includes*) y de **SSL**[5] (*Secure Sockets Layer*)

[3]En Marzo de 2005 casi el 70 % de los servidores Web usan Apache, para saber exactamente los datos en la actualidad se puede consultar
`http://news.netcraft.com/archives/web_server_survey.html`

[4]Directivas que permiten añadir funcionalidad añadida al HTML interpretándolas antes de mandar la página al navegador.

[5]Proporciona cifrado de datos para asegurar la privacidad y fiabilidad de la comunicación Web. Se utiliza criptografía asimétrica y certificados digitales para intercambiar una clave de sesión simétrica.

7. **Servlets y JSP en Java**. Es una opción que se utiliza en los servidores de aplicaciones, por ejemplo Tomcat, JBoss, Oracle IAS, WebSphere de IBM o BEA Weblogic. Su gran ventaja sería la portabilidad y escalabilidad. Desarrollamos en Java y podemos ejecutarlo en cualquier máquina virtual compatible. Un modelo muy utilizado en la actualidad es el de las arquitecturas en capas. Una arquitectura en tres capas utilizaría un cliente web para la capa de presentación, un servidor de aplicaciones que proporciona la lógica de negocio, es decir, el cómo se ejecutan los procesos dentro de la organización, y un servidor de bases de datos.

   La figura que vemos a continuación nos presenta un servidor web tradicional (en la parte derecha) y un modelo de arquitectura en tres capas (a la izquierda).

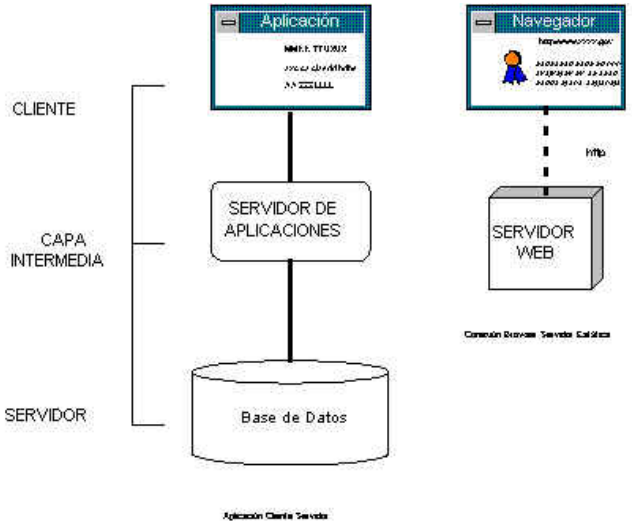


Figura 17.2: Arquitectura en capas

17.2. Instalación

Montar un servidor Web en nuestro Linux es muy sencillo. El servidor web que viene en la distribución es el Apache, funciona como servidor independiente (no lo activa `xinetd`) y escucha por defecto en el puerto 80. Nos aseguramos de que tengamos el paquete apache instalado en nuestra máquina.

17.2.1. Red Hat/Fedora

Instalemos los paquetes[6]

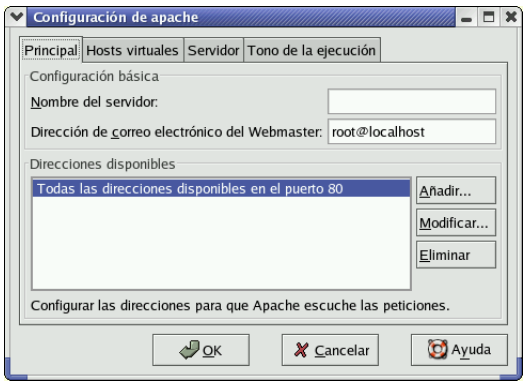
```
## apt-get install httpd httpd-manual httpd-devel system-config-httpd
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
Se instalarán los siguientes paquetes extras:
  apr apr-devel apr-util apr-util-devel httpd-suexec pcre-devel
Se instalarán los paquetes NUEVOS siguientes:
  apr apr-devel apr-util apr-util-devel httpd httpd-devel httpd-
manual httpd-suexec pcre-devel
  system-config-httpd
0 upgraded, 10 newly installed, 0 removed and 166 not upgraded.
```

[6]La lista de dependencias no tiene por qué coincidir con la de vuestros sistemas.

```
Need to get 4163kB of archives.
After unpacking 18,3MB of additional disk space will be used.
¿Quiere continuar? [S/n]
```

Estos serían el paquete básico de apache, el manual y el paquete de desarrollo (sólo el primero es imprescindible).

Hay un cuarto paquete que nos puede facilitar enormemente la configuración de Apache. Se trata de `system-config-httpd`, es una herramienta gráfica para la configuración de Apache.



Si se desea usar, se puede consultar el *Manual de personalización de Red Hat Linux* [12]. Una nota a tener en cuenta y que aparece en en él:

"Aviso

No modifique "a mano" el fichero de configuración de Apache `/etc/httpd/conf/httpd.conf` si desea utilizar esta herramienta. Dicha herramienta crea este fichero después de que haya grabado los cambios y haya salido del programa. Si desea añadir módulos u opciones que no se encuentren en la herramienta no podrá usarla."

En el momento del arranque, si existe el enlace `/etc/rc.d/rc3.d/S85httpd,` se arrancará de forma automática. Si no existe, podemos crearlo con cualquiera de las herramientas: `serviceconf`, `ntsysv` o `chkconfig`.

Si queremos activarlo de forma manual alguna vez, podemos ejecutar

```
#/etc/rc.d/init.d/httpd start
```

o

```
#service httpd start
```

y si lo queremos parar

```
#/etc/rc.d/init.d/httpd stop
```

17.2.2. Guadalinex (Debian)

En este caso, lo único que cambia es el nombre de los paquetes[7]:

[7] Es mejor asegurarnos primero de que nuestra lista de paquetes está actualizada, ejecutando `#apt-get update`

```
# apt-get install apache2-mpm-prefork apache2-doc apache2-dev
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Nota, seleccionando apache2-threaded-dev en lugar de apache2-dev
Se instalarán los siguientes paquetes extras:
  apache2-common apache2-threaded-dev autoconf autotools-
dev libapr0 libapr0-dev libdb4.2-dev libexpat1-dev
  libldap2-dev libpcre3-dev libssl-dev libtool ssl-cert zlib1g-dev
Paquetes sugeridos:
  autoconf2.13 autobook autoconf-archive gnu-standards db4.2-
doc libtool-doc g77 fortran77-compiler gcj
Paquetes recomendados
  automaken libltdl3-dev
Se instalarán los siguientes paquetes NUEVOS:
  apache2-common apache2-doc apache2-mpm-prefork apache2-threaded-
dev autoconf autotools-dev libapr0 libapr0-dev
  libdb4.2-dev libexpat1-dev libldap2-dev libpcre3-dev libssl-
dev libtool ssl-cert zlib1g-dev
0 actualizados, 16 se instalarán, 0 para eliminar y 119 no actualiza-
dos.
Necesito descargar 10,3MB de archivos.
Se utilizarán 30,5MB de espacio de disco adicional después de desempa-
quetar.
¿Desea continuar? [S/n] n
```

De nuevo, el único paquete fundamental es el primero[8]. El segundo (`apache2-doc`) es la documentación que acompaña al programa, y el tercero el paquete de desarrollo.

En general, el demonio se iniciará de forma automática y se crearán las entradas adecuadas en los script de arranque. Si esto no es así, ejecutaremos:

```
# /etc/init.d/apache2 start
```

para ponerlo en marcha, y

```
# update-rc.d apache2 defaults
```

para añadir las entradas adecuadas para que el servidor web se inicie en los niveles de ejecución estándar de Debian.

¿Qué es eso de `prefork`?

Apache 2 basa su arquitectura en módulos multiproceso (*Multi-Processing Modules*). Si desde Debian ejecutamos

```
# apt-get install apache2
```

se instala el paquete `apache2-mpm-worker`. En la distribución Sarge de Debian tenemos los paquetes

```
apache2-mpm-worker
apache2-mpm-threadpool
apache2-mpm-prefork
apache2-mpm-perchild
```

Con ellos podemos cambiar la forma en que el servidor Web inicia los procesos y las solicitudes (basándose en hijos o hilos). El significado de cada uno se estos paquetes es[9]:

[8] Debian suele instalar la versión 1.3.x de Apache por defecto con algunos programas. Puede convivir con Apache 2, porque se instalan en directorios diferentes, aunque debemos tener cuidado de arrancar solamente uno de ellos, o si no, el primero que arranque se apoderará del puerto 80 y será el "vencedor".

[9] Si se desea saber más sobre las diferencias existentes entre ellos se pueden consultar:

worker un híbrido multihilos y multiprocesos de servidor Web.

threadpool cada proceso hijo puede tener varios hilos.

prefork servidor sin hilos. Es el más fácil de enlazar con php

perchild cada proceso hijo puede tener varios hilos, además, permite que los procesos demonio puedan ser asignados a usuarios dierentes.

Hemos optado por el módulo *prefork* (servidor sin hilos, donde para cada solicitud al servidor Web es necesario que se inicie un proceso hijo que la atienda) por coherencia con el que se instala para Fedora: con este módulo Apache 2 se comporta de igual forma que la versión 1.3 de Apache, aceptando las mismas directivas.

Podemos saber qué MPM estamos usando con[10]

```
# apache2 -V
Server version: Apache/2.0.51
Server built:   Sep 18 2004 17:21:03
Server's Module Magic Number: 20020903:9
Architecture:   32-bit
Server compiled with....
 -D APACHE_MPM_DIR="server/mpm/prefork"
 -D APR_HAS_SENDFILE
 -D APR_HAS_MMAP
 -D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
 -D APR_USE_SYSVSEM_SERIALIZE
 -D APR_USE_PTHREAD_SERIALIZE
 -D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
 -D APR_HAS_OTHER_CHILD
 -D AP_HAVE_RELIABLE_PIPED_LOGS
 -D HTTPD_ROOT=""
 -D SUEXEC_BIN="/usr/lib/apache2/suexec2"
 -D DEFAULT_PIDLOG="/var/run/httpd.pid"
 -D DEFAULT_SCOREBOARD="logs/apache_runtime_status"
 -D DEFAULT_LOCKFILE="/var/run/accept.lock"
 -D DEFAULT_ERRORLOG="logs/error_log"
 -D AP_TYPES_CONFIG_FILE="/etc/apache2/mime.types"
 -D SERVER_CONFIG_FILE="/etc/apache2/apache2.conf"
```

17.2.3. ¡A navegar!

Una vez instalado el servidor, disponemos de un script en ambas distribuciones que nos permite controlar su estado, se trata de

`apache2ctl` en Debian

`apachectl` en Fedora

- Las páginas 45-46 de *Servidor Apache 2* [6]
- `http://httpd.apache.org/docs-2.0/es/mpm.html`

[10] Para Fedora

```
httpd -V
```

Como podemos observar la salida nos informa de bastantes más aspectos de la configuración del servidor Web

En el resto del tema será el que usemos tanto para Fedora como para Debian[11]. Admite los argumentos en línea de comandos:

`start` para arrancar el servidor. Si ya está en marcha, nos avisará de ello.

`graceful` con este parámetro le indicamos al servidor que relea los ficheros de configuración sin cerrar las conexionas activas. Las conexiones nuevas se iniciarán con la nueva configuración.

`restart` reinicia el servidor (en su caso con la nueva configuración), pero a diferencia del anterior, cierra las conexionas activas.

`stop` cierra el servidor y, por tanto, las conexiones activas.

➥ **Para practicar** Aunque los anteriores son los más usuales, también podemos usar: `fullstatus`, `status` y `configtest`. Comprobar qué funcionalidad tienen. ■

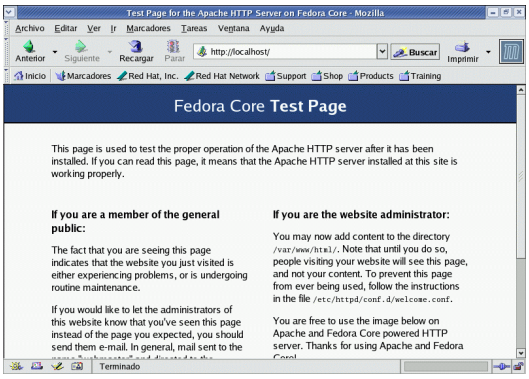
Pongamos en marcha nuestro servidor Web con

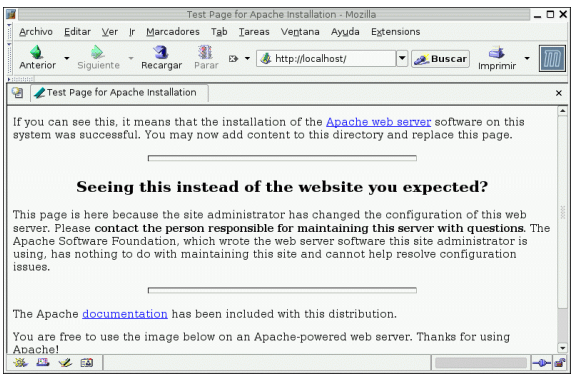
```
#apache2ctl start
```

y, para comprobar que funciona, podemos apuntar nuestro navegador preferido a la dirección

`http://172.26.0.2`[12]

Si conseguimos una pantalla de bienvenida del servidor apache (*It worked!*), ya hemos contactado con nuestro servidor.





Si hemos instalado el paquete del manual podremos acceder a la extensa documentación[13](en castellano) que acompaña al programa desde esta misma página:

[11] Dependiendo de la distribución que tengamos en ese momento como referencia, usaremos la versión adecuada. Tendréis que adecuar el comando a la distribución con que trabajéis en vuestro ordenador.

[12] Si es ésta nuestra dirección, o con `http://localhost` y nos serviría incluso si no tenemos tarjeta de red ni configuración de red.

[13] A nuestra disposición también en `http://httpd.apache.org/docs-2.0/es/`

17.3. Configuración

17.3.1. Document root

El siguiente paso es poner nuestras propias páginas en el servidor, en vez de las de bienvenida de apache. Sin más que ponerlas en el directorio

Fedora: `/var/www/html`[14]

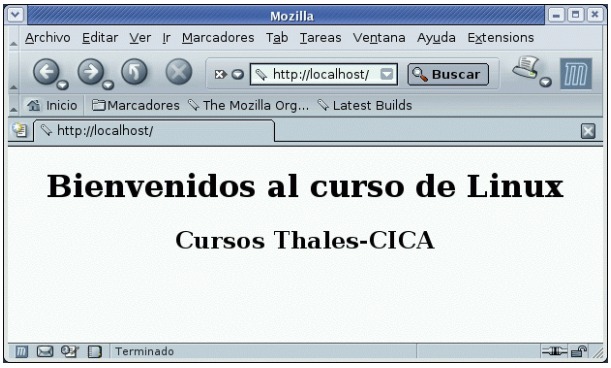
Debian: `/var/www/apache2-default/`

y empezando con la página `index.html`, podremos ver nuestras propias páginas.

➥ **Para practicar:** Comprobar que si ponemos el fichero `index.html`

```
$cat index.html
<center>
  <h1>Bienvenidos al curso de Linux</h1>
  <h2>Cursos Thales-CICA</a>
</center>
```

en la ruta adecuada se obtiene:



[14] En versiones antiguas estaban en `/home/httpd/html`

17.3.2. Ficheros de configuración

Fedora: `/etc/httpd`

Los ficheros de configuración se encuentran en el directorio `/etc/httpd`. Si no es porque realmente lo necesitemos, con la configuración que viene por defecto podemos trabajar de forma satisfactoria.

- El fichero de configuración se llama `httpd.conf`[15] y se encuentra en `/etc/httpd/conf`
- `/etc/httpd/conf.d/` en este directorio se guardan los archivos de configuración para módulos individuales como `ssl.conf`, `perl.conf, php.conf`, etc[16]. Se incluyen en el fichero de configuración a través de la directiva

  ```
  Include conf.d/*.conf
  ```

  que aparece en `/etc/httpd/conf/httpd.con` y sus nombres han de terminar en `.conf`

Debian: `/etc/apache2/apache2.conf`

En Debian el archivo de configuración principal es `/etc/apache2/apache2.conf` y aunque en ese mismo directorio existe `httpd.conf`, está vacío[17].

En el subdirectorio `/etc/apache2` se encuentran además los ficheros:

`magic` lo normal es que no tengamos que modificar nunca este fichero. En él se almacenan los datos "mágicos" del módulo `mod_mime_magic` (para determinar el tipo MIME del fichero mirando unos pocos bytes del contenido)

`ports.conf` directivas de configuración para los puertos y direcciones IP a la escucha.

y los directorios:

`conf.d/` los archivos de este directorio se incluyen en el fichero de configuración a través de la directiva

```
Include /etc/apache2/conf.d/[^.#]*
```

`mods-avalaible/` archivos `.load` (contienen las directivas de Apache para cargar un módulo) y `.conf` (necesarios para configurar ese módulo)

`mods-enabled/` si deseamos activar un módulo hay que crear un enlace simbólico en este directorio para los archivos `.load` (y `.conf` si existen) asociados con el módulo del directorio `mods-available`. Se incluyen en el fichero de configuración por medio de la directivas:

```
Include /etc/apache2/mods-enabled/*.load
Include /etc/apache2/mods-enabled/*.conf
```

Hay dos herramientas que permiten gestionar estos enlaces de forma sencilla

`a2enmod modulo` para crear los enlaces simbólicos asociados al módulo.

`a2dismod modulo` para eliminar los enlaces simbólicos para ese módulo.

[15] Antes, Apache utilizaba además `srm.conf` y `access.conf`, pero en las versiones actuales, el contenido de ambos se ha incluido en `httpd.conf`

[16] También está el fichero de configuración (`welcome.conf`) de la página de bienvenida para el caso de que no exista un fichero `index.html` en el raíz de apache.

[17] Se incluye en el fichero de configuración con:

```
Include /etc/apache2/httpd.conf
```

↬ Veamos un ejemplo sobre su uso. Supongamos que no deseamos que nuestros usuarios puedan alojar su Web en sus `$HOME` de usuario[18] y que sean acesibles mediante `http://www.misitio.com/~usuario`. Para eso necesitamos ver que no esté activo el módulo `mod_userdir` de Apache. Listemos el contenido de:

```
# ls /etc/apache2/mods-enabled/
cgi.load userdir.conf userdir.load
```

Nos encontramos con que está activo, para eliminar los enlaces simbólicos ejecutamos

```
root@eco:~# a2dismod userdir
Module userdir disabled; run /etc/init.d/apache2 force-
reload to fully disable.
root@eco:~# /etc/init.d/apache2 force-reload
Forcing reload of web server: Apache2.
root@eco:~# ls /etc/apache2/mods-enabled/
cgi.load
```

Y, tras reiniciar el servidor, comprobamos que todo está bien (se ha eliminado el enlace). Si ahora deseamos volver a la situación original sólo hemos de escribir

```
# a2enmod userdir
Module userdir installed; run /etc/init.d/apache2 force-
reload to enable.
root@eco:~# /etc/init.d/apache2 force-reload
Forcing reload of web server: Apache2.
root@eco:~# ls /etc/apache2/mods-enabled/
cgi.load  userdir.conf  userdir.load
```

Y todo queda como al principio.

`sites-available/` como `mods-available`, excepto que contiene ficheros de configuración para hosts virtuales diferentes que podrían usarse con apache (el *hostname* no tiene que corresponder exactamente con el nombre de archivo). En la instalación se crea un fichero de configuración para el host virtual por defecto de nombre `default`.

`sites-enabled/` contiene enlaces simbólicos a los lugares `sites-available` que deseamos activar. Se incluyen en `apache2.conf` por la directiva

```
Include /etc/apache2/sites-enabled/[^.#]*
```

De igual manera que antes, disponemos de dos herramientas que nos facilitan este trabajo:

`a2ensite site` para crear los enlaces simbólicos asociados al sitio.

`a2dissite site` para eliminar los enlaces simbólicos para ese sitio.

↬ Por ejemplo, si tenemos un host virtual (trataremos esto después) de nombre *matematicas*, creamos su fichero de configuración de nombre *mate* (notar que los nombres no tienen por qué coincidir) en el directorio `/etc/apache2/sites-available`. Para activarlo ejecutaremos (reiniciando Apache para que los cambios sean efectivos)

`a2ensite mate` para crear los enlaces simbólicos asociados a "matematicas" en `/etc/apache2/sites-enabled`

`a2dissite mate` para eliminar los enlaces simbólicos para el *host virtual* "matematicas".

[18] Más adelante ampliaremos sobre este tema

17.4. /etc/http/httpd.conf

Como ya se ha comentado en la introducción, nos centraremos en documentar este fichero. Si optáis por trabajar con Debian sólo hay que adecuar lo aquí explicado a los ficheros de configuración antes comentados. El que se haga así se justifica desde la perspectiva de que es el sistema más estándar y documentado de trabajar con el servidor Web Apache.

Además, en las notas a pie de página explicitaremos las diferencias respecto a la configuración por defecto con Debian.

El archivo `httpd.conf`[19] está bien comentado y es bastante autoexplicativo. La configuración predeterminada funciona para los ordenadores de la mayoría de los usuarios, así que probablemente no se necesitará cambiar ninguna de las directivas en el fichero `httpd.conf`. Sin embargo, es bueno conocer las opciones de configuración más importantes.

Consta de tres secciones:

- Configuración global.
- Configuración del servidor principal.
- Configuración de los Servidores Virtuales.

Antes de modificar el fichero `httpd.conf` es conveniente hacer una copia del fichero original, dándole por ejemplo, el nombre `httpd.conf.ori, httpd.conf.20050319` u otro que nos sea significativo[20]. Si cometemos un error mientras estamos modificando el fichero de configuración, no debemos preocuparnos, porque siempre dispondremos de una copia de seguridad.

Si cometemos un error y nuestro servidor web no funciona correctamente, el primer sitio donde acudir es a lo último que acabamos de modificar en `httpd.conf`. Después podemos consultar el fichero de archivo de errores[21] (`/var/log/httpd/error_log`), las últimas entradas deberían servirnos de ayuda para saber lo que ha pasado.

La configuración de Apache se basa en una serie de directivas que tienen posibilidad de ser usadas dentro de un contexto, es decir, un ámbito en el que pueden ser aplicadas. Hay cuatro posibilidades que no son excluyentes:

- configuración global del servidor,
- secciones para configurar los host virtuales,
- secciones de configuración de directorios
- archivos `.htaccess`

  ↬ Por ejemplo, la directiva `ErrorLog` que nos permite establecer la ubicación del fichero para el registro de error, sólo se puede usar en la configuración global del servidor o al configurar los host virtuales.

A continuación se dan breves descripciones de las directivas incluidas en el fichero `httpd.conf`, ordenadas según se encuentran en él. Para una referencia más amplia de algunas de estas directivas véase la documentación instalada (`http://localhost/manual/es/mod/quickreference.html`).

ServerRoot El comando `ServerRoot` se va a referir al directorio principal donde se encuentran todos los ficheros de configuración y trabajo del servidor. Su valor es `/etc/httpd`[22].

[19] Recordar que como material adicional en Moodle disponéis del fichero de configuración de Fedora (véase el apéndice B en la página 657).

[20] El objetivo es poder volver al estado original en caso de que algo vaya mal.

[21] `Debian: /var/log/apache2/error.log`

[22] `Debian: /etc/apache2`

User La directiva `User` establece el *userid* que utiliza el servidor para ejecutarse y responder a las peticiones. El valor de `User` determina el acceso que tendrá el servidor web a los ficheros y directorios en los que se encuentran las páginas. Cualquier fichero al que no pueda acceder este usuario, será inaccesible para el servidor web y como consecuencia, también inaccesible al visitante de la web. El comando predeterminado para `User` es `apache`[23].

El usuario `User` también es dueño de cualquier proceso CGI que arranque el servidor y no se le debería permitir ejecutar ningún código que no esté pensado para responder peticiones HTTP.

El proceso httpd padre se inicia como root durante operaciones normales, pero pasa al usuario apache inmediatamente. El servidor debe arrancar como root porque necesita un puerto por debajo de 1024 (el puerto por defecto es el 80). Los puertos por debajo de 1024 están reservados para el sistema, así que sólo se pueden usar si se es root. Una vez que el servidor se ha conectado al puerto, pasa el proceso a `User` antes de aceptar peticiones.

Group El comando `Group` es similar a `User`. `Group` establece el grupo en el que el servidor responde a las peticiones. El valor predeterminado del comando Group también es `apache`[24], en este caso como grupo, y no como usuario.

DocumentRoot `DocumentRoot` es el directorio que contiene la mayoría de los archivos HTML que se entregarán en respuesta a peticiones. El directorio predeterminado `DocumentRoot` es `/var/www/html`[25]. Por ejemplo, el servidor puede recibir una petición para el siguiente documento:

`http://localhost/prueba.html`

El servidor buscará el fichero en el siguiente directorio por defecto:

`/var/www/html/prueba.html`

Directory: Las etiquetas `<Directory /path/a/directorio>` y `</Directory>` se usan para agrupar directivas de configuración que sólo se aplican a ese directorio y sus subdirectorios. Cualquier directiva aplicable a un directorio puede usarse en las etiquetas `<Directory>`. Las etiquetas `<File>` pueden aplicarse de la misma forma a un fichero específico.

Por defecto, se aplican parámetros muy restrictivos al directorio raíz (`/`).

```
<Directory />
   Options FollowSymLinks
   AllowOverride None
</Directory>
```

Con la directiva `Options` establecemos qué características están disponibles para el directorio en el que se establece, su sintaxis es:

```
Options [+|-]opcion [+|-]opcion ...
```

y las opciones pueden ser

`All` Todas las opciones excepto para `MultiViews`. Es el entorno por defecto.

`ExecCGI` Se permite ejecutar scripts CGI usando `mod_cgi`.

`FollowSymlinks` El servidor seguirá los enlaces simbólicos en este directorio. Aunque el servidor siga los enlaces simbólicos, no cambia el nombre de path usado para comparar las secciones `<Directory>`. Esta opción se ignora si se establece dentro de una sección `<Location>`

[23] Debian: www-data
[24] Debian: www-data
[25] Debian: /var/www, aunque la orden RedirectMatch ^/$ /apache2-default/ hace que vaya a /var/www/apache2-default

`Includes` Se permiten inclusiones por parte del servidor proporcionadas por `mod_include`.

`IncludesNOEXEC` Se permiten inclusiones por parte del servidor, pero están desactivados `#exec cmd` y `#exec cgi`. Está activo para scripts CGI `#include` virtual desde directorios `ScriptAlias`.

`Indexes` Si hay una petición de una URL de un directorio y en él no hay `DirectoryIndex` (ej: `index.html`), `mod_autoindex` devolverá un listado formateado del directorio.

`MultiViews` Los contenidos negociados "MultiViews" se permiten usando `mod_negotiation`.

`SymLinksIfOwnerMatch` El servidor sólo seguirá los enlaces simbólicos para aquellos archivos o directorios que posean la misma identidad de usuario que el enlace. Esta opción se ignora si se establece dentro de una sección `<Location>`

Normalmente, si se pueden aplicar varias `Options` a un directorio sólo se usa la más específica, ignorándose las demás; las opciones no se mezclan. En cualquier caso, si todas las opciones de la directiva `Options` van precedidas por el símbolo + o -, se mezclarán. Cualquier opción precedida por + se añadirá a las opciones en vigor, y cualquiera precedida por -, se eliminará.

Tal cual está, es equivalente a

```
Options FollowSymLinks -ExecCGI -Includes -Indexes -Multiviews
```

e implicaría que está permitido atravesar los enlaces simbólicos en todo el sistema.

Con la opción *`AllowOverride`* puesta al valor *`None`* establecemos que el servidor no leerá el archivo especificado en `FileName` (por defecto, `.htaccess`). Esta directiva permite especificar qué partes del servidor pueden ser establecidas en los archivos `.htaccess`, los valores que puede tomar (además del comentado) son:

`AuthConfig` permite el uso de directivas de autorización (por ejemplo: `AuthName`, `AuthType`, `Require`, ...)

`FileInfo` permite el uso de directivas que establecen el tipo de documento (por ejemplo: `DefaultType`, `ErrorDocument`, `...`)

`Indexes` permite usar directivas para controlar la forma en que se realizan los índices de los directorios (por ejemplo: `AddIcon`, `IndexOptions`, etc)

`Limit` permite el uso de directivas para establecer el control de acceso (`Allow`, `Deny` y `Order`)

`Options` permite usar directivas que controlan opciones específicas del directorio (`Options` y `XBitHack`)

Con esta configuración, cualquier directorio del sistema que necesite valores más permisivos ha de ser configurado explícitamente.

El directorio cgi-bin está configurado para permitir la ejecución de scripts CGI, con la opción `ExecCGI`. Si se necesita ejecutar un script CGI en cualquier otro directorio, habrá que configurar ExecCGI para ese directorio. Por ejemplo, si cgi-bin es `/var/www/cgi-bin`, pero se quieren ejecutar scripts CGI desde `/home/usuario/cgi-bin`, se añadirá una directiva `ExecCGI` y un par de directivas `Directory` como las siguientes, en el fichero `httpd.conf`:

```
<Directory /home/usuario/cgi-bin>
   Options +ExecCGI
</Directory>
```

Para permitir la ejecución de scripts CGI en `/home/usuario/cgi-bin`, habrá que llevar a cabo pasos extra aparte de configurar `ExecCGI`. El valor de los permisos para scripts CGI y el recorrido entero a los scripts, debe ser de 0755 (accesible para el usuario que ejecuta apache). Además, el dueño del script y del directorio deben ser el mismo.

UserDir `UserDir` es el nombre del subdirectorio, dentro del directorio de cada usuario, donde estarán los archivos HTML que podrán ser servidos. Por defecto, el subdirectorio es `public_html`. Por ejemplo, el servidor podría recibir la siguiente petición:

`http://localhost/~usuario/prueba.html`

El servidor buscaría el fichero:

`/home/usuario/public_html/prueba.html`

En el ejemplo, `/home/usuario` es el directorio del usuario.

Hay que asegurarse que

- Los permisos sean los adecuados
  - De los directorios de usuario sean correctos (por ejemplo 711).
  - Los bits de lectura (r) y ejecución (x) deben estar activados en el directorio `public_html` (0755 valdrá).
  - El valor de los permisos con que se servirán los ficheros desde `public_html` debe ser 0644 por lo menos.
- Para **Fedora**[26] permitir el acceso usando el módulo `mod_userdir`. Con él conseguimos que Apache permita o deniegue esta forma de acceso. Así, si deseamos activar esta posibilidad, hemos de cambiar la sección como sigue:

```
<IfModule mod_userdir.c>
  #UserDir disable
  UserDir public_html
</IfModule>
```

y releer después la configuración del servicio[27].

ErrorLog `ErrorLog` nombra el fichero donde se guardan los errores del servidor. Por defecto, el fichero de error del servidor es `/var/log/httpd/error_log`[28]. El `log` de errores es un buen sitio para ver si el servidor genera errores y no se sabe muy bien qué pasó.

↪ Una línea de ejemplo puede ser

```
[Sun Mar 21 05:39:18 2004] [error] [client 66.90.73.73] Fi-
le does not exist:
/var/www/html/sumthin
```

CustomLog con esta directiva establecemos la ubicación y formato del archivo de registro de los accesos. Por defecto[29]

```
CustomLog log/ac-
cess.log "%h%l%u%t \"%r\"%>s%b \"%{Referer}i\" \"%{User-Agent}i\""
```

es decir[30]:

- registramos el host remoto (`%h`), la identidad del cliente (`%l`), si se necesita autenticación para la URL solicitada, el nombre de usuario (`%u`) y el tiempo de solicitud (`%t`).

[26] En Guadalinex se activa por defecto (si no fuese así: `#a2enmod userdir` y reiniciar el servidor). Para ver su fichero de configuración consultar:

`/etc/apache2/mods-available/userdir.conf`

[27] `# service httpd reload`

[28] `Debian: /var/log/apache2/error.log`

[29] Equivale a `combined`. Se define en el propio fichero de configuración de apache.

[30] No se analizan todos los posibles valores, sólo los que aparecen por defecto.

- se entrecomilla la primera línea de la solicitud (`%r`), almacenamos el estado devuelto por el servidor en respuesta a la solicitud (`%>s`) y los bytes enviados (`%b`)
- además de la cabecera enviada por el cliente al solicitar la página web, almacenamos la URL de la página solicitada (`%{Referer}i`) y el navegador web usado (`%{User-Agent}i`).

↬ Una línea de ejemplo:

```
80.83.190.1 - paco [21/Mar/2005:15:47:59 +0100] "GET /isoqlog/images/pk.gif
HTTP/1.0" 200 246 "http://www.midominio.com/isoqlog/" "Mozi-
lla/5.0 (X11; U; Linux
i686; es-ES; rv:1.4.1) Gecko/20031114"
```

17.4.1. Debian

Como ya hemos comentado, la forma de organizar la configuración del servidor Apache en Guadalinex es más modular que la Fedora (véase 17.3.2 en la página 303). Cuestiones específicas que merece la pena destacar o recordar respecto a lo ya estudiado:

- Recordar que el fichero principal de configuración del servidor es `/etc/apache2/apache2.conf` y que con varias directivas `Include` se incluyen en él los ficheros de configuración de
  - Los módulos activos del directorio `/etc/apache2/mods-enabled`
  - El fichero `/etc/apache2/httpd.conf` que en principio está vacio[31].
  - Los ficheros de configuración de los hosts virtuales activos: `/etc/apache2/sites-enabled`
- Como ya hemos comentado, en el fichero `/etc/apache2/sites-avalaible/default` definimos la configuración del host virtual por defecto. De él comentar sólo dos cuestiones:
  1. Con los asteriscos de las directivas

     ```
     NameVirtualHost *
     <VirtualHost *>
         ....
     </Virtualhost>
     ```

     indicamos que se aplica a cualquier dirección IP y puerto en los que escuche Apache. Es decir, se aplica a todos nuestros interfaces de red. Para saber más véase `http://httpd.apache.org/docs-2.0/es/mod/core.html#virtualhost`.
  2. Con la directiva

     ```
     RedirectMatch ^/$ /apache2-default/
     ```

     redirigimos las solicitudes de la página principal (especificada mediante una expresión regular[32]) a la nueva ubicación. Es decir, al escribir `http://www.midominio.org` se nos redirigirá a `http://www.midominio.org/apache2-default/`. Si bien lo mantendremos así, sería buena opción comentarla.

➥ **Para practicar: Web de usuarios**

1. Montar el servidor web Apache y comprobar que los usuarios del sistema pueden acceder a sus páginas web personales[33]. Supongamos que en nuestra máquina hay un usuario de nombre THALES

[31] Después veremos una práctica en la que lo modificamos.

[32] Con `^/$` indicamos la cadena que comienza (`^`) y termina (`$`) con `/`, es decir, la petición es `"/"`, como por ejemplo `http://localhost/`.

[33] Con Guadalinex sólo hay que realizar los apartados: a), b) y e)

a) Para el usuario THALES crear el directorio `$HOME/public_html`

```
$mkdir public_html
```

b) Poner en él un fichero html simple de nombre `index.html`, por ejemplo:

```
<html>
<body>
<h1>Esta es la web de thales</h1>
</body>
</html>
```

c) Modificar los permisos del `$HOME` de THALES, así como del directorio `public_html` para que Apache pueda acceder a él:

```
$chmod o+x $HOME
$chmod o+rx $HOME/public_html
```

d) Permitir que Apache acceda a directorios de usuario mediante `http://servidor_web/~usuario`. Para ello cambiemos la sección del fichero de configuración del servidor como sigue

```
<IfModule mod_userdir.c>
# UserDir disable
UserDir public_html
</IfModule>
```

y releer después la configuración del servicio

```
#apachectl restart
```

e) Comprobar que funciona apuntando con nuestro navegador a la página web

```
http://127.0.0.1/~thales
```

2. Si bien la solución anterior es la estándar, "afea" bastante eso de tener que escribir la virgulilla[34] para acceder a las Web de usuario. Veamos la forma de apañar el entuerto. Partimos de que se ha realizado la práctica anterior y

a) Tenemos que instalar `mod_perl`, de manera que podremos ejecutar código basado en este lenguaje interactuando con el servidor Apache. Para instalarlo:

1) Fedora

```
# apt-get install mod_perl
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
Se instalarán los paquetes NUEVOS siguientes:
  mod_perl
0 upgraded, 1 newly installed, 0 removed and 170 not upgraded.
Need to get 1486kB of archives.
After unpacking 3890kB of additional disk space will be used.
```

2) Guadalinex

```
# apt-get install libapache2-mod-perl2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes extras:
  libdevel-symdump-perl
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-perl2 libdevel-symdump-perl
0 actualizados, 2 se instalarán, 0 para eliminar y 0 no actualiza-
dos.
Necesito descargar 623kB de archivos.
Se utilizarán 2781kB de espacio de disco adicional después de desem-
paquetar.
¿Desea continuar? [S/n]
```

[34] Incluso es difícil de localizar a veces.

Y, puede que activar el módulo
`#a2enmod perl`

b) Añadamos en el fichero de configuración de apache (podemos optar por ponerlo en ambos sistemas en `httpd.conf`) el script de perl

```
<Perl>
        opendir H, '/home/';
        my @dir = readdir(H);
        closedir H;
        foreach my $u (@dir) {
                next if $u =~ m/^\./;
                if (-e "/home/$u/public_html") {
                        push @Alias, ["/$u", "/home/$u/public_html
                            /"];
                }
        }
</Perl>
```

En ambos sistemas, habrá que reiniciar el servicio:

```
#apache2ctl restart
```

Básicamente, con él lo que hacemos es leer los directorios de usuario que contiene un directorio `public_html` y crear Alias asociados a ellos de la forma

```
Alias /thales /home/thales/public_html
```

Es decir, cuando escribamos `http://localhost/thales`, se redirigirá al directorio para el que hemos creado el Alias (`/home/thales/public_html`)

¿Queda mejor así, verdad? **-:)** ■

17.5. Autenticación

Podemos conseguir que para acceder a determinados recursos un cliente tenga que autenticarse ante el servidor. Por ejemplo, si deseamos mantener información sensible en nuestro sitio web (el módulo encargado es `mod_auth`[35]). La información contenida en esa zona sólo deberá ser vista por el usuario o grupo que establezcamos.

El proceso de autenticación es simple. El cliente envía su nombre y contraseña[36]. A continuación Apache comprueba su archivo[37] de nombres y contraseñas cifradas para ver si el cliente tiene derecho a acceder.

Podemos establecer de dos formas diferentes la autenticación:

- Globalmente: agregando una sección `<Directory /path/a/directorio>` y `</Directory>` en nuestro archivo `httpd.conf`[38] por cada directorio que deseemos proteger.

  ↪ Por ejemplo con
  `<Directory /public/>`

[35] No hay que hacer nada para cargarlo.
[36] En estos casos, además debemos configurar para que utilice SSL, porque un usuario y una contraseña en protocolos no cifrados, durarán sin ser conocidos, menos que un cubito de hielo en agosto.
[37] Puede ser un archivo de texto o una base de datos
[38] En Debian `/etc/apache2/sites-available/default`

```
  AuthType Basic
  AuthName "Pagina de thales"
  AuthUserFile /var/www/passwd/.htpasswd
  AuthGroupFile /dev/null
  require user thales
</Directory>
```

conseguimos el mismo resultado que con la práctica que se realiza un poco más adelante en esta página.

- Usando los ficheros especiales `.htaccess`[39]. Las directivas que se pongan dentro de los ficheros `.htaccess` se aplicarán sólo al directorio que lo contiene, así como a todos sus subdirectorios. Los archivos `.htaccess` se leen cada vez que hay una petición de páginas y, por tanto, no hay que reiniciar el servidor Web para que se activen los cambios que realicemos en ellos.

Las directivas de autenticación del módulo `mod_auth` (módulo de autenticación de Apache) son:

AuthUserfile asigna el nombre del archivo de texto que contendrá los nombres de usuario y contraseñas usadas en la autenticación HTTP básica

AuthGroupFile asigna el nombre del archivo de texto que contendrá la lista de grupos de usuarios usadas en la autenticación HTTP básica.

↬ Una línea de este archivo (en él se crea un grupo de nombre curso con tres usuarios) puede ser

```
curso: thales mileto pitagoras
```

AuthAuthoritative toma los valores `On` y `Off` (por defecto está en `On`). Permite que si usamos en un directorio varios métodos de autenticación diferentes y falla el primero, se pase al segundo.

➥ **Para practicar** Crear un directorio con acceso restringido al usuario THALES

1. Creemos el directorio[40]:

   ```
   # mkdir /var/www/html/public
   ```

   y pongamos en él una página web simple (la de antes nos puede servir) de nombre `index.html`.

2. Creemos el directorio en donde almacenar las claves de acceso, por ejemplo:

   ```
   # mkdir /var/www/passwd
   ```

   Hay varias formas de trabajar con archivos de contraseñas. Si son "pocos" usuarios[41]:

[39] Se puede especificar cualquier otro nombre en la directiva `AccessFileName`, pero éste es el valor por defecto.

[40]
```
Debian: # mkdir /var/www/apache2-default/public
```

[41]

- Este "pocos" hay que entenderlo con cierta flexibilidad. Hasta 100 usuarios más o menos va "de muerte". Si el número de usuarios a autenticar es mucho mayor, mejor usar el módulo `mod_auth_mysql`.
- Si no queremos que se resienta la seguridad, es muy importante tener en cuenta que el fichero `.htpasswd` esté situado fuera de DocumentRoot
- En Debian también podemos usar

  ```
  # htpasswd2 -c /var/www/passwd/.htpasswd thales
  ```

```
# htpasswd -c /var/www/passwd/.htpasswd thales
```

Así creamos (`-c`) el archivo con el primer usuario y se nos pedirá la contraseña (hacer notar que no tiene por qué ser un usuario del sistema). Después, para añadir otros usuarios, el parámetro `-c` no hay que ponerlo[42].

3. Creemos en `/var/www/html/public`[43] el fichero `.htaccess`

```
# cat /var/www/html/public/.htaccess
AuthType Basic
AuthName "Pagina restringida de thales"
AuthUserFile /var/www/passwd/.htpasswd
AuthGroupFile /dev/null
require user thales
```

Comentemos un poco el fichero: con la directiva `AuthType` con el valor `Basic` indicamos que la contraseña se negociará en texto plano. En el cuadro de verificación de contraseña, veremos el texto "Página restringida de thales". Por último indicamos a Apache el archivo en donde buscar la contraseña, que el grupo no importa y que el nombre de usuario requerido es THALES.

4. Modifiquemos el fichero[44] `/etc/httpd/conf/httpd.conf`

Si desde nuestro navegador web intentamos cargar la página:

`http://127.0.0.1/public/index.html`

podremos cargarla sin problema. Esto se debe a que en el fichero

`/etc/httpp/conf/http.conf`

hay una sección[45] como la que sigue (pero con menos comentarios y en inglés):

```
<Directory "/var/www/html">
# Puede ser "None", "All", o cualquier ócombinacin de "Indexes",
# "Includes", "FollowSymLinks", "ExecCGI", o "MultiViews".
#
# Notar que "MultiViews" debe ser *íexplcitamente* llamado --- "Options
    All"
# no se lo proporciona.
# Si se descomenta, ípodramos ver el contenido de los subdirectorios
    para
# los que no haya fichero html de inicio y áadems ípodramos seguir
# enlaces ósimblicos (con el problema de seguridad que representa)
  # Options Indexes FollowSymLinks
#
# Controla équ opciones pueden omitir los archivos .htaccess de los
    directorios
# Puede ser "All", o cualquier ócombinacin de "Options", "FileInfo",
# "AuthConfig" y "Limit"
# En vuestro fichero áestar la ílnea que sigue y eso implica
```

[42] Por defecto en Fedora el fichero creado tiene de dueño al root y de permisos 611. De esa forma, el servicio httpd puede leerlo sin problemas.

Como el servicio httpd se ejecuta como usuario apache, si usamos una versión de Apache que lo cree con permisos 600, el demonio no podrá leer su contenido. En este caso hay que cambiarlo de dueño (o relajarle los permisos)

```
# chown apache /var/www/passwd/.htpasswd
```

de esta forma el servidor Web podrá leer la contraseña introducida.

[43]

```
Debian: /var/www/apache2-default/public
```

[44] En Guadalinex el fichero a modificar es:
`/etc/apache2/mods-available/userdir.conf`

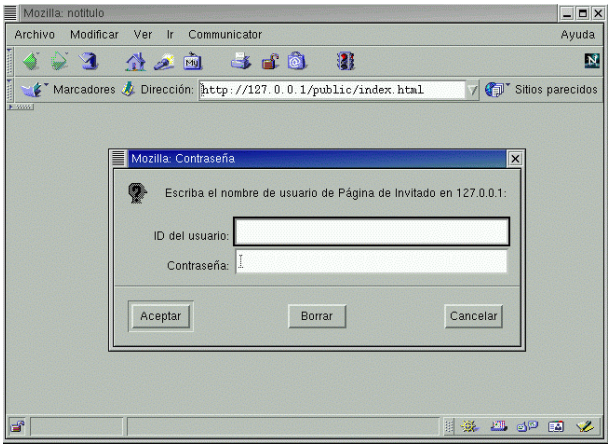
[45] En Guadalinex la sección se corresponde con `/var/www`

```
# que los ficheros .htacces no pueden modificar nada.
  # AllowOverride None
# Al poner esta otra permitimos que controlen la óautenticacin
  AllowOverride AuthConfig
#
# Controla équin puede obtener la respuesta de este servidor.
# Tal cual áest primero se procesan primero las directivas allow y
# édespus las deny, áadems, la seguna ílnea permite el acceso a todo el
    mundo
  Order allow,deny
  Allow from all
</Directory>
```

O sea que, cambiamos el valor `None` de la directiva `AllowOverride` a `AuthConfig` para que podamos controlar la autentificación desde los archivos `.htaccess`, hacemos que apache relea el fichero de configuración

```
#apachectl restart
```

y ya debería ir todo bien.



17.6. Host Virtuales

Mediante los *host* virtuales, Apache nos brinda la posibilidad de alojar varios dominios en una sola máquina.

↬ Supongamos que queremos albergar dos nombres de servidor web en la misma máquina `servidor1.midominio.com` y `servidor2.midominio.com`[46] respondiendo los mismos a una sola IP y con una sola instancia de Apache configurada y que responde las llamadas por el puerto 80. Usando host virtuales, podemos conseguir que en el caso de que sea invocado `servidor1.midominio.com` vaya a leer los archivos en el directorio que hayamos configurado como `DocumentRoot`, y consecuentemente los mande al navegador de quien lo haya pedido, y en el caso que sea convocado como `servidor2.midominio.com`, vaya a leer los archivos a otro directorio. Es decir, habrá un directorio (`DocumentRoot`) para cada uno de los servidores virtuales que definamos.

Apache soporta dos tipos de host virtuales:

[46] Puede ser también de otro dominio totalmente distinto, como por ejemplo, servidor2.otrodominio.org

Host virtuales basados en nombres permiten alojar varios nombres de host (o dominios) en una misma máquina

Host virtuales basados en IP una máquina responde a diferentes direcciones IPs. Ya que las direcciones "públicas" no las regalan, no es lo más habitual para un centro y no lo vamos a estudiar.

En general, el caso más interesante es el primero (que una sola máquina responda a varios nombres) y es el que vamos a analizar.

➥**Para practicar:** vamos a configurar Apache para que responda de forma diferente cuando se invoque como `www.midominio.com` y `tux.midominio.com`. [47] Para eso supondremos que trabajamos sobre la IP 172.26.0.2.


El host virtual heredará los parametros del host principal que no se cambien para él.

1. El primer paso consiste en configurar los servicios DNS para que apunten a la misma dirección IP[48].
2.

Fedora: Modiquemos el fichero de configuración de apache como sigue[49]:

```
NameVirtualHost 172.26.0.2
<VirtualHost www.midominio.com>
        ServerAdmin webmaster@midominio.com
        DocumentRoot /var/www/htmlwww
        ServerName www.midominio.com
        ErrorLog /var/log/httpd/www-error.log
        CustomLog /var/log/httpd/www-access.log common
</VirtualHost>
<VirtualHost tux.midominio.com>
        ServerAdmin webmaster1@midominio.com
        DocumentRoot /var/www/htmltux
        ServerName tux.midominio.com
        ScriptAlias /cgi-bin/ /var/www/tux/cgi-bin
        Alias /images /var/www/tux/images
        ErrorLog /var/log/httpd/tux-error.log
        CustomLog /var/log/httpd/tux-access.log common
</VirtualHost>
```

Debian:

Podemos optar por crear dos ficheros de contenido

```
$cat /etc/apache2/sites-avalaible/www
NameVirtualHost 172.26.0.2
<VirtualHost www.midominio.com>
        ServerAdmin webmaster@midominio.com
        DocumentRoot /var/www/htmlwww
        ServerName www.midominio.com
        ErrorLog /var/log/apache2/www-error.log
```

[47] Podemos trasladar de forma inmediata el ejemplo al caso de que sean dominios diferentes.

[48] Si trabajamos sólo en local lo podemos hacer desde el fichero `/etc/hosts`

[49] Los directorios que aparecen en el ejemplo se tienen que crear previamente y hay que adecuarlos a nuestra configuración personal.

Si usamos

```
NameVirtualHost *
<VirtualHost *>
....
```

Apache escuchará en todos los interfaces de red que tengamos en la máquina.

```
        CustomLog /var/log/apache2/www-access.log common
</VirtualHost>
$cat /etc/apache2/sites-avalaible/tux
<VirtualHost tux.midominio.com>
        ServerAdmin webmaster1@midominio.com
        DocumentRoot /var/www/htmltux
        ServerName tux.midominio.com
        ScriptAlias /cgi-bin/ /var/www/tux/cgi-bin
        Alias /images /var/www/tux/images
        ErrorLog /var/log/apache2/tux-error.log
        CustomLog /var/log/apache2/tux-access.log common
</VirtualHost>
```

Y activarlos con

```
#a2ensites www
#a2ensites tux
```

3. Reiniciar el servidor para que lea los cambios realizados en el fichero de configuración.

Con la directiva `NameVirtualHost 172.26.0.2` le decimos a Apache que las peticiones para la dirección IP sean subdivididas por nombre. Opcionalmente puede añadirse también un puerto.

La sección `<VirtualHost>` estará identificada por la dirección IP del sitio que queremos que sirva Apache. La dirección IP tiene que ser la misma que la dirección definida en `NameVirtualHost`. El efecto de esto es que cuando Apache recibe una petición dirigida al nombre del `host`, comprueba los bloques `<VirtualHost>` que tienen la misma dirección IP que la declarada con la directiva `NameVirtualHost,` para encontrar uno que incluya el nombre de servidor (indicado en `ServerName`) que se ha solicitado. En caso que no usemos `NameVirtualHost`, Apache buscará un bloque `<VirtualHost>` con la dirección IP correcta y usa el `ServerName` en la respuesta.

En cada uno de los bloques modificamos la dirección de correo del administrador y el `DocumentRoot` para el *host* virtual especificado en `ServerName`. Además, optamos por variar el directorio que contendrá los scripts CGI (`ScriptAlias /cgi-bin/ /var/www/tux/cgi-bin`) y directorios de imágenes (`Alias /images /var/www/tux/images`). Optamos también por dividir los ficheros para almacenar la salida de errores y log, ya que esto nos puede ayudar a la hora de comprobar el comportamiento por separado de los distintos servidores virtuales.

Con la configuración anterior, cualquier acceso con un navegador web a la dirección IP o un nombre distinto de los definidos en el bloque `<VirtualHost>` dará como resultado que Apache nos dé acceso al primero de los servidores virtuales definidos.

Hay que destacar que las posibilidades que ofrece Apache para el tratamiento son amplias, permitiendo la combinación de las opciones basadas en nombre e IP en una misma instancia de Apache. ■

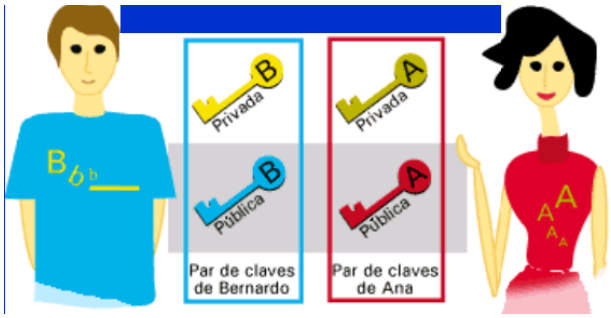
17.7. Servidores Seguros

Muy someramente, *https* se basa en los dos tipos de criptografía que conocimos en la entrega anterior: criptografía simétrica y criptografía asimétrica. ¿Porqué utilizar las dos? Porque aprovecha las ventajas de cada una y evita sus inconvenientes.

La criptografía asimétrica[50] es muy buena para la autenticación (ya que cada usuario protege su clave secreta), pero es muy lenta para el cifrado. Sin embargo, la criptografía simétrica es muy rápida en el cifrado y mala para la gestión de claves.

En el ejemplo, nuestros amigos Bernardo y Ana tienen su pareja de claves asimétricas (pública y privada), debiendo custodiar su clave privada para que nadie pueda conocerla.

[50] En la criptografía asimétrica, existen dos claves: una privada y otra pública. La clave privada debe permanecer bajo el exclusivo control del propietario y la pública puede (y debe) ser conocida por todos.



Así, el esquema utilizado por SSL, basado en el intercambio de claves Diffie-Hellman, quedaría como sigue: se utiliza la criptografía de clave pública o asimétrica, para realizar el intercambio seguro de una clave simétrica. No importa que el cifrado asimétrico sea lento, porque sólo se intercambia una clave, que es muy poca información. Esta clave simétrica, que es rápida para el cifrado, es la que se utilizará para cifrar los datos transmitidos en la sesión.

Un certificado digital contiene la clave pública, a la que se le añaden una serie de datos identificativos[51] y todo ello firmado por alguien en quien el resto de usuarios confían, denominado Autoridad Certificadora (CA). Esta tercera parte de confianza es la que permite que personas que no se conocen entre sí, puedan confiar en los certificados que se presentan el uno al otro. Por ejemplo, uno muy conocido puede ser la Fábrica Nacional de Moneda y Timbre[52].

La Autoridad Certificadora (AC[53]) lo que hace es firmar los certificados que emite. Un certificado contiene el nombre de la AC, el nombre del usuario, la clave pública del usuario y cualquier otra información, como puede ser un indicador de tiempo de validez. El certificado se firma con la clave privada de la AC. Todos los usuarios poseen la clave pública de la AC. Esto permite que cualquiera pueda comprobar la validez del certificado y si éste ha sido modificado.

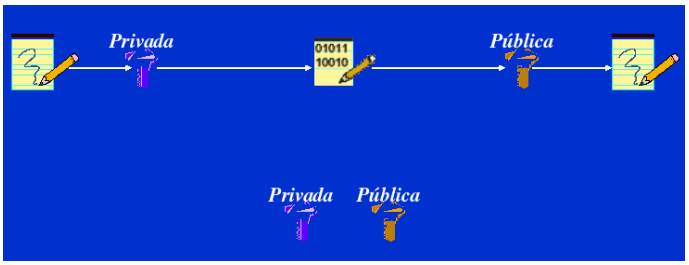
El proceso de firma electrónica se basa en lo siguiente:

1. Se cifra el mensaje[54] con la clave privada del remitente (la persona que firma).

2. Para comprobar la firma basta con descifrar con la clave pública del remitente, que es conocida. Si coincide, podemos asegurar dos cosas: que el mensaje no ha sido modificado y que

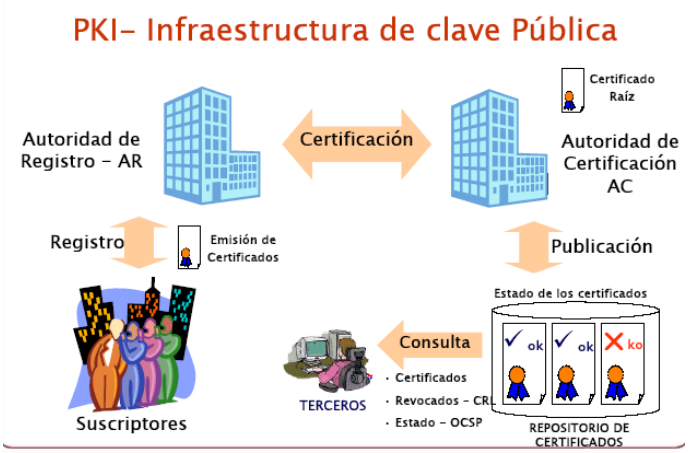
[51] Para un servidor pueden ser su nombre DNS (www.midominio.com). Para una persona física pueden ser su nombre, apellidos y DNI.
[52] Si confiamos en los billetes que hace, ¿no vamos a confiar en sus certificados...?
[53] O más común, CA, de Certification Authority
[54] Normalmente no es el mensaje lo que se cifra, sino una huella (o hash) del mismo.

la única persona que lo ha podido firmar ha sido el remitente, ya que es el único que posee la clave privada que se corresponde con la clave pública con que se ha descifrado.



Podemos solicitar un certificado a una Autoridad Certificadora, que nos puede llevar un dinero por ello, pero mediante openssl podemos erigirnos en nuestra propia Autoridad Certificadora[55], transmitir la información cifrada y sin gastarnos un solo euro.

Una infraestructura de clave pública (PKI[56]) es compleja. A continuación presentamos un gráfico con todos los elementos que pueden componerla.



No necesitaremos todos los elementos para montar nuestra PKI casera, aunque OpenSSL nos ofrece toda las funciones necesarias. Vayamos al grano.

Si queremos que nuestro servidor funcione con el protocolo SSL (https) para el envío de información cifrada, tenemos que instalar los paquetes `mod_ssl` y `openssl`.

Instalémoslos:

Fedora

```
# apt-get install mod_ssl openssl
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
openssl is already the newest version.
Se instalarán los siguientes paquetes extras:
  distcache
Se instalarán los paquetes NUEVOS siguientes:
  distcache mod_ssl
0 upgraded, 2 newly installed, 0 removed and 170 not upgraded.
Need to get 205kB of archives.
After unpacking 397kB of additional disk space will be used.
¿Quiere continuar? [S/n]
```

[55] Claro está, que solamente nuestros amigos y familiares confiarán en nuestra Autoridad Certificadora.
[56] De *Public Key Infraestructure*

El fichero de configuración para el servidor en modo SSL se encuentra en `/etc/httpd/conf.d/ssl.conf`. Para un modo de trabajo normal, no es necesario modificarlo. Solamente personalizamos, por ejemplo, dónde se encuentran las páginas a mostrar (**DocumentRoot** y **Directory**).

Bajo el directorio `/etc/httpd/conf/` nos encontramos el directorio `ssl.key` con el fichero `server.key`. Este fichero contiene las claves pública y privada para nuestro servidor. En el directorio `ssl.crt` aparece el fichero `server.crt` que contiene el certificado (clave pública más los datos identificativos del servidor, firmados por la CA). Si queremos construir nuestro propio certificado seguiremos los siguientes pasos. Nos situamos en el directorio `/etc/httpd/conf`. Modificamos el fichero `server.key`, por ejemplo con

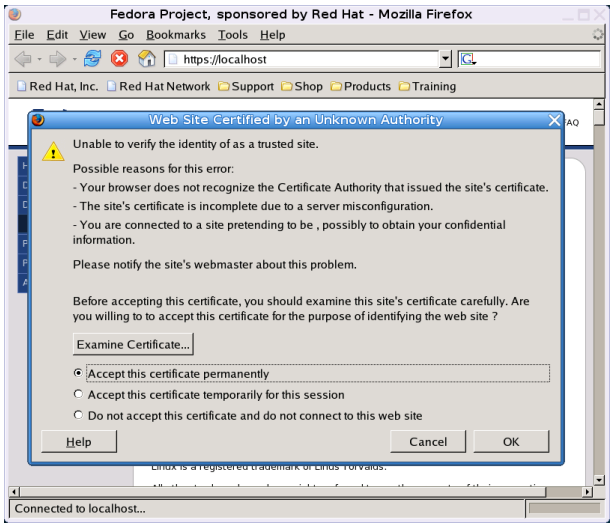
```
#touch ssl.key/server.key
```

y a continuación

```
# make testcert
umask 77 ; \
/usr/bin/openssl genrsa -des3 1024 > /etc/httpd/conf/ssl.key/server.key
Generating RSA private key, 1024 bit long modulus
.....++++++
.....++++++
e is 65537 (0x10001)
Enter pass phrase:
Verifying - Enter pass phrase:
umask 77 ; \
/usr/bin/openssl req -new -key /etc/httpd/conf/ssl.key/server.key -x509 -
    days 365 -out /etc/httpd/conf/ssl.crt/server.crt
Enter pass phrase for /etc/httpd/conf/ssl.key/server.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:
State or Province Name (full name) [Berkshire]:
Locality Name (eg, city) [Newbury]:
Organization Name (eg, company) [My Company Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:
Email Address []:
```

Donde podemos poner los datos que se ajusten a nuestro servidor.

Si reiniciamos apache, tendremos disponible la conexión por el puerto 443 mediante SSL. Podemos comprobarlo apuntando nuestro navegador a `https://localhost`.



Podemos conseguir que haya dos zonas en nuestro servidor: una en la que usaremos el servidor seguro (`/var/www/htmls`) y otra de acceso "no seguro" (`/var/www/html`). Para ello, sólo hay que modificar el fichero `/etc/httpd/conf.d/ssl.conf`, adecuando las líneas que siguen a nuestro objetivo

```
...
<VirtualHost _default_:443>
# General setup for the virtual host
DocumentRoot "/var/www/htmls"
ServerName www.midominio.org:443
ServerAdmin root@localhost
....
```

Debian

La configuración la vamos a realizar sobre el servidor web de ejemplo `www.midominio.com`. En primer lugar activemos el módulo `mod_ssl` con[57]

```
# a2enmod ssl
```

Situémonos en `/etc/apache2/ssl`, que estará vacío. Generamos las claves asimétricas y el certificado de la CA, y que en nuestro caso también van a ser del servidor SSL.

```
root@guadalinex:/etc/apache2/ssl# apache2-ssl-certificate
creating selfsigned certificate
replace it with one signed by a certification authority (CA)
enter your ServerName at the Common Name prompt
If you want your certificate to expire after x days call this programm
with -days x
Generating a 1024 bit RSA private key
.....++++++
.....++++++
writing new private key to '/etc/apache2/ssl/apache.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
```

[57] Para desactivarlo usaremos:
`# a2dismod ssl`

```
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:ES
State or Province Name (full name) [Some-State]:Andalucia
Locality Name (eg, city) []:MiLocalidad
Organization Name (eg, company; recommended) []:Thales-CICA
Organizational Unit Name (eg, section) []:MiIES
server name (eg. ssl.domain.tld; required!!!) []:www.midominio.com
Email Address []:webmaster@midominio.com
```

Con ello hemos creado el fichero `apache.pem` que contiene las claves (RSA PRIVATE KEY) y el certificado (CERTIFICATE) pertenecientes al servidor `www.midominio.com`, firmado por nosotros mismos[58].

```
root@guadalinex:/etc/apache2/ssl# more apache.pem

  -----BEGIN RSA PRIVATE KEY-----
  MIICXAIBAAKBgQCpAwn3fz9mqI+7UmD+kWsuUiw948U8wA43RHE/b0BErWBWwBNV
  ........
  ........
  rLvUPW5CBkOmlEe29xupk8wc39X1fcKgYrAtrPeb9vk=
  -----END RSA PRIVATE KEY-----
  -----BEGIN CERTIFICATE-----
  MIICuzCCAiQCCQDTe7ZZdsCq7TANBgkqhkiG9w0BAQQFADCBoTELMAkGA1UEBhMC
  ..........
  ..........
  +2jF+jYy+wyGHO3MSF+CwRRve4DEA8aAemNZEL4/aA==
  -----END CERTIFICATE-----
```

Pasemos a configurar el servidor SSL. Como base tomaremos el fichero de ejemplo proporcionado por Debian, que se encuentra en `/usr/share/doc/apache2/examples/ssl.conf.gz`. Lo descomprimimos y copiamos en `/etc/apache2/sites-available`. A continuación listamos las líneas de ese fichero que hemos descomentado/modificado para dejarlo listo para funcionar:

```
root@guadalinex:/etc/apache2/sites-available# more ssl.conf
#
...
<VirtualHost www.midominio.com:443>
...
DocumentRoot "/var/www/htmls"
ServerName www.midominio.com:443
ServerAdmin you@midominio.com
ErrorLog /var/log/apache2/error443_log
TransferLog /var/log/apache2/access443_log
...
SSLCertificateFile /etc/apache2/ssl/apache.pem
...
SSLCertificateKeyFile /etc/apache2/ssl/apache.pem
...
SSLCertificateChainFile /etc/apache2/ssl/apache.pem
...
SSLCACertificatePath /etc/apache2/ssl
SSLCACertificateFile /etc/apache2/ssl/apache.pem
```

[58] Para "aligerar" la salida hemos puestos puntos supensivos en lo que serían las claves generadas.

```
...
SSLVerifyClient none
...
```

Para que esté listo y en funcionamiento, creamos el directorio `/var/www/htmls`, donde vamos a situar las páginas del servidor seguro. Al menos, situad un fichero `index.html` para que podáis probar.

Añadimos a los sitios disponibles con:

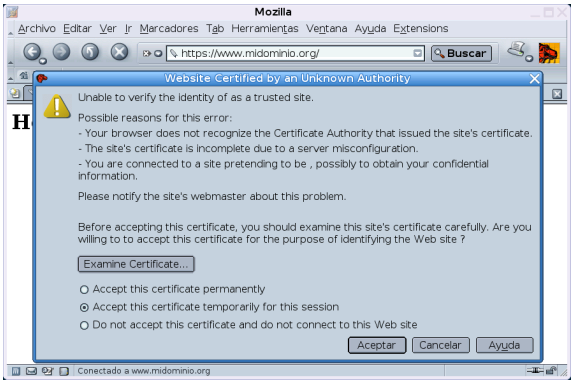
```
#a2ensite ssl.conf
```

Y reiniciamos apache para que obtenga la nueva configuración[59].

```
#/etc/init.d/apache2 restart
```

Sin más dilación, apuntamos nuestro navegador a `https://www.midominio.com`[60]

```
y ¡FUNCIONA!
```



¡Funcionó! ¡El Servidor Apache Seguro ha sido instalado con éxito!

El ajustar los valores a vuestro servidor Web se deja como ejercicio -:)

17.7.1. Autenticación del cliente mediante certificados

Lo visto hasta ahora, que no es poco, nos sirve para que las comunicaciones entre el servidor web y los navegadores se realice cifrada. Pero aún hay un paso más. El servidor se ha identificado mediante un certificado, pero podemos hacer que el cliente también lo haga.

Lo que tenemos que hacer es generar certificados para los clientes y poner la directiva SSLVerifyClient con el valor "require" (en vez del valor "none" que trae por defecto).

Adecuamos el entorno para que nuestra Autoridad Certificadora funcione correctamente (vamos a utilizar los parámetros que vienen por defecto en el fichero openssl.cnf)

Creamos los directorios y ficheros necesarios, basándonos en la estructura Debian.

```
#cd /etc/apache2/ssl
#mkdir demoCA
#mkdir demoCA/private
#mkdir demoCA/newcerts
#touch demoCA/index.txt
```

Copiar las claves de la CA al directorio correspondiente con el nombre cakey.pem

```
#cp  apache.pem demoCA/private/cakey.pem
```

Copiar el certificado de la CA al lugar necesario

```
#cp apache.pem demoCA/cacert.pem
```

[59] O con `#apache2ctl restart`

[60] Este nombre debe existir en el DNS o en nuestro fichero `/etc/hosts`.

Editar el fichero demoCA/serial y guardarlo en modo texto, tal que su contenido sea una línea con el valor 01, como se muestra a continuación.

```
#more demoCA/serial
01
```

Generaramos un Certificado para uso personal.

Generar clave privada

```
#openssl genrsa -des3 -out usuario.key 1024
Generating RSA private key, 1024 bit long modulus
............++++++
.....................................................................................++++++
e is 65537 (0x10001)
Enter pass phrase for usuario.key:
Verifying - Enter pass phrase for usuario.key:
```

Generar la petición de certificado en el formato estándar PKCS#10 para que la firme la Autoridad Certificadora. La extensión suele ser .csr, de Certificate Signing Request[61] y contiene la clave pública y los datos identificativos para que los firme la CA.

En el caso de una persona, los campos identificativos serán el Common Name (CN) con el nombre y DNI de la persona (esto es una convención utilizada, por ejemplo, por la Fábrica Nacional de Moneda y Timbre) y la dirección de correo electrónico, que servirá para enviar correos cifrados y firmados. En este ejemplo, la persona es Juan Perez Gomez con DNI. 29.999.999 y dirección de correo electrónico juan.perez@midominio.com.

El valor del campo Organization Name (Nombre de la Organización), debe coincidir con el de la Autoridad Certificadora. En nuestro ejemplo, Thales-CICA.

```
#openssl req -new -key usuario.key -out usuario.csr
Enter pass phrase for usuario.key:
You are about to be asked to enter information that will be incorpora-
ted
into your certificate request.
What you are about to enter is what is called a Distinguished Na-
me or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:ES
State or Province Name (full name) [Some-State]:Andalucia
Locality Name (eg, city) []:MiLocalidad
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Thales-CICA
Organizational Unit Name (eg, section) []:MiIES
Common Name (eg, YOUR name) []:Juan Perez Gomez DNI: 29.999.999
Email Address []:juan.perez@midominio.com
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

La Autoridad Certificadora firma (avala) el certificado.

[61] Petición de Firma de Certificado

```
$openssl ca -in usuario.csr -out usuario.crt
Using configuration from /usr/lib/ssl/openssl.cnf
Enter pass phrase for ./demoCA/private/cakey.pem:
DEBUG[load_index]: unique_subject = "yes"
Check that the request matches the signature
Signature ok
Certificate Details:
Serial Number: 2 (0x2)
Validity
Not Before: Feb 26 20:48:49 2005 GMT
Not After : Feb 26 20:48:49 2006 GMT
Subject:
countryName = ES
stateOrProvinceName = Andalucia
organizationName = Thales-CICA
organizationalUnitName = MiIES
commonName = Juan Perez Gomez DNI: 29.999.999
emailAddress = juan.perez@midominio.com
X509v3 extensions:
X509v3 Basic Constraints:
CA:FALSE
Netscape Comment:
OpenSSL Generated Certificate
X509v3 Subject Key Identifier:
C1:30:57:F7:CB:DC:D2:F4:CB:AC:C8:EF:02:39:9C:0C:A4:A2:87:34
X509v3 Authority Key Identifier:
keyid:6C:A3:10:13:42:EB:77:CD:6D:28:A4:F5:E5:D4:6E:5C:DC:EC:1A:86
DirName:/C=ES/ST=Andalucia/L=MiLocalidad/O=Thales-
CICA/OU=MiIES/CN=CA CEP/emailAddress=webmaster@midominio.com
serial:85:2D:C3:B5:5E:95:44:5E
Certificate is to be certified un-
til Feb 26 20:48:49 2006 GMT (365 days)
Sign the certificate? [y/n]:y
1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

El siguiente paso es exportar el certificado y las claves a formato PKCS#12. El fichero PKCS#12 contiene las claves asimétricas (privada y pública) y el certificado (con la clave pública y los datos del usuario, ya firmados por la autoridad certificadora).

```
$openssl pkcs12 -export -in usuario.crt -inkey usuario.key -
out usuario.p12
Enter pass phrase for usuario.key:
Enter Export Password:
Verifying - Enter Export Password:
```

El fichero PKCS#12 irá protegido por una contraseña. Podemos grabarlo en un soporte e importarlo desde un navegador,cliente de correo..., para ser utilizado. Debemos guardarlo bien, porque contiene nuestra clave privada.

Una vez importado en el navegador, nos conectamos a nuestro servidor que nos requerirá identificarnos con un certificado de cliente válido. Y si no, no nos permitirá entrar.

17.8. Reescribir las URL

Apache tiene mucha más funcionalidad que la hasta ahora comentada: SSI, proxy, ... Además permite reescribir las URL[62]. De esta forma, podemos redirigir una llamada al servidor a otro archivo distinto o a una URL diferente. Para hacer esto, es necesario usar las reglas de reescritura de URLs. Su estudio se escapa de lo que pretendemos en esta entrega, así que sólo vamos a analizar una de las múltiples posibilidades que presentan, usando un problema de ejemplo.

➥ **Problema:** en nuestro centro de enseñanza tenemos un dominio contratado de nombre `micentro.org`. Deseamos que los departamentos de matemáticas y lengua accedan a su Web de centro con una URL de la forma `www.matematicas.micentro.org` y `www.lengua.micentro.org`. Además, para facilitar la gestión de ambos sitios, deseamos que los `$HOME` de usuario coincidan con los directorios usados por Apache para servir las páginas Web[63]

Para resolver el problema debemos:

1. Añadir los registros correpondientes en nuestro DNS usando un registro CNAME (alias) para cada departamento didáctico.

2. Crear los `$HOME` de usuario adecuados para esos departamentos didácticos. Por ejemplo:

```
#mkdir /var/www/matematicas
#chown matematicas.matematicas /var/www/matematicas
#chmod 711 /var/www/matematicas
```

3. Añadir los usuarios al sistema y modificar el fichero `/etc/passwd` para que el `$HOME` de usuario de matemáticas sea el directorio antes creado.

```
#adduser matematicas
$cat /etc/passwd
  ...
matematicas:x:507:511::/var/www/matematicas:/bin/bash
```

4. Añadiremos las reglas de reescritura que siguen a nuestro fichero de configuración de Apache:

```
# por defecto está en Off, con esta directiva se
# activan las reglas de reescritura
RewriteEngine On
# Comprobamos si la variable de entorno HTTP-HOST es del tipo deseado
RewriteCond%{HTTP_HOST} ^www\.[^.]+\.micentro\.org
RewriteRule ^(.+)%{HTTP_POST}$1
RewriteRule ^www\.([^.]+)\.micentro\.com(.*) /var/www/$1/$2
```

Lo que faltaba: ¡expresiones regulares!. Sólo vamos a comentar un poco qué se hace para que no parezca un texto de Mortadelo y Filemón (`¡Arg#'¡@·`).

Con la primera regla comprobamos que la variable de `HTTP_HOST` es del tipo deseado

(`www.departamento.micentro.org`), después, encadenamos dos reglas (`C`) cuya misión es detectar las variables que se van a sustituir. En expresiones regulares la primera cadena de referencia (`$1`) se corresponde con la cadena encontrada en el primer paréntesis[64] `(^[.]+)` y `$2` con el segundo paréntesis[65] `(.*)`[66]

[62] El módulo encargado de esta labor es `mod_rewrite`.
En Guadalinex hemos de activarlo con

```
#a2enmod rewrite
```

y reiniciar el servidor.

[63] Notar que este problema se puede resolver fácilmente con `host virtuales`.

[64] En el ejemplo se corresponde con "matematicas"

[65] Si nuestra URL es de la forma `http://www.matematicas.micentro.org/algebra`, se corresponde con "algebra"

[66] El carácter punto (`.`) sustituye a cualquier carácter excepto el fin de línea. El asterisco (`*`) significa cero o más repeticiones de la expresión regular de que se trate y el símbolo de más (`+`), significa repeticiones de una o más veces de la expresión regular. Luego `.*` significa cualquier cadena de caracteres hasta el fin de línea.

Cuando desde un navegador Web escribamos `http://www.matematicas.micentro.org` se obtendría la página deseada.

No os asustéis, esto es solamente un ejemplo de la potencia que podéis llegar a tener, pero para la gran mayoría de los casos no tendréis que utilizarla. Tenéis un coche de 230 caballos, pero por ciudad debéis ir a 50 km/h. ■

17.9. Loganalizadores[67]

Todo servidor web que se precie tiene que disponer de estadísticas de uso para que el resto del mundo sepa la ingente cantidad de páginas web servidas. Vamos a analizar dos analizadores de accesos de Apache

17.9.1. webalizer

Se trata de un clásico (`http://www.mrunix.net/webalizer/`) . Si optamos por instalarlo desde internet[68] usaríamos:

```
# apt-get install webalizer
```

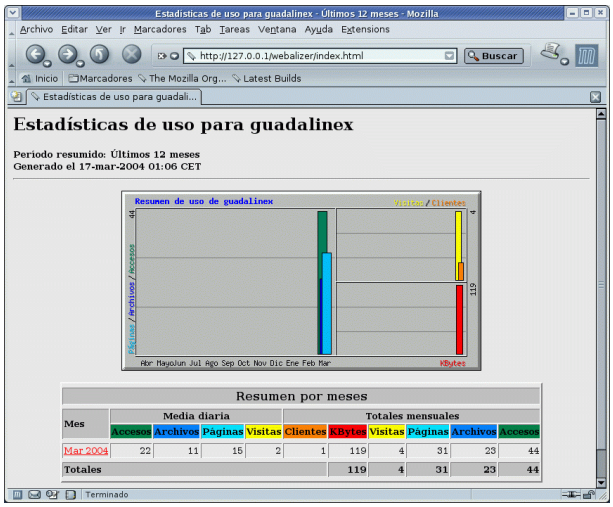
para ambas versiones de GNUlinux. En la instalación para Debian hemos de responder a un par de cuestiones que no presentan mayor dificultad (salvo adecuar el directorio DocumentRoot de Debian y la ruta del fichero de *log* de Apache que es `/var/log/apache2/access.log.1`). Ambas cuestiones se pueden modificar después.

Con

```
#webalizer
```

generaremos las estadísticas que obtiene el programa. En ambos sistemas añade una entrada a `/etc/cron.daily` y, por tanto, nuestras estadísticas de uso se actualizarán a diario.

[67] O analizadores de accesos
[68] En Fedora lo tenemos en los CDs: `webalizer-2.01_20-25.i386.rpm`



La configuración por defecto para acceder a ellas desde Fedora es `http://localhost/usage`.

El fichero de configuración de Webalizer es `/etc/webalizer.conf` y no presenta dificultad. En general, no es necesario cambiarlo. Si deseamos modificar los parámetros configurados en la instalación de Webalizer para Debian, lo haremos con las directivas:

```
LogFile       /var/log/apache2/acces_log
OutputDir     /var/www/html/usage
ReporTitle    Usage Statistic for
```

adecuándolas a nuestro sistema.

➥ **Para practicar:** Webalizer en castellano con Fedora

Vamos a "construirnos" la versión castellanizada de Webalizer. Para conseguirlo:

1. Nos bajamos el fichero fuente de la aplicación:

   a) Descomentamos la línea

   ```
   rpm-
   src http://ayo.freshrpms.net fedora/linux/3/i386 core updates freshrpms
   ```

   del fichero `/etc/apt/source.list`

   b) Nos bajamos el fichero fuente y lo desempaquetamos

   ```
   # apt-get update; apt-get source webalizer
   # rpm -ivh webalizer-2.01_10-25.src.rpm
   ```

2. Modifiquemos el fichero `/usr/src/redhat/SPEC/webalizer.spec`, en la sección `%configure` añadamos `--with-language=spanish`. Quedaría:

   ```
   %configure --enable-dns --with-dblib=/lib --with-language=spanish
   ```

3. Creemos el nuevo paquete:

   ```
   # rpmbuild -bb /usr/src/redhat/SPEC/webalizer.spec
   ```

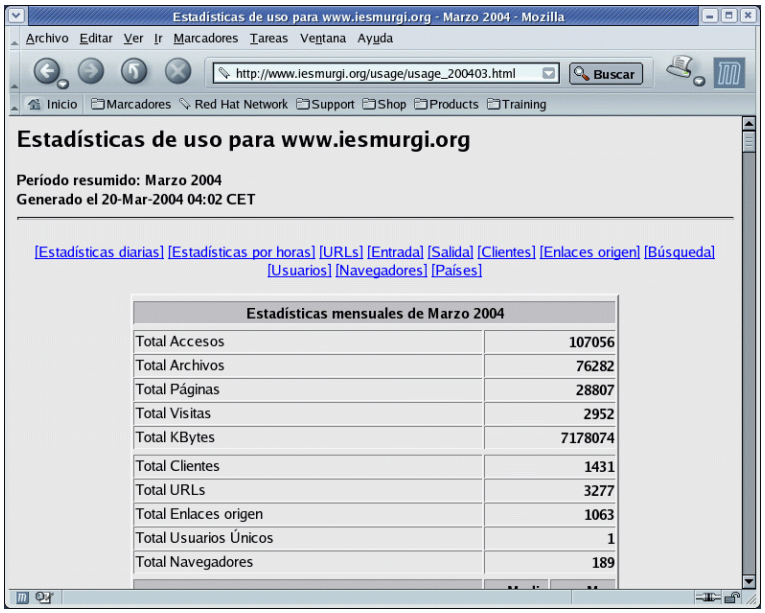
4. Lo instalamos[69]

[69] Si ya habíamos instalado el de Fedora habrá que eliminarlo

```
#rpm -e webalizer
```

```
#rpm -Uvh /usr/src/redhat/RPMS/i386/webalizer-2.01_10-25.i386.rpm
```

Como muestra:



17.9.2. awstats

Se trata de un loganalizador para Apache `http://www.awstats.org/` que se puede configurar para que nos muestre estadísticas para nuestro agente de transporte de correo[70] (MTA, como postfix, sendmail, etc.)[71]. Es más completo que el anterior.

Para Fedora.

Nos bajamos el paquete
`http://prdownloads.sourceforge.net/awstats/awstats-6.4-1.noarch.rpm`
y lo instalamos con

```
# rpm -ivh awstats-6.4-1.noarch.rpm
Preparing...                ########################################### [100%]
   1:awstats                ########################################### [100%]

     ----- AWStats 6.4 - Laurent Destailleur -----
     AWStats files have been installed in /usr/local/awstats

     If first install, follow instructions in documentation
     (/usr/local/awstats/docs/index.html) to setup AWStats in 3 steps:
     Step 1 : Install and Setup with awstats_configure.pl
     Step 2 : Build/Update Statistics with awstats.pl
     Step 3 : Read Statistics
```

[70] Más sobre esto un poco más adelante.
[71] Véase `http://cvs.sourceforge.net/viewcvs.py/awstats/awstats/docs/awstats_faq.html?rev=1.52`

Configuración. Nos situamos en el lugar adecuado

```
cd /usr/local/awstats
```

y ejecutemos el script de Perl[72]

```
# tools/awstats_configure.pl

----- AWStats configure 1.0 (build 1.4) (c) Laurent Destailleur -----
This tool will help you to configure AWStats to analyze statistics for
one web server. If you need to analyze load balanced servers log files, to
analyze downloaded log files without web server, to analyze mail or ftp log
files, or need to manage rotated logs, you will have to complete the config
file manually according to your needs.
Read the AWStats documentation (docs/index.html).

-----> Running OS detected: linux

-----> Check for web server install
configure did not find your Apache web server path.

Enter full config file path of you web server.
Example: /etc/httpd/apache.conf
Example: d:\Program files\apache group\apache\conf\httpd.conf
Config file path (CTRL+C to cancel):
> /etc/httpd/conf/httpd.conf

-----> Check and complete web server config file '/etc/httpd/conf/httpd.conf'
  AWStats directives already present.

-----> Update model config file '/etc/awstats/awstats.model.conf'
  File awstats.model.conf updated.

-----> Need to create a new config file ?
Do you want me to build a new AWStats config/profile
file (required if first install) [y/N] ? y

-----> Define config file name to create
What is the name of your web site or profile analysis ?
Example: www.mysite.com
Example: demo
Your web site, virtual server or profile name:
> midominio.com

Directory path to store config file(s) (Enter for default):
> /etc/awstats/awstats.midominio.com.conf
-----> Create config file '/etc/awstats/aws-
tats.midominio.com.conf/awstats.picasa.org.conf'
 Config file /etc/awstats/aws-
tats.midominio.com.conf/awstats.picasa.org.conf created.
-----> Restart Web server with '/sbin/service httpd restart'
Parando httpd:                                             [  OK  ]
Iniciando httpd:                                           [  OK  ]
-----> Add update process inside a scheduler
Sorry, configure.pl does not support automatic add to cron yet.
You can do it manually by adding the following command to your cron:
```

[72] Las líneas de tamaño mayor son las que tenemos que adecuar a nuestro sistema

```
/usr/local/awstats/wwwroot/cgi-bin/awstats -update -config=midominio.com
Or if you have several config files and prefer having only one command:
/usr/local/awstats/tools/awstats_updateall.pl now
Press ENTER to continue...

A SIMPLE config file has been created: /etc/awstats/awstats.midominio.com.conf
You should have a look inside to check and change manually main parameters.
You can then manually update your statistics for 'midominio.com' with command:
> perl awstats.pl -update -config=midominio.com
You can also read your statistics for 'midominio.com' with URL:
> http://localhost/awstats/awstats.pl?config=midominio.com

Press ENTER to finish...
```

Los únicos valores que hemos introducido se corresponden con: la ruta del fichero de configuración del servidor apache y el nombre del dominio.

Se creará el fichero de configuración para este dominio en `/etc/awstats/awstats.midominio.com.conf`. Tenemos que ajustarle el nombre del fichero de log de apache

```
LogFile="/var/log/httpd/acces_log"
```

y crear el directorio (de dueño apache) en donde se almacenarán los datos de los distintos dominios

```
# mkdir /var/lib/awstats
# chown apache /var/lib/awstats/
```

En la parte final de la salida del script aparece información relevante sobre la forma de activar el programa:

- Para que se actualicen las estadísticas usando `cron`, añadiremos (si deseamos que se actualicen cada hora[73]):

  ```
  # cat /etc/cron.hourly/awstats
  /usr/local/awstats/wwwroot/cgi-bin/awstats.pl -update -
  config=midominio.com
  ```

- Si tenemos distintos ficheros de configuración y preferimos tener sólo un comando:

  ```
  /usr/local/awstats/tools/awstats_updateall.pl now
  ```

- Para actualizar de forma manual las estadísticas de `midominio.com` escribiremos:

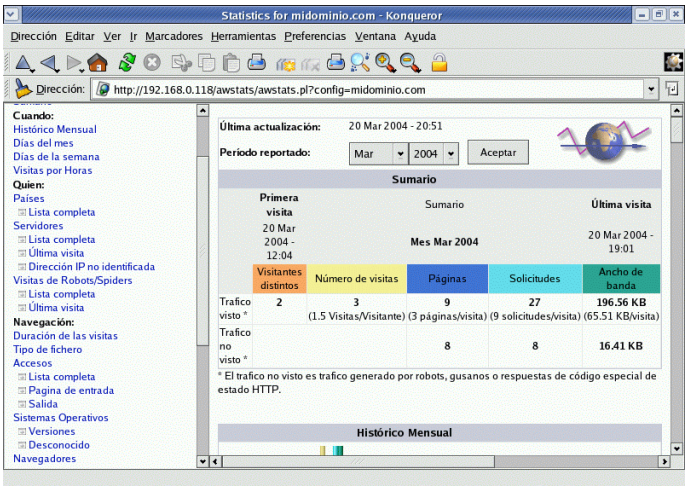
  ```
  perl /usr/local/awstats/wwwroot/cgi-bin/awstats.pl -update -
  config=midominio.com
  ```

- Podemos acceder a la Web de estadísticas de `midominio.com` usando la URL:

  ```
  http://localhost/awstats/awstats.pl?config=midominio.com
  ```

Para acceder a las estadísticas escribiremos la URL antes comentada:

[73] Los permisos han de ser adecuados.



Para Debian:

Para poder trabajar con él hay que adecuar nuestro Apache para que permita trabajar con scripts cgi:

```
#a2enmod cgi; apache2ctl graceful
```

La instalamos con

```
# apt-get update awstats
```

Crea automáticamente una entrada en `/etc/cron.d` que tendremos que ajustar a nuestro sistema:

```
# cat /etc/cron.d/awstats
0,10,20,30,40,50 * * * * www-data [ -x /usr/lib/cgi-bin/awstats.pl
-a -f /etc/awstats/awstats.conf -a -r /var/log/apache/access.log ]
&& /usr/lib/cgi-bin/awstats.pl -config=awstats -update >/dev/null
```

El cambio en el fichero anterior consiste en sustituir el nombre en donde se almacenan los log de apache (`/var/log/apache2/access.log`).

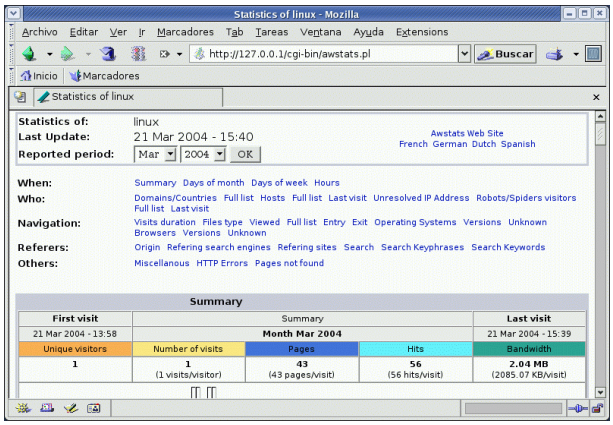
Nos crea un fichero de configuración (`/etc/awstats/awstats.conf`) y disponemos otro de ejemplo[74] `/usr/share/doc/awstats/examples/awstats.model.conf.gz`.

En `awstats.conf` hemos de modificar al menos las directivas:

```
LogFile="/var/log/apache2/acces.log"
SiteDomain="midominio.com"
```

Una vez que todo está como debe, podemos acceder a las estadísticas generadas por el programa escribiendo `http://www.midominio.com/cgi-bin/awstats.pl`

[74] En ese mismo directorio están algunas de las herramientas antes comentadas para Fedora. Por ejemplo: `awstats-update` que usaremos si deseamos actualizar de forma manual las estadísticas.


Statistics of linux - Mozilla
Archivo Editar Ver Ir Marcadores Tab Tareas Ventana Ayuda Extensions
http://127.0.0.1/cgi-bin/awstats.pl
Buscar
Inicio Marcadores
Statistics of linux
Statistics of: linux
Last Update: 21 Mar 2004 - 15:40
Reported period: Mar 2004 OK
Awstats Web Site
French German Dutch Spanish
When: Summary Days of month Days of week Hours
Who: Domains/Countries Full list Hosts Full list Last visit Unresolved IP Address Robots/Spiders visitors Full list Last visit
Navigation: Visits duration Files type Viewed Full list Entry Exit Operating Systems Versions Unknown Browsers Versions Unknown
Referers: Origin Refering search engines Refering sites Search Search Keyphrases Search Keywords
Others: Miscellanous HTTP Errors Pages not found
Summary
First visit
Summary
Last visit
21 Mar 2004 - 13:58
Month Mar 2004
21 Mar 2004 - 15:39
Unique visitors
Number of visits
Pages
Hits
Bandwidth
1
1 (1 visits/visitor)
43 (43 pages/visit)
56 (56 hits/visit)
2.04 MB (2085.07 KB/visit)

Capítulo 18

Correo electrónico

> Asegurarse de que el correo electrónico de los usuarios se envía y recibe correctamente, es uno de los trabajos más importantes de un administrador de sistemas, y que se hace extremadamente visible en caso de que las cosas vayan mal. *Administering E-mail.* Aeleen Frisch

18.1. Introducción

Reconocido como la aplicación más utilizada de Internet, junto con la todopoderosa Web, el correo electrónico es utilizado para prácticamente cualquier tipo de comunicación. Cada vez más, las empresas y organizaciones dependen de su buen funcionamiento para las relaciones entre sus empleados y con el exterior. Paraos a pensar cuántos mensajes mandáis y recibís a lo largo del día en el trabajo, en casa, desde un cibercafé... Y si se combinan con los SMS a móviles, realmente nos damos cuenta de que estamos un nuevo tipo de sociedad.

Una de los factores que han llevado al éxito al correo electrónico es su simplicidad de uso. Cualquier persona[1] con unas breves nociones de acceso al sistema operativo y de uso del programa de correo electrónico, rápidamente es capaz de enviar y recibir correos con una facilidad pasmosa. Sin embargo, no son tan conocidos los mecanismos que hacen que los mensajes lleguen a través de Internet al destinatario que se encuentra a cientos o miles de kilómetros de distancia. ¿Qué son SMTP, POP o IMAP, los agentes de transporte y los agentes de usuario? Enseguida lo sabremos.

Seguramente hoy habrás recibido varios, o puede que muchos, mensajes de correo electrónico. Para aceder a ellos y verlos, utilizas algún cliente de correo como Outlook, Eudora, Mozilla o Ximian Evolution. También puedes acceder por medio de un navegador de Internet, lo que se conoce como webmail.

Sin embargo, sea del tipo que sea el cliente, como mínimo hará las siguientes cosas:

* Presentar una lista de los mensajes que han llegado a tu dirección de correo electrónico, mostrando una cabecera que dice quién envió el mensaje, el motivo del mensaje y la fecha y hora.

* Permite seleccionar las cabeceras de los mensajes y leer el cuerpo del mensaje seleccionado.

* Permite componer nuevos mensajes y enviarlos. Indicamos al menos la dirección de correo del destinatario, el tema del mensaje y el contenido o cuerpo del mensaje.

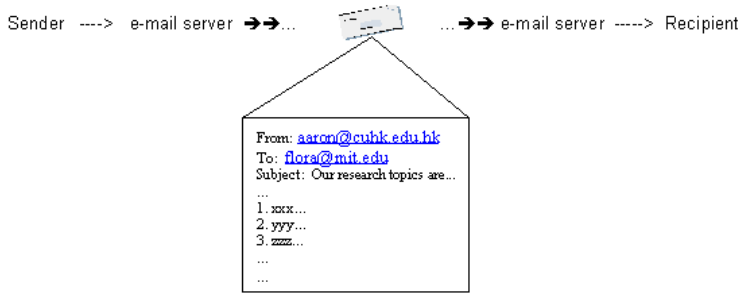
* Permite adjuntar ficheros a los mensajes y recuperar los ficheros adjuntos a los mensajes recibidos.

El formato de un mensaje de correo electrónico de Internet es simple: varios atributos obligatorios y otros opcionales, que forman una *cabecera*, separados por una línea en blanco del *cuerpo* del mensaje, que constituyen los datos objeto de la comunicación. Este formato viene definido en el RFC 2822[2]. La cabecera, al igual que en un sobre de correo postal, tiene toda la información

[1] Y si es joven, aún más.

[2] Este RFC ha sustituido al famoso RFC 822, que podemos encontrar en muchas referencias. RFC-2822 `http://www.faqs.org/rfcs/rfc2822.html`

que necesita el correo para llegar a su destino, o para su devolución al remitente en caso de que no haya sido posible su entrega.



Si en un sistema Linux enviamos un mensaje[3] y vemos el buzón donde se guardan los correos de un determinado usuario[4], podemos observar algo como lo siguiente:

```
From josber@midominio.com Sun Mar 21 22:44:17 2004
Return-Path: <josber@midominio.com>
Received: from greco.midominio.es (greco.midominio.com [195.123.25.23])
by mileto.cica.es (8.12.8/8.12.5) with ESMTP id i2LLiHJY006254
for <jabernal@mileto.cica.es>; Sun, 21 Mar 2004 22:44:17 +0100
Received: (from nobody@localhost)
by greco.midominio.com (8.8.5/8.8.5) id XAA11277
for <jabernal@mileto.cica.es>; Sun, 21 Mar 2004 23:03:40 +0100
From: josber@midominio.com
Message-Id: <200403212203.XAA11277@greco.midominio.com>
X-Authentication-Warning: greco.midominio.com: nobody set sender
          to <josber@midominio.com> using -f
Received: from alberti.midominio.com by greco.midominio.com via
          smap (V2.1/2.1+anti-relay+anti-spam)
id xma011274; Sun, 21 Mar 04 23:03:31 +0100
To: jabernal@mileto.cica.es
Subject: prueba
Date: Sun, 21 Mar 2004 21:44:26 GMT
X-Mailer: Endymion MailMan Standard Edition v3.0.35
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="MailMan_Boundary"

This is a multi-part message in MIME format.
--MailMan_Boundary
Content-Type: text/plain
Hola,
Te adjunto el fichero que solicitaste.
Un saludo.
--MailMan_Boundary
Content-Type: text/plain; name="hola.txt"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="hola.txt"
aG9sYQo=
--MailMan_Boundary--
```

[3]Cuando tengamos correctamente configurado nuestro sistema podremos practicar.

[4]Por ejemplo, en el directorio `/var/spool/mail`, y con el nombre de fichero de nuestro usuario. Atención: el superusuario podrá ver el correo de todos los usuarios, a menos que se envíe cifrado.

Si vamos desmenuzando el fichero, podemos ver que:

- La primera línea nos indica el inicio de este primer mensaje. Si hubiera varios mensajes en el buzón[5] del usuario, cada uno iría precedido por una línea de este tipo, con el remitente, la fecha y hora de recepción.
- El resto corresponde al mensaje, que se divide en dos partes, una cabecera y un cuerpo. La cabecera se compone de líneas con pares `campo:valor`. Tras una línea en blanco aparece el cuerpo del mensaje.
  - En la cabecera, el primer campo que aparece en nuestro ejemplo es el campo `Return-Path` que indica a dónde debe devolverse el mensaje en caso de que no se pueda entregar satisfactoriamente. En caso de que no exista este campo, se utilizaría el valor del campo `From`.
  - Cada campo `Received`, nos indica el periplo que ha realizado el mensaje hasta llegar a su destino. Por cada estafeta de correos[6] por las que pasa, se le añaden nuevas líneas que indican de dónde se ha recibido y a quién debe entregarla. En este caso concreto, vemos que el viaje del mensaje ha sido desde la máquina `alberti.midominio.com` hasta `greco.midominio.com` en un primer paso. De ahí llega a `mileto.cica.es` que es la máquina donde se encuentra su destino.
  - El campo `From` indica el remitente del correo. En este caso josber@midominio.com.
  - El campo `To` nos indica a quién va dirigido el mensaje. En este caso a jabernal@mileto.cica.es.
  - El campo `Subject` nos sirve para indicar brevemente el tema del mensaje.
  - El campo `MIME-Version` nos indica que el cuerpo se encuentra expresado en el formato MIME que se describe en las RFC siguientes: RFC2045, RFC2046 y RFC2049. Este formato nos permite enviar imágenes, ficheros binarios, de música... codificados de manera que pueden ser representados por caracteres ASCII que son los que son transportados por el correo electrónico. Normalmente en codificados en Base64
  - El campo `Content-Type` nos dice que el cuerpo se divide en varias partes separadas por la palabra `MailMan_Boundary`
- Tras la línea en blanco, viene el cuerpo del mensaje, que es el verdadero contenido. Éste consta de dos partes separadas por la palabra `MailMan_Boundary`:
  - La primera parte es un texto plano, que es lo que escribimos normalmente en el cuerpo del mensaje.
  - En la segunda parte, se nos indica que hay un fichero adjunto que se corresponde con un fichero de texto plano cuyo nombre será `hola.txt`. El fichero se ha mandado como adjunto[7] y está codificado para su transmisión. El contenido es `aG9sYQo=`, que corresponde a la codificación en base64 del contenido del fichero[8].

18.1.1. ¿Cuántos invitados tenemos para cenar?

El emisor de un mensaje de correo electrónico utiliza un programa para crear y enviar el correo. Este programa se denomina Agente de Usuario de Correo[9].

Una vez creado, el mensaje se traslada hacia el destinatario sobre un medio de transporte, que puede ser Internet o una red privada, utilizando uno o varios Agentes de Transporte de Correo[10].

[5] *Mailbox*: Nombre que recibe el fichero o estructura que guarda los mensajes de cada usuario.
[6] Que son los servidores con los agentes de transporte.
[7] *Attachment*, en inglés.
[8] El contenido era simplemente "hola".
[9] *Mail User Agent* (MUA)
[10] *Mail Transfer Agent* (MTA)

El mensaje al fin es entregado al destinatario usando un Agente de Entrega de Correo[11] y almacenado en su buzón[12] hasta que éste lo lee utilizando nuevamente un Agente de Usuario.

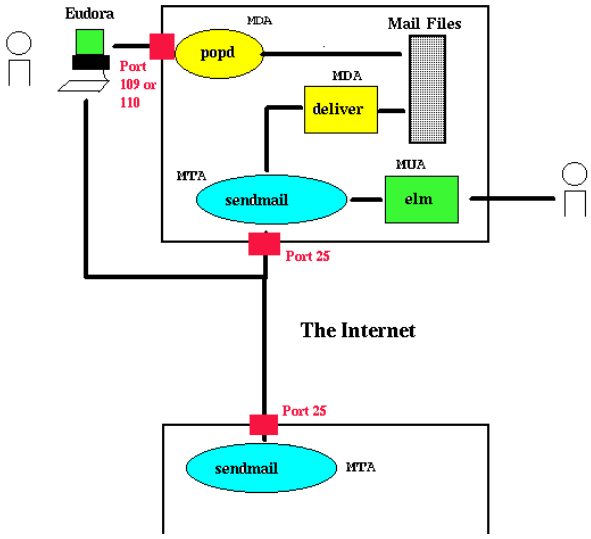
Así, los componentes del sistema de correo son:

***Mail User Agent* (MUA)** Un programa usado para crear y recibir mensajes de correo electrónico.

***Mail Transfer Agent* (MTA)** El medio por el cual los mensajes de correo electrónico se transfieren de máquina en máquina hasta llegar a su destino.

***Mail Delivery Agent* (MDA)** El programa que se encarga de depositar el mensaje de correo en el buzón del destinatario, una vez entregado por un MTA al servidor de correo.

En la siguiente figura podemos observar cómo interaccionan algunos Agentes de Usuario (MUA) en verde (Eudora y elm), Agentes de Entrega (MDA) en amarillo (popd y deliver) y Agentes de Transporte (MTA) en azul (sendmail).



Para cada una de las categorías de programas, tenemos varias alternativas, dependiendo de nuestras preferencias y de cuál sea nuestro sistema operativo. Incluso hay programas, que realizan más de una de las funciones. Como ejemplos de programas dentro de cada una de ellas, podemos citar:

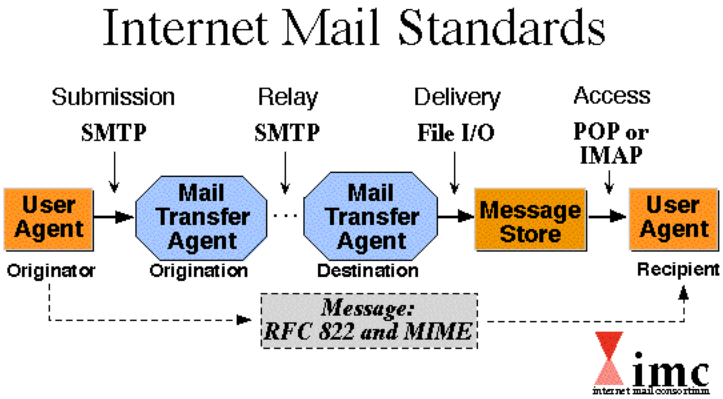
MUA Outlook, Eudora, Netscape Mail, Mozilla, Sylpheed, ximian-evolution, mail, elm, mutt, pine, mh/nmh.

MTA Sendmail, Postfix, smail, qmail, exim.

MDA mail, cyrus, deliver, mail.local, procmail, fetchmail.

El mecanismo funciona de la siguiente manera. El Agente de Usuario confecciona el mensaje y lo envía mediante el protocolo SMTP a un Agente de Transporte. Este Agente de Transporte lo envía a través de la red a otros Agentes de Transporte, hasta que al final llega al Agente de Transporte que corresponde al destinatario del mensaje. El Agente de Entrega[13] lo deposita en el buzón de correo electrónico del destinatario. Allí el mensaje espera hasta que el destinatario acceda con un Agente de Usuario mediante los protocolos POP o IMAP.

[11] *Mail Delivery Agent* (MDA)
[12] *Mailbox*, en inglés.
[13] Muchas veces este Agente de Entrega se encuentra integrado con el Agente de Transporte.



Podemos obervar cómo el contenido del mensaje llega desde el remitente al destinatario siguiendo los formatos RFC2822 y MIME.

18.1.2. ¿Cómo se encamina el correo?

Cuando un Agente de Transporte recibe el encargo de transportar un mensaje de correo electrónico, supongamos que a la dirección linux@cica.es, lo primero que hace es comprobar si es él mismo el encargado de manejar el correo para *el* dominio *cica.es*. Si lo es, no tiene que encargarle el trabajo a otro agente de transporte, sino que él mismo será capaz de entregarlo. En caso de que no lo sea, le pregunta al sistema DNS qué máquina o máquinas son las encargadas de ello. Los registros MX[14] son los que ofrecen esta información.

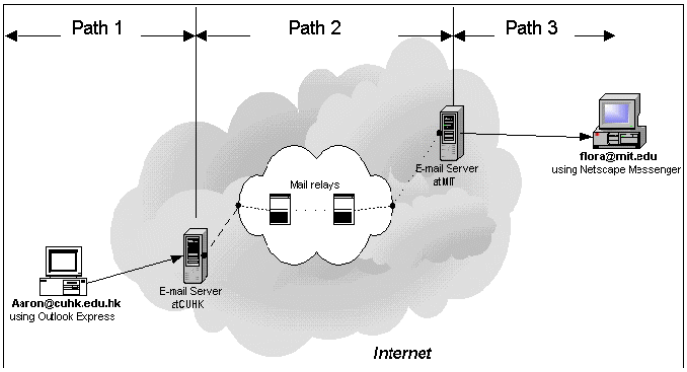
Por ejemplo, una consulta al DNS sobre los registros MX del dominio `cica.es` nos daría:

```
cica.es. 961 IN MX 15 mailgw2.cica.es.
cica.es. 961 IN MX 20 mail.rediris.es.
cica.es. 961 IN MX 10 mailgw.cica.es.
```

Veamos qué implicaciones tienen estos datos en el correo. Cada una de las líneas indica que el registro MX designa a una máquina que recibe correo para el dominio cica.es. De todas ellas, la preferente será la que tenga la prioridad más baja (el valor 10 será el preferido antes que el 15, y éste antes que el 20), y si no está disponible se irá al siguiente con menor prioridad. En este caso, si no hay ningún problema en la red o la máquina, `mailgw.cica.es` será la máquina que recibirá los correos para la dirección linux@cica.es.

En el caso de que no exista registro MX y el destino sea un host[15] (por ejemplo `mileto.cica.es`) también se le puede enviar correo electrónico a esa máquina concreta, como por ejemplo a la dirección linux@mileto.cica.es.

En la siguiente figura observamos que el correo puede pasar por varios Agentes de Transporte (MTA) hasta llegar a su destino.

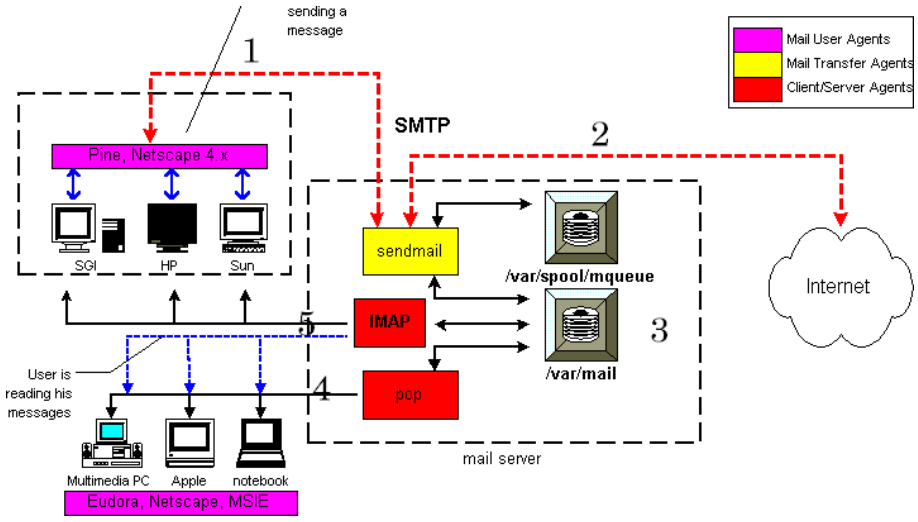


[14] *Mail eXchanger*

[15] Existiendo un registro tipo A en el DNS para él.

18.1.3. Eso no es todo, aún hay más

Descendamos a un plano más práctico y veamos qué programas y protocolos nos encontramos.



En la figura anterior, podemos ver los diferentes pasos en el proceso de envío y recepción del correo electrónico.

- En el paso 1, vemos que mediante un Agente de Usuario (pine, Netscape, Outlook...) componemos un mensaje y lo enviamos mediante el protocolo SMTP a un servidor de correo.
- En el paso 2, el Agente de Transporte debe mirar a qué otro Agente de Transporte debe enviarlo en caso de que él no sea el receptor.
- En el caso de que el correo para la dirección del destinatario lo gestione ese servidor de correo, en el paso 3 se guarda en el buzón correspondiente.
- Cuando el destinatario quiere leer su correo, lo hace bien mediante el protocolo POP (paso 4) o el protocolo IMAP (paso 5).

Protocolo SMTP

SMTP (*Simple Mail Transfer Protocol*[16]) es un protocolo cliente-servidor basado en TCP. Su funcionamiento es muy simple. Una vez que se establece la conexión, el cliente envía comandos al servidor con la cabecera y el cuerpo del mensaje.

Este protocolo se basa en el envío de comandos de cuatro caracteres y códigos de respuesta de tres dígitos, más una serie de comentarios que lo hacen más legible. Acualmente se utiliza una versión conocida como SMTP Extendido o ESMTP.

A continuación mostramos una conversación entre un cliente y un servidor SMTP. Con una fuente un poco mayor mostramos los comandos que vamos tecleando

```
[root@linux entrega04-3]# telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 linux.midominio.org ESMTP Send-
mail 8.12.10/8.12.10; Wed, 24 Mar 2004 21:30:4 8 +0100
EHLO linux.midominio.com
```

[16] Protocolo Simple de Transferencia de Correo

```
250-linux.midominio.com Hello localhost.localdomain [127.0.0.1], plea-
sed to meet you
250-ENHANCEDSTATUSCODES
250-PIPELINING
250-8BITMIME
250-SIZE
250-DSN
250-ETRN
250-AUTH DIGEST-MD5 CRAM-MD5 LOGIN PLAIN
250-DELIVERBY
250 HELP
MAIL From:<jabernal@linux.midominio.com>
250 2.1.0 <jabernal@linux.midominio.com>... Sender ok
RCPT To:<linux@mileto.cica.es>
250 2.1.5 <linux@mileto.cica.es>... Recipient ok
DATA
354 Enter mail, end with "." on a line by itself
saludos
.
250 2.0.0 i2OKUlgx002363 Message accepted for delivery
QUIT
221 2.0.0 linux.midominio.com closing connection
Connection closed by foreign host.
```

Vamos a actuar como lo haría un Agente de Usuario. Nos conectamos al puerto 25 de nuestra máquina local que es donde escucha el demonio SMTP por convención. La línea precedida por el código de respuesta 220 nos dice que el servidor es `linux.midominio.com`, que habla el protocolo ESMTP y que es Sendmail.

- Somos corteses y le mandamos el comando Hola[17]. Nos contesta con las capacidades que tiene el servidor.
- Con el comando `MAIL` le indicamos que vamos a enviar un mensaje e indicamos el remitente con el valor `From:`.
- El destinatario lo indicamos con el comando `RCPT` y con el valor del campo `To:`.
- Tras el comando `DATA` iniciamos el cuerpo del mensaje que acabaremos con una línea que empieza por punto y finalmente nos salimos con `QUIT`.

Como véis, SMTP no es complejo, pero tampoco es como para que estemos hablando SMTP con todo bicho viviente que nos encontremos. El cliente de correo lo hablará por nosotros y solamente tendremos que preocuparnos de rellenar los campos correspondientes a la información.

El protocolo SMTP, como habéis visto, es fácil de engañar. Hasta hace poco tiempo no se ha tomado en serio la seguridad y la consecuencia son los virus, spam, hoax y demás jungla. Afortunadamente, poco a poco va incorporando medidas de seguridad como autenticación, cifrado, certificados digitales, etc. en el correo.

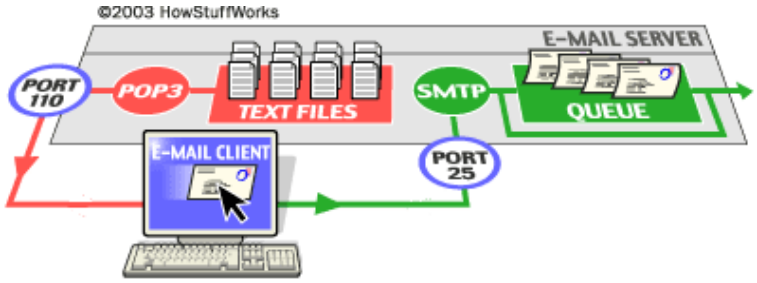
Protocolo POP

El protocolo POP (*Post Office Protocol*[18]) se diseñó para permitir una gestión del correo sin tener que estar conectados continuamente con el servidor. La idea es conectarse con el servidor, descargarse al ordenador local los correos electrónicos y poder trabajar con ellos sin necesidad de

[17] No, no es un error, es así EHLO
[18] Protocolo de Oficina de Correos

estar conectados con el servidor continuamente, ni siquiera conectados a la red[19]. Lo normal es que el correo al descargarlo, se borre del servidor, aunque hay opciones para conservarlo allí.

La siguiente figura es muy descriptiva de cómo el cliente de correo (MUA) envía al servidor (MTA) el correo al puerto 25 mediante el protocolo SMTP y lo recibe conectándose al puerto 110 mediante el protocolo POP.



Veamos un ejemplo de una sesión POP a "pelo[20]". Como con el protocolo SMTP, nuestros comandos los pondremos en una fuente mayor.

```
[root@linux entrega04-3]# telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
+OK POP3 localhost.localdomain v2003.83rh server ready
user juanjo
+OK User name accepted, password please
pass secreto
+OK Mailbox open, 6 messages
list
+OK Mailbox scan listing follows
1 615
2 687
3 842
4 913
5 1027
6 1109
.
retr 1
+OK 615 octets
Return-Path: <jabernal@linux.midominio.com>
Received: from linux.midominio.com (localhost.localdomain [127.0.0.1])
by linux.midominio.com (8.12.10/8.12.10) with ESMTP id i2NN73UL003368
for <jabernal@linux.midominio.com>; Wed, 24 Mar 2004 00:07:04 +0100
Received: (from jabernal@localhost)
by linux.midominio.com (8.12.10/8.12.10/Submit) id i2NN72dK003366
for jabernal; Wed, 24 Mar 2004 00:07:02 +0100
Date: Wed, 24 Mar 2004 00:07:02 +0100
From: jabernal@linux.midominio.com
Message-Id: <200403232307.i2NN72dK003366@linux.midominio.com>
To: jabernal@linux.midominio.com
Subject: sd
Status: RO
sd
.
```

[19] ¡Recuerdas aquellos tiempos en los que no había ADSL!.
[20] No le contéis a nadie que estáis hablando SMTP y POP. Puede que os tomen por locos.

```
quit
+OK Sayonara
Connection closed by foreign host.
```

¿Qué hemos hecho?

- Pues nos hemos conectado al puerto 110 de nuestro servidor, en donde está escuchando nuestro servidor POP.
- Nos identificamos poniendo nuestro nombre de usuario y nuestra password[21].
- El comando `list` nos muestra los mensajes que están en nuestro buzón en el servidor, con su número de orden y su tamaño. Si quisiéramos recuperar alguno, con el comando ***retr*** y el número del mensaje, podremos descargárnoslo.

Protocolo IMAP

El protocolo IMAP (*Internet Messaging Access Protocol*[22]) es más potente que POP en la mayoría de los casos. En el modo desconectado (*offline*) sus capacidades son similares, pero es en el modo conectado (*online*) donde IMAP lo supera con creces. IMAP permite la manipulación de buzones en el servidor remoto como si fueran locales.

En conexiones de poco ancho de banda, permite capturar la estructura del mensaje sin descargarlo[23] y seleccionar qué parte del mensaje nos interesa descargarnos.

Posee adicionalmente la capacidad de manipular un mensaje en el buzón remoto, permitiendo marcar los mensajes como leídos, borrados, contestados. La tendencia es a utilizar servidores con este protocolo en vez de POP. Pero claro está, esto depende de que nuestro proveedor del servicio de correo o administrador del sistema nos ofrezca esta posibilidad.

Si utilizamos un sistema de webmail[24] y deseamos poder crear carpetas, éste es el protocolo necesario.

El servidor Cyrus es un potente sistema que soporta IMAP, POP y sus equivalentes seguros, IMAPS y POPS. Es complejo, pero quien quiera probarlo, puede encontrar un excelente tutorial en `http://www.linuxsilo.net/articles/postfix.html`

18.2. Agentes de Transporte

18.2.1. Postfix

¿Qué es Postfix[25]? Es un servidor de correo (MTA) que inició su existencia en 1998 como una alternativa de *Wietse Venema* al ampliamente usado Sendmail. Inicialmente se distribuyó bajo el nombre de *IBM Secure Mailer*, pasando posteriormente a la denominación de Postfix. En su diseño han primado factores como la seguridad, la eficiencia y la facilidad de configuración y administración, junto con la compatibilidad con Sendmail y con otros sistemas de correo. El exterior está "sendmailizado" pero su interior es totalmente diferente.

Siendo el correo electrónico hoy día una herramienta de trabajo vital en multitud de entornos de trabajo, plantearse sustituir el sistema de correo actual por otro nuevo es una decisión muy delicada. Se debe garantizar que la migración se va a producir sin inconvenientes para los usuarios y con el mínimo tiempo de parada del servicio. Algunas de las virtudes de Postfix que pueden decidir su uso son:

[21] Como ya sabéis, esta información no cifrada viajando por la red es un peligro. Afortunadamente, el servidor POP ofrece un servicio cifrado POP3S en el puerto 995.
[22] Protocolo de Acceso a Mensajería de Internet.
[23] Cosa que no puede hacer POP.
[24] En esta entrega veremos squirrelmail.
[25] Más información en `http://www.postfix.org/`

- Diseño modular, no es un único programa monolítico.
- La seguridad y el rendimiento han sido condicionantes desde el comienzo de su diseño.
- Soporte para las tecnologías más usadas hoy día: LDAP, Bases de datos (MySQL), autentificación mediante SASL, LMTP, etc.
- Soporte para dominios virtuales.
- Facilidad de configuración.
- Compatibilidad hacia/desde fuera con Sendmail.
- Abundante documentación, y de calidad.
- Fácil integración con antivirus.
- Tiene múltiples formas de obtener información de "lo que está pasando" para resolver problemas o simplemente, para aprender.
- Se pueden lanzar varias instancias de Postfix en la misma máquina con distintas configuraciones, usando cada una distintas direcciones IP, distintos puertos, etc.
- Filtrado de cabeceras y cuerpos de mensajes por expresiones regulares.
- Utilidades para la gestión del correo, entre otras para la gestión de las colas de mensajes.

Por último, pero no menos importante, hay que decir que el código fuente de Postfix (por supuesto de dominio público) es un ejemplo de diseño, claridad y documentación, lo cual facilita su mantenimiento (por su autor o, en el futuro, por otros) así como la incorporación de nuevas capacidades, corrección de errores, etc.

En el website de Postfix, `http://www.postfix.org/`, pueden encontrarse enlaces a documentación que profundiza en sus características y en otro aspectos.

Arquitectura de Postfix

Al contrario que Sendmail, que se basaba en una estructura monolítica[26], el diseño de Postfix se basa en la división en distintos procesos del tratamiento que se realiza del correo a través del MTA. Estos procesos se comunican entre sí a través de sockets, siendo la información transmitida la mínima posible. El conjunto de todos estos procesos constituye Postfix.

Una gran contribución a la estabilidad y velocidad del servidor Postfix es la forma inteligente en que su creador implementó las colas de correo. Postfix utiliza varias colas diferentes, cada una manejada de forma diferente:

- *Maildrop queue*. El correo que es entregado localmente en el sistema es aceptado por la cola *Maildrop*. El correo se chequea para formatearlo apropiadamente antes de ser entregado a la cola *Incoming*.
- *Incoming queue*. Esta cola recibe correo de otros *hosts*, clientes o de la cola *Maildrop*. Mientras sigue llegando correo y Postfix no puede manejarlo, en esta cola se quedan los correos.
- *Active queue*. Es la cola utilizada para entregar los mensajes. La *Active queue* tiene un tamaño limitado, y los mensajes solamente serán aceptados si hay espacio en ella. Esto quiere decir que las colas *Incoming* y *Deferred* tienen que esperar hasta que la cola *Active* pueda aceptar más mensajes.

[26] Un único programa que lo hace todo.

Instalación de Postfix

Guadalinex

```
#apt-get install postfix
```

El resultado de la ejecución se muestra a continuación:

```
root@guadalinex:/home/mowgli# apt-get install postfix
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Paquetes sugeridos:
procmail postfix-mysql postfix-pgsql postfix-ldap postfix-pcre
Paquetes recomendados
resolvconf
Se áinstalarn los siguientes paquetes NUEVOS:
postfix
0 actualizados, 1 se áinstalarn, 0 para eliminar y 575 no actualizados.
Necesito descargar 801kB de archivos.
Se áutilizarn 1970kB de espacio de disco adicional édespus de desempaquetar.
```

En este momento se inicia de manera automática la configuración de postfix:

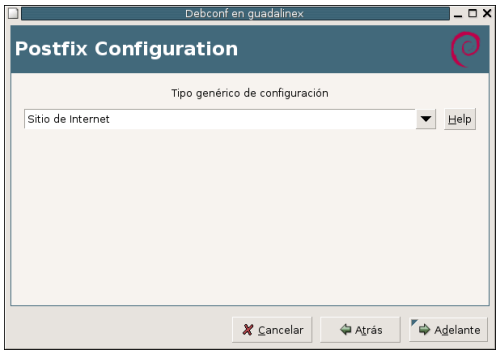


Figura 18.1: Instalacion Postfix - Tipo genérico de configuración

Como tipo genérico de configuración dejamos la opción por defecto, **Sitio de Internet**, y pulsamos [**Adelante**], el asistente de configuración nos muestra la siguiente pantalla[27]:

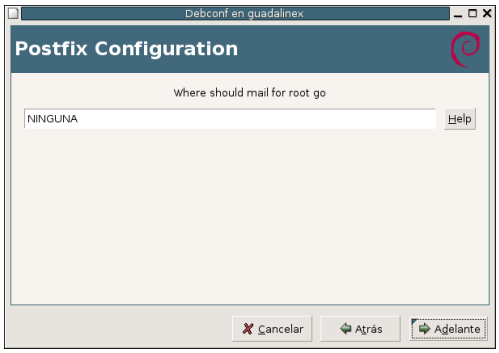


Figura 18.2: Instalacion Postfix - Redirección correo root

[27]Para ver los distintos tipos de instalación y una descripción de los mismos podemos visualizar la pantalla de ayuda a través del botón [**Help**]

Si queremos que el correo dirigido a `root` o a cualquier usuario con `uid 0` se redirija a algún alias, escribiremos uno en la pantalla anterior. Este alias se añade al fichero `/etc/aliases`. En el caso de no querer que se redireccione a ningún alias, se deja la opción por defecto y el correo se redireccionará a `/var/mail/nobody`. En nuestro caso dejaremos la opción por defecto y pulsaremos [**Adelante**], mostrándose la siguiente pantalla:

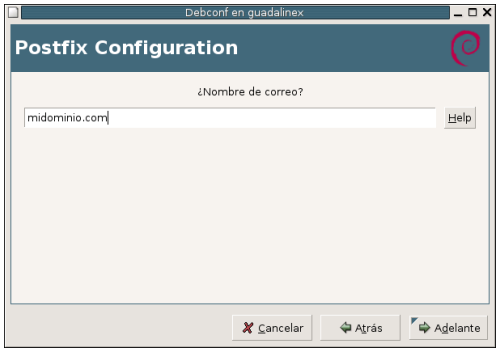


Figura 18.3: Instalación Postfix - Dominio de correo

El "nombre de correo" es la porción del nombre de máquina de la dirección que será mostrada en las noticias y correos salientes (después del nombre de usuario y el signo @). Este nombre será usado por otros programas además de Postfix; deberá ser el único nombre de dominio completo (FDQN) desde el que parecerá originarse el correo. Para nuestro caso utilizaremos `midominio.com`, y pulsaremos [**Adelante**], se muestra la siguiente pantalla del asistente de configuración:

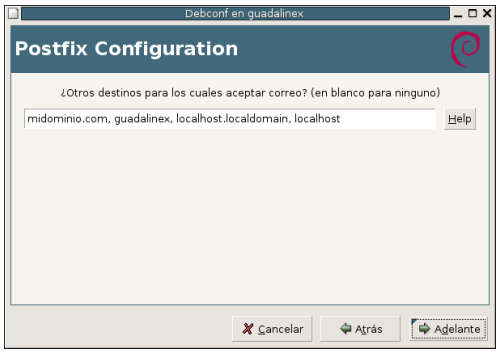


Figura 18.4: Instalación Postfix - Destinos para los que se acepta correo

Deberemos insertar una lista separada por comas, de dominios de los que esta máquina deberá considerarse destino final, en nuestro caso dejaremos los que muestra por defecto, que no son más que el dominio que hemos configurado y el correo local, pulsamos [**Adelante**] y se muestra la siguiente pantalla del asistente:

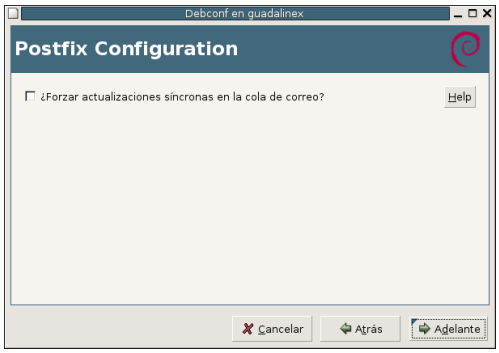


Figura 18.5: Instalación Postfix - Actualización síncrona cola correo

Si se fuerzan las actualizaciones síncronas, el correo será procesado más lentamente. Si no se fuerzan, existe la posibilidad remota de perder algunos correos si el sistema se colapsa en un momento inoportuno y no está usando un sistema de ficheros transaccional (como `ext3`). En nuestro caso el sistema de ficheros es `ext3`, luego dejamos la configuración por defecto `off`.

Pulsamos [**Adelante**] en el resto de opciones, aceptando los valores por defecto, hasta que concluye el asistente de configuración. Continúa la instalación de los paquetes.

Como podemos observar, Postfix queda configurado e iniciado una vez que concluye la instalación[28]. Comprobaremos que la instalación se ha realizado correctamente enviando un correo a través de nuestro servidor (en negrita y a mayor tamaño mostramos lo que vamos tecleando:

```
root@guadalinex:/home/mowgli# telnet localhost 25
Trying ::1...
Connected to localhost.
Escape character is '^]'.
220 guadalinex ESMTP Postfix (Debian/GNU)
|EHLO localhost|
250-guadalinex
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250 8BITMIME
|MAIL FROM: <mowgli@midominio.com>|
250 Ok
|RCPT TO: <curso@midominio.com>|
250 Ok
|DATA|
354 End data with <CR><LF>.<CR><LF>
|SUBJECT: Prueba de correo
FROM: Mowgli <mowgli@midominio.com>
TO: Curso <curso@midominio.com>
Prueba de correo|
.
250 Ok: queued as EAE8E3B7E7
|QUIT|
221 Bye
Connection closed by foreign host.
```

Si todo ha ido correctamente, podemos conectarnos al buzón del usuario curso, y comprobar que ha recibido el correo.

[28] En el caso de que ya tuviésemos instalado Postfix o quisieramos volver a realizar la configuración básica, podemos ejecutar el script de postinstalación de este paquete con el siguiente comando: `dpkg-reconfigure postfix`

```
root@guadalinex:/home/mowgli# more /var/mail/curso
From mowgli@midominio.com Sun Jan 23 12:22:07 2005
Return-Path: <mowgli@midominio.com>
X-Original-To: curso@midominio.com
Delivered-To: curso@midominio.com
Received: from localhost (localhost [IPv6:::1])
by guadalinex (Postfix) with ESMTP id EAE8E3B7E7
for <curso@midominio.com>; Sun, 23 Jan 2005 12:20:53 +0100 (CET)
SUBJECT: Prueba de correo
From: Mowgli <mowgli@midominio.com>
To: Curso <curso@midominio.com>
Message-Id: <20050123112053.EAE8E3B7E7@guadalinex>
Date: Sun, 23 Jan 2005 12:20:53 +0100 (CET)
Prueba de correo
```

Personalizar la configuración

La configuración de Postfix se realiza mediante dos ficheros principales (situados en el directorio `/etc/postfix`) y varias tablas opcionales que puede crear el administrador de correo. Los ficheros de configuración son:

`master.cf` Aquí se configuran los procesos que pueden arrancarse y algunos parámetros como el número de cada uno que puede haber simultáneamente, etc. Normalmente sólo hay que tocarlo si queremos usar un sistema alternativo de entrega de correo local (si usamos Cyrus, Courier, por ejemplo), si queremos integrar un antivirus y cosas así.

`main.cf` Todos los parámetros relacionados con la función que debe realizar Postfix los definimos aquí. Será el fichero sobre el que se realicen las modificaciones más habituales.

En el fichero `main.cf` podemos asignar valores a dos cosas:

- Parámetros. Les asignamos valores como nombres de hosts, direcciones IP, número de bytes en el caso de algunos límites, etc.
- Clases de restricciones. Les asignamos una serie de "restricciones", que definen fundamentalmente de/para quién vamos a aceptar correo.

A las clases de restricciones (que en realidad también son parámetros) se le asignan una serie de valores que definen, fundamentalmente, de/para quién se aceptará correo. En Postfix hay un gran número de parámetros, pero la mayoría define situaciones fuera de lo común o aspectos relacionados con la administración avanzada (afinar el rendimiento, límites, códigos SMTP a devolver a los clientes ante determinadas circunstancias, etc). En realidad lo normal es que haya que tocar poco más de media docena de ellos, aparte de las tablas necesarias para alias, dominios virtuales, etc.

En cuanto a las tablas de configuración pueden estar en una gran variedad de formatos, en dos variantes: De acceso por clave (Berkeley DB, MySQL, etc.) o De acceso secuencial (expresiones regulares).

Tal como se realizó la configuración por defecto se generaron los ficheros de configuración. En el caso de `main.cf` se obtiene el siguiente fichero[29]:

```
# See /usr/share/postfix/main.cf.dist for a commented, more complete version

smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no
```

[29] Además, añadir que tal cual aparece en ese fichero, en **`/usr/share/postfix/main.cf.dist`** hay una versión completa y comentada del mismo.

```
# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

myhostname = guada04
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
mydestination = midominio.com, guadalinex, localhost.localdomain, localhost
relayhost =
mynetworks = 127.0.0.0/8
mailbox_command =
mailbox_size_limit = 0
recipient_delimiter = +
myorigin = /etc/mailname
```

Listado 18.1: /conf/main.cf

Algunos de los parámetros más interesantes en la configuración de Postfix en el fichero `main.cf`:

`queue_directory` Especifica la localización de la cola de Postfix.

`daemon_directory` Especifica la localización de todos los demonios de Postfix

`myhostname` Especifica el nombre de host en internet para este sistema de correo. Por defecto se utiliza el FQDM obtenido con `gethostname()`. Este parámetro se utiliza posteriormente dentro de otros parámetros.

`mydomain` Especifica el nombre de dominio local. Por defecto se utiliza el valor de `$myhostname` menos el primer componente. Este parámetro se utiliza posteriormente dentro de otros parámetros.

`myorigin` Especifica el dominio que aparece en los correos locales como origen. Por defecto se utiliza `$myhostname`, aunque en el caso de utilizar un dominio con múltiples máquinas debe usarse `$mydomain`.

`mydestination` Especifica la lista de dominios para los que esta máquina se considera destino final.

`mynetworks_style` Especifica la lista de clientes SMTP en los que se confia y a los que se les permite el envío de correo a través de Postfix. Mediante el uso de los valores `subnet`, `class` y `host` se permite el acceso a los clientes pertenecientes a la subred, clase o únicamente al host local respectivamente.

`mynetworks` Especifica el direccionamiento para el cual se permite el reenvío de correo a través de Postfix. En caso de definir este parámetro se ignora lo definido en `$mynetworks_style`.

`relay_domains` Especifica los dominios para los que está permitido el reenvío de correo.

`relayhost` Especifica el host hacia el que se envía el correo y que hace de reenviador en caso de que no esté directamente conectado a internet.

`alias_maps` Especifica la lista de base de datos de alias para el reenviador de correo local.

`alias_database` Especifica la base de datos de alias que se genera cada vez que se ejecuta `newaliases`.

`home_mailbox` Especifica la ruta, relativa al directorio `$HOME`, del fichero con el buzón de correo. Por defecto es `/var/spool/mail/user` o `/var/mail/user`. Si se especifica `Maildir/` se utilizará el formato utilizado por qmail.

`mail_spool_directory` Especifica la ruta donde se almacenan los buzones con formato Unix mailbox.

`header_checks` Especifica una tabla de patrones con la que se compara las cabeceras de los mensajes.

El otro fichero que utiliza Postfix en su configuración es `master.cf`. Define el comportamiento del programa master. Dicho programa forma parte de Postfix y se ejecuta de forma continua en el servidor de correo. Recibe indicaciones de unos procesos e inicia otros. Es el "director de orquesta" de Postfix. Mediante `master.cf` se define la forma correcta en que se debe llamar a cada uno de los procesos. Cada entrada en el fichero es un conjunto de ocho campos separados por blancos o tabuladores:

- *Service*. Nombre del servicio que se está configurando.
- *Type*. Tipo de comunicación de transporte utilizado por el servicio.
- *Private*. Restricciones de seguridad a procesos externos.
- *Unprivileged*. Ejecución en modo no privilegiado.
- *Chroot*. Indica si el servicio se ejecuta en un directorio de acceso restringido.
- *Wakeup*. Segundos que deben transcurrir para que el proceso master despierte el servicio.
- *Maxprocess*. Número máximo de procesos que puede usar el servicio.
- *Command*. Nombre del programa a ejecutar y parámetros a pasar.

Este fichero no es necesario modificarlo con la instalación por defecto:

```
#
# Postfix master process configuration file.  Each logical line
# describes how a Postfix daemon program should be run.
#
# A logical line starts with non-whitespace, non-comment text.
# Empty lines and whitespace-only lines are ignored, as are comment
# lines whose first non-whitespace character is a '#'.
# A line that starts with whitespace continues a logical line.
#
# The fields that make up each line are described below. A "-" field
# value requests that a default value be used for that field.
#
# Service: any name that is valid for the specified transport type
# (the next field).  With INET transports, a service is specified as
# host:port.  The host part (and colon) may be omitted.  Either host
# or port may be given in symbolic form or in numeric form.  Examples
# for the SMTP server:  localhost:smtp receives mail via the loopback
# interface only; 10025 receives mail on port 10025.
#
# Transport type: "inet" for Internet sockets, "unix" for UNIX-domain
# sockets, "fifo" for named pipes.
#
# Private: whether or not access is restricted to the mail system.
# Default is private service.  Internet (inet) sockets can't be private.
#
# Unprivileged: whether the service runs with root privileges or as
# the owner of the Postfix system (the owner name is controlled by the
# mail_owner configuration variable in the main.cf file).  Only the
# pipe, virtual and local delivery daemons require privileges.
```

```
#
# Chroot: whether or not the service runs chrooted to the mail queue
# directory (pathname is controlled by the queue_directory configuration
# variable in the main.cf file). Presently, all Postfix daemons can run
# chrooted, except for the pipe, virtual and local delivery daemons.
# The proxymap server can run chrooted, but doing so defeats most of
# the purpose of having that service in the first place.
# The files in the examples/chroot-setup subdirectory describe how
# to set up a Postfix chroot environment for your type of machine.
#
# Wakeup time: automatically wake up the named service after the
# specified number of seconds. A ? at the end of the wakeup time
# field requests that wake up events be sent only to services that
# are actually being used.  Specify 0 for no wakeup. Presently, only
# the pickup, queue manager and flush daemons need a wakeup timer.
#
# Max procs: the maximum number of processes that may execute this
# service simultaneously. Default is to use a globally configurable
# limit (the default_process_limit configuration parameter in main.cf).
# Specify 0 for no process count limit.
#
# Command + args: the command to be executed. The command name is
# relative to the Postfix program directory (pathname is controlled by
# the daemon_directory configuration variable). Adding one or more
# -v options turns on verbose logging for that service; adding a -D
# option enables symbolic debugging (see the debugger_command variable
# in the main.cf configuration file). See individual command man pages
# for specific command-line options, if any.
#
# General main.cf options can be overridden for specific services.
# To override one or more main.cf options, specify them as arguments
# below, preceding each option by "-o".  There must be no whitespace
# in the option itself (separate multiple values for an option by
# commas).
#
# In order to use the "uucp" message tranport below, set up entries
# in the transport table.
#
# In order to use the "cyrus" message transport below, configure it
# in main.cf as the mailbox_transport.
#
# SPECIFY ONLY PROGRAMS THAT ARE WRITTEN TO RUN AS POSTFIX DAEMONS.
# ALL DAEMONS SPECIFIED HERE MUST SPEAK A POSTFIX-INTERNAL PROTOCOL.
#
# DO NOT SHARE THE POSTFIX QUEUE BETWEEN MULTIPLE POSTFIX INSTANCES.
#
# ==========================================================================
# service type  private unpriv  chroot  wakeup  maxproc command + args
#               (yes)   (yes)   (yes)   (never) (100)
# ==========================================================================
127.0.0.1:smtp inet n       -       -       -       -       smtpd
::1:smtp       inet n       -       -       -       -       smtpd
#submission inet n       -       -       -       -       smtpd
#       -o smtpd_etrn_restrictions=reject
#628        inet  n       -       -       -       -       qmqpd
pickup     fifo  n       -       -       60      1       pickup
cleanup    unix  n       -       -       -       0       cleanup
qmgr       fifo  n       -       -       300     1       qmgr
```

```
#qmgr     fifo   n       -       -       300     1       oqmgr
rewrite   unix   -       -       -       -       -       trivial-rewrite
bounce    unix   -       -       -       -       0       bounce
defer     unix   -       -       -       -       0       bounce
trace     unix   -       -       -       -       0       bounce
verify    unix   -       -       -       -       1       verify
flush     unix   n       -       -       1000?   0       flush
proxymap  unix   -       -       n       -       -       proxymap
smtp      unix   -       -       -       -       -       smtp
relay     unix   -       -       -       -       -       smtp
#         -o smtp_helo_timeout=5 -o smtp_connect_timeout=5
showq     unix   n       -       -       -       -       showq
error     unix   -       -       -       -       -       error
local     unix   -       n       n       -       -       local
virtual   unix   -       n       n       -       -       virtual
lmtp      unix   -       -       n       -       -       lmtp
anvil     unix   -       -       n       -       1       anvil
#
# Interfaces to non-Postfix software. Be sure to examine the manual
# pages of the non-Postfix software to find out what options it wants.
#
# maildrop. See the Postfix MAILDROP_README file for details.
#
maildrop  unix   -       n       n       -       -       pipe
  flags=DRhu user=vmail argv=/usr/local/bin/maildrop -d ${recipient}
uucp      unix   -       n       n       -       -       pipe
  flags=Fqhu user=uucp argv=uux -r -n -z -a$sender - $nexthop!rmail (
    $recipient)
ifmail    unix   -       n       n       -       -       pipe
  flags=F user=ftn argv=/usr/lib/ifmail/ifmail -r $nexthop ($recipient)
bsmtp     unix   -       n       n       -       -       pipe
  flags=Fq. user=bsmtp argv=/usr/lib/bsmtp/bsmtp -d -t$nexthop -f$sender
    $recipient
scalemail-backend unix -       n       n       -       2       pipe
  flags=R user=scalemail argv=/usr/lib/scalemail/bin/scalemail-store ${
    nexthop} ${user} ${extension}

# only used by postfix-tls
#tlsmgr    fifo   -       -       n       300     1       tlsmgr
#smtps     inet   n       -       n       -       -       smtpd -o
    smtpd_tls_wrappermode=yes -o smtpd_sasl_auth_enable=yes
#587       inet   n       -       n       -       -       smtpd -o
    smtpd_enforce_tls=yes -o smtpd_sasl_auth_enable=yes
```

Listado 18.2: /conf/master.cf

Arranque de Postfix

Una vez instalado, se procede a arrancar el demonio, si no lo está ya. Para ello se utiliza el script creado en `/etc/init.d`. Con este script se comprueba también la correcta definición en el fichero de configuración `main.cf`

```
#/etc/init.d/postfix check
```

Una vez que se ha comprobado que no hay errores, es necesario indicar a Postfix que cargue de nuevo la configuración:

```
#/etc/init.d/postfix reload
```

Para obtener más información referente a posibles errores o el estado de ejecución de Postfix se recurrirá a los ficheros de log:

```
/var/log/mail.err
/var/log/mail.info
/var/log/mail.log
/var/log/mail.warn
```

Dependiendo del nivel de criticidad del aviso se almacenará en uno u otro fichero.

18.2.2. Sendmail

> El Rey Gordio de Frigia hizo un nudo tan fuerte que nadie pudo deshacerlo. El Nudo Gordiano permaneció así, o al menos eso dice la historia, hasta que llegó Alejandro Magno y utilizó una forma diferente de deshacer nudos. Sería interesante si el nudo que es sendmail pudiera ser desatado con un rápido golpe de una nueva visión, pero !ay¡, no es posible. En vez de ello, se debe coger un enfoque más mundano, así que en este libro lo desataremos de la manera difícil, hebra a hebra. *Sendmail* Bryan Costales y Eric Allman.

Estas palabras describen a la perfección lo que ha sido Sendmail durante mucho tiempo, una de las bestias negras de los administradores de sistemas Unix. Una relación amor/odio se entablaba con esta gran obra de ingeniería. Por una parte, su potencia y capacidades eran insustituibles y por otra, su complejidad de configuración y sus errores de seguridad le hacían temible. Por ello, Postfix está ganando poco a poco cuota de poder en el dominio de los agentes de transporte de correo en el "mundo libre".

Por fortuna, esa situación ha cambiado un poco. La inclusión del preprocesador de macros *m4* para la configuración y sus reescrituras para mejorar el diseño y la seguridad, han mejorado un poco la situación.

Sendmail fue escrito por Eric Allman en la Universidad de California en Berkeley para el Unix de BSD. Ha sido portado a todas las plataformas existentes y todas las distribuciones de Linux la incorporan. Vamos a hablar sobre él por motivos históricos y porque aún se utiliza mucho, aunque postfix va imponiéndose.

Instalación de Sendmail

Para instalar los paquetes que necesitamos para el correo utilizaremos com de costumbre, `apt-get`.

Si vamos a utilizar sendmail, es importante utilizar una versión reciente, ya que son muchas las mejoras de seguridad que incorporan. Por ejemplo, el que vamos a instalar de prueba, será sendmail-8.12.10. Con la utilidad apt-get, obtendremos una versión actualizada.

Los paquetes que instalaremos serán[30] *sendmail*, el agente de transporte, *sendmail-cf*, las utilidades para la configuración y *dovecot*, es el paquete en que se encuentran los servidores de POP e IMAP.

```
[root@linux entrega04-3]# apt-get install sendmail sendmail-cf dovecot
Leyendo listas de paquetes... Done
Construyendo Árbol de dependencias... Done
Se instalarán los paquetes NUEVOS siguientes:
imap sendmail sendmail-cf
0 upgraded, 3 newly installed, 0 removed and 51 not upgraded.
Need to get 2600kB of archives.
After unpacking 5926kB of additional disk space will be used.
```

[30] En Debian:
`#apt-get install sendmail dovecot`

```
Get:1 http://ayo.freshrpms.net fedora/linux/1/i386/core dove-
cot 1:2002d-3 [1287kB]
Get:2 http://ayo.freshrpms.net fedora/linux/1/i386/core send-
mail 8.12.10-1.1.1 [1018kB]
Get:3 http://ayo.freshrpms.net fedora/linux/1/i386/core sendmail-
cf 8.12.10-1.1.1 [294kB]
Fetched 2600kB in 1m41s (25,5kB/s)
Committing changes...
Preparing... ########################################### [100%]
1:sendmail-cf ########################################### [ 33%]
2:dovecot ############################################### [ 67%]
3:sendmail ############################################## [100%]
Done.
```

Una vez instalados, mediante la utilidad[31] `setup`, podremos configurarlos para que arranquen automáticamente. Los servicios que activaremos serán[32]:

sendmail Demonio del Agente de Transporte. Utiliza el puerto 25.

imap Servidor para acceder a los buzones de usuario utilizando el protocolo IMAP. Utiliza el puerto 143.

imaps Igual que imap pero con un protocolo cifrado. Utiliza el puerto 993.

ipop3 Servidor del protocolo POP[33]. Utiliza el puerto 110.

ipop3s Servidor POP seguro. Utiliza el puerto 995.

Configuración de Sendmail

Pasemos a configurar sendmail. Miremos el fichero `sendmail.cf`, que se encuentra en el directorio `/etc/mail`[34].

```
# strip group: syntax (not inside angle brackets!) and trailing semico-
lon
R$* $: $1 <@> mark addresses
R$* < $* > $* <@> $: $1 < $2 > $3 unmark <addr>
R@ $* <@> $: @ $1 unmark @host:...
R$* [ IPv6 : $+ ] <@> $: $1 [ IPv6 : $2 ] unmark IPv6 addr
R$* :: $* <@> $: $1 :: $2 unmark node::addr
R:include: $* <@> $: :include: $1 unmark :include:...
R$* : $* [ $* ] $: $1 : $2 [ $3 ] <@> remark if leading colon
R$* : $* <@> $: $2 strip colon if marked
R$* <@> $: $1 unmark
R$* ; $1 strip trailing semi
R$* < $+ :; > $* $@ $2 :; <@> catch <list:;>
R$* < $* ; > $1 < $2 > bogus bracketed semi
```

Como véis, quien sea capaz de entender esto, no debe ser una persona normal. Es una de las razones de la mala fama (y en parte merecida) de sendmail.

[31] Si bien se ejecuta de forma automática

```
# update-rc.d sendmail defaults
```

para Debian.

[32] Al menos sendmail y un protocolo de acceso a los buzones.

[33] El protocolo POP en su versión 2 (ipop2) no se utiliza normalmente.

[34] O en `/etc`

Pero gracias a la utilización del preprocesador de macros `m4`, la tarea se nos ha vuelto más fácil. Bueno, aún así nos llevará un poco comprenderla totalmente. Nuestro fichero de configuración será `/etc/mail/sendmail.mc`[35] y a partir de él obtendremos el fichero `sendmail.cf`, que es el que leerá sendmail.

En el directorio `/usr/share/sendmail-cf/cf` existen ejemplos de ficheros `.mc` para múltiples sistemas. Escogeremos el correspondiente a nuestro sistema.

Tenemos un punto a nuestro favor. La configuración por defecto nos servirá casi sin modificaciones en un tanto por ciento muy elevado de casos. Cuando lo tengamos a nuestro gusto, simplemente tecleamos `#make` en el directorio `/etc/mail` y se generará automáticamente el fichero `sendmail.cf`[36]. Comentaremos las líneas más interesantes.

```
[root@linux mail]# more sendmail.mc
divert(-1)dnl
dnl #
dnl # This is the sendmail macro config file for m4. If you make chan-
ges to
dnl # /etc/mail/sendmail.mc, you will need to regenerate the
dnl # /etc/mail/sendmail.cf file by confirming that the sendmail-
cf package is
dnl # installed and then performing a
dnl #
dnl # make -C /etc/mail
dnl #
```

Las líneas `divert` y `dnl` son comentarios.

```
include(‘/usr/share/sendmail-cf/m4/cf.m4’)dnl
```

Carga el fichero `cf.m4` que necesita.

```
VERSIONID(‘setup for Red Hat Linux’)dnl
OSTYPE(‘linux’)dnl
```

Decimos la versión y el sistema operativo. Le servirá para adoptar opciones personalizadas. En este caso, cargará el fichero `/usr/share/sendmail-cf/ostype/linux.m4`.

```
dnl #
dnl # Uncomment and edit the following line if your out-
going mail needs to
dnl # be sent out through an external mail server:
dnl #
dnl define(‘SMART_HOST’,‘smtp.miproveedor.com’)
```

Opción que está comentada. El `Smart_host` es un agente de transporte al que le paso la pelota y que él se encargue de los siguientes pasos. Muy útil si estamos en una red privada y solamente ese "host inteligente" puede salir hacia el exterior. Para utilizarla, tendríamos que poner cuál es ese host en nuestro caso y descomentarla quitando el `dnl`.

[35] En un sistema Fedora o RedHat

[36] También podemos ejecutar a mano:
`#m4 ${CFDIR}/m4/cf.m4 fichero.mc >fichero.cf`

- `m4` es el procesador de macros
- necesita del fichero `cf.m4`
- actúa sobre el fichero `.mc`
- genera el fichero `.cf`

```
dnl #
define(‘confDEF_USER_ID’,"8:12")dnl
```

Usuario y grupo que ejecutarán el proceso sendmail (normalmente usuario mail y grupo mail).

```
dnl define(‘confAUTO_REBUILD’)dnl
define(‘confTO_CONNECT’, ‘1m’)dnl
define(‘confTRY_NULL_MX_LIST’,true)dnl
define(‘confDONT_PROBE_INTERFACES’,true)dnl
define(‘PROCMAIL_MAILER_PATH’,‘/usr/bin/procmail’)dnl
define(‘ALIAS_FILE’, ‘/etc/aliases’)dnl
```

Cuál es el fichero de alias. Crea una dirección de correo virtual y la asocia a otra dirección. Por ejemplo la línea `webmaster: admin` dice que todos los correos que vayan a la dirección webmaster@dominio-configurado.com, siendo `dominio-configurado.com` el que está recogiendo nuestro sendmail, vayan a la dirección admin@dominio-configurado.com

```
dnl define(‘STATUS_FILE’, ‘/etc/mail/statistics’)dnl
define(‘UUCP_MAILER_MAX’, ‘2000000’)dnl
define(‘confUSERDB_SPEC’, ‘/etc/mail/userdb.db’)dnl
define(‘confPRIVACY_FLAGS’, ‘authwar-
nings,novrfy,noexpn,restrictqrun’)dnl
define(‘confAUTH_OPTIONS’, ‘A’)dnl
dnl #
dnl # The following allows relaying if the user authenticates, and di-
sallows
dnl # plaintext authentication (PLAIN/LOGIN) on non-TLS links
dnl #
dnl define(‘confAUTH_OPTIONS’, ‘A p’)dnl
dnl #
dnl # PLAIN is the preferred plaintext authentication met-
hod and used by
dnl # Mozilla Mail and Evolution, though Outlook Express and ot-
her MUAs do
dnl # use LOGIN. Other mechanisms should be used if the connec-
tion is not
dnl # guaranteed secure.
dnl #
dnl TRUST_AUTH_MECH(‘EXTERNAL DIGEST-MD5 CRAM-MD5 LOGIN PLAIN’)dnl
dnl define(‘confAUTH_MECHANISMS’, ‘EXTERNAL GSSAPI DIGEST-MD5 CRAM-
MD5 LOGIN PLAIN’)dnl
dnl # Rudimentary information on creating certificates for send-
mail TLS:
dnl # make -C /usr/share/ssl/certs usage
dnl #
dnl define(‘confCACERT_PATH’,‘/usr/share/ssl/certs’)
dnl define(‘confCACERT’,‘/usr/share/ssl/certs/ca-bundle.crt’)
dnl define(‘confSERVER_CERT’,‘/usr/share/ssl/certs/sendmail.pem’)
dnl define(‘confSERVER_KEY’,‘/usr/share/ssl/certs/sendmail.pem’)
dnl #
dnl # This allows sendmail to use a keyfile that is shared with OpenL-
DAP’s
dnl # slapd, which requires the file to be readble by group ldap
dnl #
dnl define(‘confDONT_BLAME_SENDMAIL’,‘groupreadablekeyfile’)dnl
```

```
dnl #
dnl define(‘confTO_QUEUEWARN’, ‘4h’)dnl
```

Los mensajes que recoge sendmail los pone en una cola (un directorio en donde los va guardando) y los envía cuando puede. Por ejemplo, si nuestra conexión a Internet no es permanente o el agente de transporte destino no está operativo. Este parámetro designa el tiempo (4 horas) que al cumplirse, nos envía un mensaje indicando que no lo ha podido entregar.

```
dnl define(‘confTO_QUEUERETURN’, ‘5d’)dnl
```

Si en 5 días no ha conseguido entregarlo al destinatario, nos lo devuelve.

```
dnl define(‘confQUEUE_LA’, ‘12’)dnl
dnl define(‘confREFUSE_LA’, ‘18’)dnl
define(‘confTO_IDENT’, ‘0’)dnl
dnl FEATURE(delay_checks)dnl
FEATURE(‘no_default_msa’,‘dnl’)dnl
FEATURE(‘smrsh’,‘/usr/sbin/smrsh’)dnl
FEATURE(‘mailertable’,‘hash -o /etc/mail/mailertable.db’)dnl
FEATURE(‘virtusertable’,‘hash -o /etc/mail/virtusertable.db’)dnl
FEATURE(redirect)dnl
FEATURE(always_add_domain)dnl
```

Siempre añade el dominio para completar las direcciones de correo electrónico. Por ejemplo, estando en el dominio midominio.com, un correo enviado al usuario linux, se completará como linux@midominio.com.

```
FEATURE(use_cw_file)dnl
FEATURE(use_ct_file)dnl
dnl #
dnl # The -
t option will retry delivery if e.g. the user runs over his quota.
dnl #
FEATURE(local_procmail,‘’,‘procmail -t -Y -a $h -d $u’)dnl
FEATURE(‘access_db’,‘hash -T<TMPF> -o /etc/mail/access.db’)dnl
FEATURE(‘blacklist_recipients’)dnl
EXPOSED_USER(‘root’)dnl
dnl #
dnl # The following causes sendmail to only listen on the IPv4 loop-
back address
dnl # 127.0.0.1 and not on any other network devices. Remove the loop-
back
dnl # address restriction to accept email from the internet or intra-
net.
dnl #
DAEMON_OPTIONS(‘Port=smtp,Addr=127.0.0.1, Name=MTA’)dnl
```

Muy importante. Debemos poner nuestra dirección de red local en vez de 127.0.0.1 si queremos que los clientes puedan comunicar con el servidor.

```
dnl #
dnl # The following causes sendmail to additionally lis-
ten to port 587 for
dnl # mail from MUAs that authenticate. Roa-
ming users who can’t reach their
```

```
dnl # preferred sendmail daemon due to port 25 being blocked or redi-
rected find
dnl # this useful.
dnl DAEMON_OPTIONS(‘Port=submission, Name=MSA, M=Ea’)dnl
dnl #
dnl # The following causes sendmail to additionally lis-
ten to port 465, but
dnl # starting immediately in TLS mode upon connec-
ting. Port 25 or 587 followed
dnl # by STARTTLS is preferred, but roaming clients using Outlook Ex-
press can’t
dnl # do STARTTLS on ports other than 25. Mozi-
lla Mail can ONLY use STARTTLS
dnl # and doesn’t support the deprecated smtps; Evolu-
tion <1.1.1 uses smtps
dnl # when SSL is enabled--
STARTTLS support is available in version 1.1.1.
dnl #
dnl # For this to work your OpenSSL certificates must be configured.
dnl #
dnl DAEMON_OPTIONS(‘Port=smtps, Name=TLSMTA, M=s’)dnl
dnl #
dnl # The following causes sendmail to additionally lis-
ten on the IPv6 loopback
dnl # device. Remove the loopback address restriction lis-
ten to the network.
dnl #
dnl # NOTE: binding both IPv4 and IPv6 daemon to the same port requires
dnl # a kernel patch
dnl #
dnl DAEMON_OPTIONS(‘port=smtp,Addr=::1, Name=MTA-v6, Family=inet6’)dnl
dnl #
dnl # We strongly recommend not accepting unresolvable do-
mains if you want to
dnl # protect yourself from spam. However, the laptop and users on com-
puters
dnl # that do not have 24x7 DNS do need this.
dnl #
FEATURE(‘accept_unresolvable_domains’)dnl
dnl #
dnl FEATURE(‘relay_based_on_MX’)dnl
dnl #
dnl # Also accept email sent to "localhost.localdomain" as local email.
dnl #
LOCAL_DOMAIN(‘localhost.localdomain’)dnl
dnl #
dnl # The following example makes mail from this host and any additio-
nal
dnl # specified domains appear to be sent from mydomain.com
dnl #
dnl MASQUERADE_AS(‘mydomain.com’)dnl
```

Si nuestra máquina se llama `mail.midominio.com`, el correo que salga de ella, si no hacemos algo en contrario, será con direcciones del tipo: usuario@mail.midominio.com. Si queremos que salgan

con direcciones del dominio, es decir, usuario@midominio.com, utilizamos el enmascaramiento. Es normal dentro de una organización utilizar el dominio para el correo, y no direcciones de máquinas particulares.

```
dnl #
dnl # masquerade not just the headers, but the envelope as well
dnl #
dnl FEATURE(masquerade_envelope)dnl
dnl #
dnl # masquerade not just @mydomainalias.com, but @*.mydomaina-
lias.com as well
dnl #
dnl FEATURE(masquerade_entire_domain)dnl
dnl #
dnl MASQUERADE_DOMAIN(localhost)dnl
dnl MASQUERADE_DOMAIN(localhost.localdomain)dnl
dnl MASQUERADE_DOMAIN(mydomainalias.com)dnl
dnl MASQUERADE_DOMAIN(mydomain.lan)dnl
MAILER(smtp)dnl
MAILER(procmail)dnl
```

Lo conveniente es no tocar el fichero `sendmail.cf`, sino todos los cambios realizarlos sobre el fichero `sendmail.mc` y de éste generar el fichero `sendmail.cf`.

Control del Spam con Sendmail

El spam o correo no solicitado hoy día es un grave problema. Cuando surgió Internet, lo fundamental era que los protocolos y los sistemas funcionasen. Además, las comunidades de usuarios seguían unos códigos de conducta ética denominados "etiquetas de red[37]" y cualquiera podía utilizar el servidor SMTP para enviar sus correos sin mayores restricciones.

Hoy día esto no es así. Personas sin escrúpulos utilizan la red para sus fines no demasiado éticos, sin importarles mucho el resto de usuarios. Alguien puede utilizar nuestro agente de transporte y mandar miles de correos a través de él. Además, si hemos dejado nuestra máquina desprotegida y alguien la ha utilizado para enviar correo basura, nos pueden meter en una lista negra y no permitírsenos el envío de correo. Sendmail ha tenido que adaptarse a este nuevo entorno muy diferente de aquél en el que nació.

Por ello, la cofiguración por defecto, cada vez viene más cerrada. Para que se puedan enviar correos desde las máquinas clientes de nuestra red local, hay que permitíselo expresamente. Para aquellas IP locales o dominios a los que optemos por permitir que envíen correos a través de nuestro sendmail, añadiremos una entrada en el fichero `/etc/mail/access` del tipo:

```
dirección_IP RELAY
```

por ejemplo, para permitir utilizar el envío SMTP a las máquinas de la red local, añadiríamos la línea:

```
172.26.0.* RELAY
```

después de salvar el fichero, para que los cambios tengan efecto ejecutaremos en el directorio `/etc/mail` el comando

```
# make
```

para generar, a partir del fichero de texto `/etc/mail/access`, el fichero de base de datos mucho más eficiente `/etc/mail/access.db`.

[37] Net etiquette

¿Qué pasa cuando los clientes que deben conectar al servidor acceden desde el exterior con direcciones diferentes y no controlables? Existen dos soluciones posibles para controlar el uso de nuestro sendmail. Una solución se llama "POP before SMTP" y consiste en realizar una conexión POP con usuario y contraseña antes de poder conectar por SMTP. La otra se denomina SMTP_AUTH y consiste en mandar un usuario y contraseña para conectarnos al servidor SMTP. Presentamos las líneas que hay que utilizar en el fichero `sendmail.mc` de Fedora para esta segunda opción:

```
define(‘confAUTH_OPTIONS’, ‘A’)dnl
TRUST_AUTH_MECH(‘EXTERNAL DIGEST-MD5 CRAM-MD5 LOGIN PLAIN’)dnl
define(‘confAUTH_MECHANISMS’, ‘EXTERNAL GSSAPI DIGEST-MD5 CRAM-
MD5 LOGIN PLAIN’)
```

18.3. Agente de entrega: Fetchmail

Fetchmail es una utilidad que permite recuperar y reenviar correo[38]. Se encarga de buscar el correo en los servidores remotos y lo reenvía al sistema de reparto de la máquina local, desde donde podemos recuperarlo con cualquiera de los MUA normales. Podemos entenderlo como un programa que nos va a permitir bajarnos el correo de los distintos servidores y redistribuirlo entre los distintos usuarios del sistema. En sistemas con conexión permanente a internet no es muy útil.

Fetchmail se puede ejecutar en modo demonio para que sondee uno o más sistemas en un intervalo determinado.

El programa puede recoger correo de servidores que soporten cualquiera de los siguientes protocolos: POP2, POP3, IMAP2bis, IMAP4 e IMAPPrev1; también puede usar la extensión ESMTP ETRN y ODMR.

Como fetchmail se pensó principalmente para su uso sobre enlaces no permanentes TCP/IP (por ejemplo conexiones SLIP o PPP por módem), puede ser útil como un agente de transferencia de mensajes para aquellos sitios que no permiten (por motivos de seguridad) transaciones SMTP con `sendmail`.

Normalmente, fetchmail reparte cada mensaje recuperado vía SMTP hacia el puerto 25 sobre la máquina en que se está ejecutando (`localhost`), como si pasara sobre un enlace TCP/IP normal. El correo será después repartido localmente por el MDA (*Mail Delivery Agent*) de su sistema (postfix, sendmail, smail, mmdf, exim, qmail). Todos los mecanismos de reparto y agentes de transporte local funcionarán automáticamente.

Si no hay ningún puerto 25 a la escucha (y en la configuración de fetchmail indicamos un MDA local) se usará el MDA para el reparto local.

Para instalarlo ejecutaremos[39]:

```
#apt-get install fetchmail
```

En Fedora se instala por defecto, mientras que en Guadalinex tendremos que configurar varias opciones en el proceso de instalación.

[38] Agente de entrega o MDA

[39] Si en Debian tenemos problemas con la instalación, lo podemos solucionar con:

1. Añadiremos un grupo de nombre `nogroup`:

   ```
   $cat /etc/group
   ...
   nogroup:x:65534
   ```

2. Añadiremos el usuario fetchmail

   ```
   #useradd -g 65534 -d /var/run/fetchmail fetchmail
   ```

18.3.1. Configuración

Para configurar fetchmail se usa el fichero `$HOME/.fetchmailrc`. Todos los parámetros disponibles para este fichero se pueden pasar a fetchmail desde la línea de comandos. Un fichero `.fetchmailrc` puede tener:

- Opciones globales de la conexión. Los parámetros más comunes para esta sección son:

  `set daemon segundos` se ejecuta como demonio e intenta bajar el correo cada "segundos" segundos.

  `set postmaster usuario` todos los correos con problemas de entrega se mandan a "usuario"

  `set syslog /directorio/fichero` fichero para registrar los logs de fetchmail (`/var/log/maillog`)

- Opciones de servidor en el que se busca el correo. Se escriben detrás de `poll` o `skip`.

  - Con `poll` le decimos a fetchmail que baje el correo del servidor especificado cuando se ejecuta sin argumentos, es el habitual.
  - Si anteponemos `skip`, no se bajará el correo de ese servidor salvo que se lo pasemos a fetchmail como argumento en la línea de comandos.

  Las más usuales son

  `interval n` sólo se chequea este servidor cada `n` ciclos. Útil para configurar los servidores de los que rara vez recibimos correo.

  `port puerto` para asignar un número de puerto distinto del habitual

  `proto PROTOCOLO` para especificar el protocolo[40]: POP2, POP3, IMAP, APOP, KPOP

- Opciones de usuario necesarias para autenticarse ante un servidor de correo en concreto[41].

`fetchall` recoger todos los mensajes del servidor o servidores, incluso los ya vistos. Si no se especifica, se bajarán sólo los mensajes nuevos.

`fetchlimit n` número máximo de mensajes para bajar en una conexión

`flush` elimina los mensajes ya vistos, antes de iniciar la descarga de los mensajes nuevos.

`keep` para no borrar los mensajes del servidor

`limit numerobytes` limitamos el tamaño de los correos bajados

`pass8bits` permite caracteres de 8 bits

`password` contraseña a usar para ese usuario (equivale a `pass`)

`ssl` conecta con el servidor usando una conexión SSL siempre que el servidor la soporte.

`to usuario` nombre de usuario local al que enviar el correo

`user usuario` nombre de usuario en el servidor de correo

[40] Cuidado que tienen que ir en mayúsculas.

[41] A `fetchall`, `flush`, `keep`, `pass8bits` se le puede anteponer "no" para hacer lo contrario de lo explicado. Por ejemplo `no keep` borra los mensajes.

Para conocer todas las opciones disponibles: `http://www.catb.org/~esr/fetchmail/fetchmail-man.html`

Vamos a configurarlo como root[42], el motivo de hacerlo así (no es obligatorio) es que sea éste el encargado de bajarse el correo de los distintos servidores para después distribuirlo a los distintos usuarios.

Para eso crearemos en el directorio del `/root` un fichero[43] de nombre `.fetchmailrc` con las líneas:

```
# valores por defecto
defaults
# recoger todos los mensajes del servidor/es
fetchall
#borrarlos édespus de bajarlos. Si en vez de flush
#escribimos keep los mensajes no se borran del servidor.
flush
#permite caracteres de 8 bits
pass8bits
#una entrada poll por cada servidor de correo
poll servidor_de_correo_1
   #protocolo usado por el servidor. En general áser POP3
   proto pop3
   #nombre de usuario en el servidor de correo
   user "usuario1"
   #ñcontrasea en el servidor de correo
   pass "password1"
   #usuario local al que dirigir el correo
   to usuario_local_1
poll servidor_de_correo_2
   proto pop3
   user "usuario2"
   pass "password2"
   to usuario_local_2
#todos tenemos amigos de esos que piensan que el correo es
#para mandar fotos, ívdeos, etc. Si deseamos limitar el ñtamao
#del correo bajado a un ámximo de 2MB escribiremos
limit 2000000
#si deseamos que trabaje como demonio ñaadiremos esta ílnea
#para controlar nosotros la óejecucin del programa comentar esta
    ílnea
set daemon 300
```

Veamos un fichero `.fetchmailrc` de ejemplo para dos servidores de correo y reenvío a distintos usuarios del sistema (thales y mileto):

```
$ cat .fetchmailrc
defaults
fetchall
flush
#keep
pass8bits
poll tux.midominio.com
  proto pop3
  user "pvillegas"
  pass "contraseña_1"
  to thales
poll mileto.cica.es
  proto pop3
  user "ed00linux"
  pass "contraseña_2"
```

[42] Lo aquí expuesto es igualmente válido si en vez del root es cualquier usuario del sistema.

[43] Para que en Debian Sarge fecthmail trabaje como demonio el fichero de configuracion es `/etc/fetchmailrc` con permisos 0600

```
    to mileto

set daemon 300
```

Donde lo único que se ha modificado son las contraseñas de acceso a ambos servidores de correo. Este fichero debe tener unos permisos de lectura y escritura sólo del root y nadie más, así tendremos que usar

```
# chmod 0600 /root/.fetchmailrc
```

Para recibir el correo sólo tenemos que ejecutar como el usuario que tiene el fichero `.fetcmahilrc` el comando[44]:

```
$fetchmail
```

Los correos así bajados se almacenan en `/var/spool/mail` en espera de que los leamos con nuestro MUA preferido.

Si hemos optado por dejar `fetchmail` a la escucha y deseamos matarlo hay que usar

```
fetchmail --quit
```

Para Fedora: En general, interesa que `fetchmail` se inicie en el arranque, para eso sólo tenemos que poner en el subdirectorio `/etc/init.d` el fichero

```
$ cat /etc/init.d/fetchmaild
#!/bin/sh
#
# description: Automatiza el arranque de fetchmail con
# el arranque del sistema.
#
# chkconfig: 345 11 92
# config:/root/.fetchmailrc
# pidfile: /var/run/gpm.pid
# Fuente de funciones
. /etc/init.d/functions
# Obtenemos óconfiguracin
. /etc/sysconfig/network
# Verificamos que haya óconexin a red.
if [ ${NETWORKING} = "no" ]
  then
   exit 0
fi
# Comprobamos si áest presente fetchmail
[ -x /usr/bin/fetchmail ] || exit 0
start() {
  echo -n "Starting mail retrieval: fetchmail "
  /usr/bin/fetchmail -f /root/.fetchmailrc
  echo "."
}
stop() {
  echo -n "Stopping mail retrieval: fetchmail "
  /usr/bin/fetchmail -q
  echo "."
```

[44] Si hemos escrito la línea

`set daemon 300`

se quedará cargado como demonio y nos bajará el correo cada 5 minutos. Si no la hemos añadido, tendremos que ejecutar este comando cada vez que deseemos bajarnos el correo.

```
}
case "$1" in
 start)
 start;;
 stop)
 stop;;
 status)
 status fetchmail
 RETVAL=$?;;
*)
echo "Usage: /etc/init.d/fetchmail {start|stop|status}" >&2
exit 1;;
esac
exit 0
```

y añadirlo al nivel de arranque deseado.

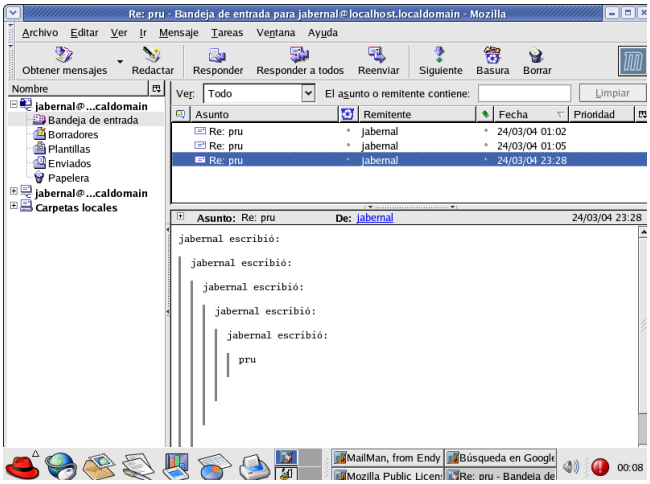
18.4. Agentes de usuario: Mozilla Mail y Ximian Evolution

Como agentes de usuario, elegiremos Mozilla Mail y Ximian Evolution. Aunque cada uno se sentirá más cómodo con su cliente de correo favorito. Los presentados aquí son a efectos de mostrar la configuración en ambos.

18.4.1. Mozilla Mail

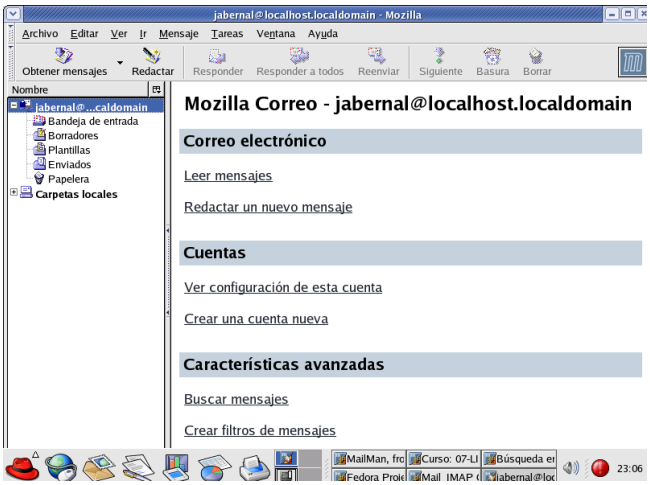
El de Mozilla tiene prácticamente todo lo que podemos necesitar y podemos utilizarlo tanto en sistemas Linux como en sistemas Windows.

Tras lanzar Mozilla y seleccionar **Ventana→Correo y Noticias**, nos aparece la ventana del cliente de correo de Mozilla.



Mozilla Mail, que así se llama el cliente de correo de Mozilla, permite gestionar varias cuentas de correo simultáneamente. Vemos que en la parte izquierda de la ventana aparecen las distintas cuentas de correo con sus carpetas correspondientes. En la parte derecha disponemos del área de mensajes, que nos muestra la lista de mensajes y el cuerpo de los que seleccionemos.

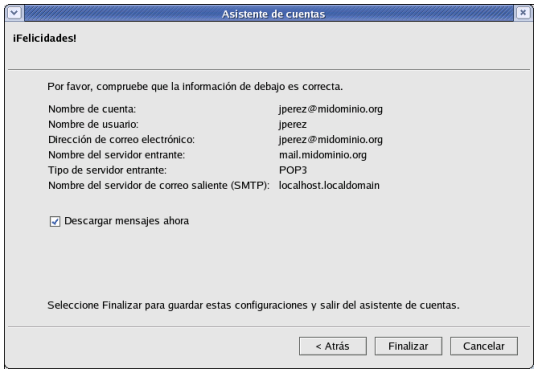
Para crear una cuenta nueva, nos situamos en alguna de las cuentas de correo existentes y nos presentará las opciones principales del cliente de correo.



También desde **Editar→Configuración de cuentas de correo y noticias**, podemos añadir acceso a una nueva cuenta de correo electrónico. La primera elección es de si se trata de una cuenta de correo electrónico o de noticias (*News*[45])

Pasamos a detallar nuestra identidad para esa cuenta de correo, indicando nuestro nombre y dirección de correo electrónico de la que se trata.

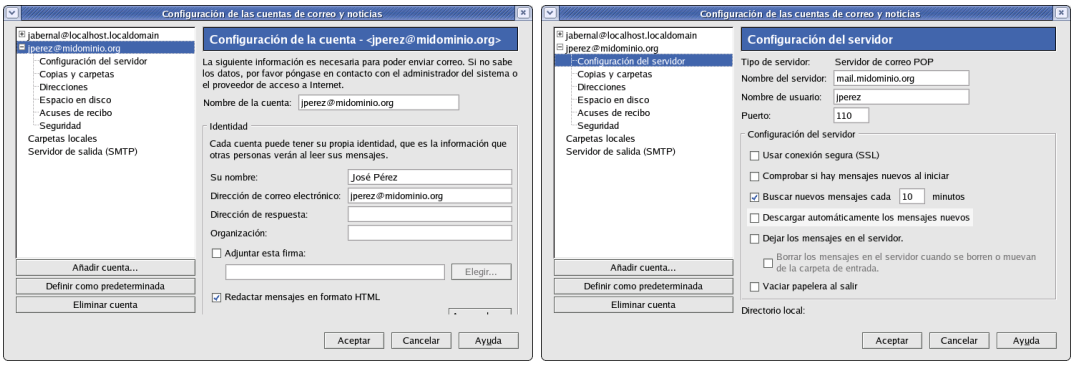
Seleccionamos a continuación si el acceso para descargarnos los correos estará disponible por POP o por IMAP, y a qué servidor nos conectaremos.



El proceso nos muestra los datos introducidos antes de aceptar la configuración.

Posteriormente, podremos acceder a los datos de la cuenta para comprobarlos o modificarlos en caso necesario, mediante la **Configuración de las cuentas de correo y noticias**.

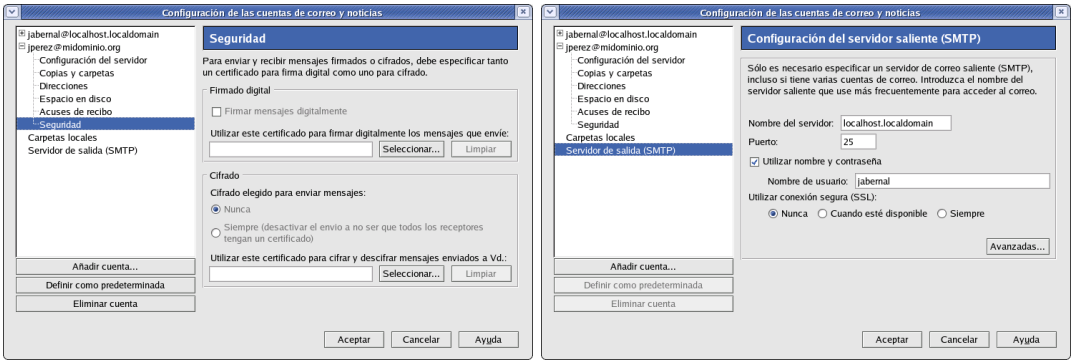
[45] Hoy día la mayor parte de grupos de news son accesibles mediante navegador web, no siendo necesarias cuentas de news específicas.



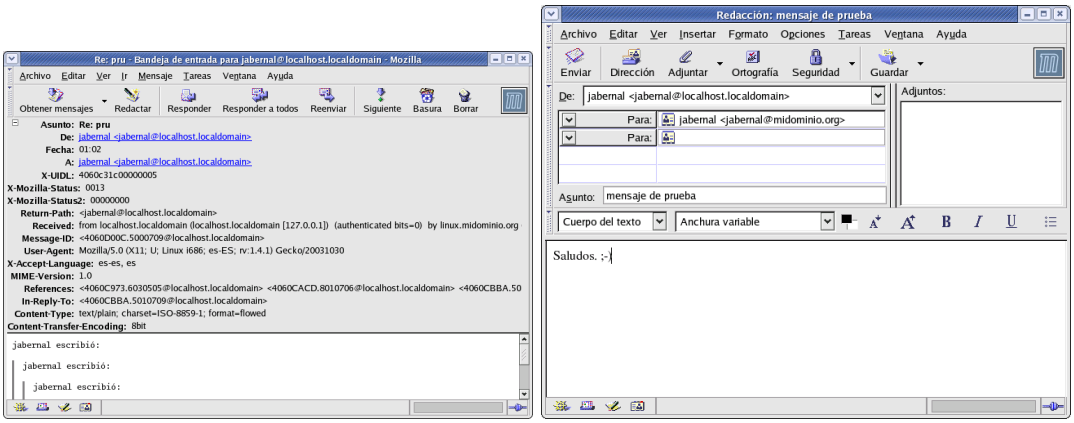
En la opción de Configuración del servidor podremos cambiar las opciones del servidor POP (o IMAP) si hubiera sido el caso. Podemos modificar el servidor, el nombre de usuario que se conecta, el puerto (que por defecto es 110), y especificar una conexión segura mediante POPS. Para POP, la opción de dejar los mensajes en el servidor nos puede permitir acceder desde distintos lugares (desde el trabajo, desde casa...) y acceder a los mensajes. Si no los dejamos en el servidor y los descargamos a un cliente, por ejemplo desde el trabajo, ya no podremos verlos desde otro lugar. Para no cargar demasiado el servidor, podemos dejar solamente aquellos de los últimos 10 días, o borrarlos al dejar la **Bandeja de Entrada** (o *Inbox*).

En las opciones de seguridad, podemos optar por firmar y cifrar digitalmente los mensajes de correo, utilizando certificados digitales x509.v3.

Pasemos a las opciones del servidor SMTP, que nos servirá para enviar el correo. Especificamos el nombre del servidor y el puerto en el que escucha, nomalmente el 25. Si hemos configurado SMTP_AUTH, podemos indicarle el nombre del usuario y la contraseña, para que nos permita utilizarlo. Además, tenemos la opción de utilizar el cifrado SSL.



En la siguiente ventana, se muestra un mensaje mostrando todos los campos de la cabecera y parte del cuerpo del mensaje.

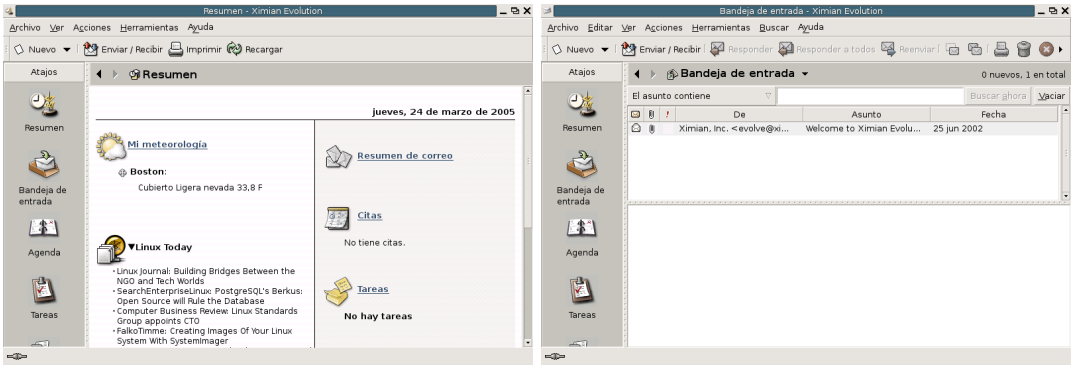


La opción de **Redactar** nos permite componer un nuevo mensaje, que enviaremos mediante la conexión al servidor de correo y el protocolo SMTP.

18.4.2. Agente de Usuario: Ximian Evolution

Otro agente de usuario es Ximian Evolution. La elección de éste está motivada porque es el que instala por defecto Guadalinex, es bastante completo y tiene todo lo que necesitamos.

Tras lanzarlo, una vez configurada la cuenta inicial, nos aparecerá la siguiente ventana:



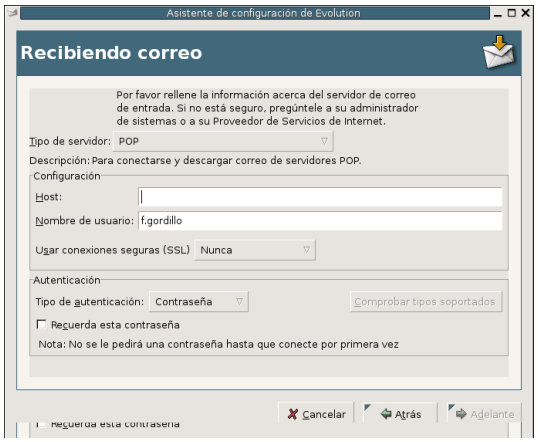
Como podemos observar, Ximian Evolution, además de la opción de cliente de correo, nos ofrece la posibilidad de gestionar una agenda, ... En nuestro caso lo que nos interesa son sus capacidades como cliente de correo, para ello en la parte derecha seleccionaremos **Resumen de Correo**. Esta opción nos mostrará el área correspondiente al cliente de correo:

Ximian Evolution permite gestionar varias cuentas de correo simultáneamente. Si seleccionamos **Ver→Barra de Carpetas**, vemos que en la parte izquierda de la ventana aparecen las distintas cuentas de correo con sus carpetas correspondientes. En la parte derecha disponemos del área de mensajes, que nos muestra la lista de mensajes y el cuerpo de los que seleccionemos.

La configuración inicial es similar a la necesaria para crear una cuenta nueva. Para crear la nueva cuenta en las opciones de menú seleccionamos **Herramientas→ Configuración**, se muestra la ventana de configuración:

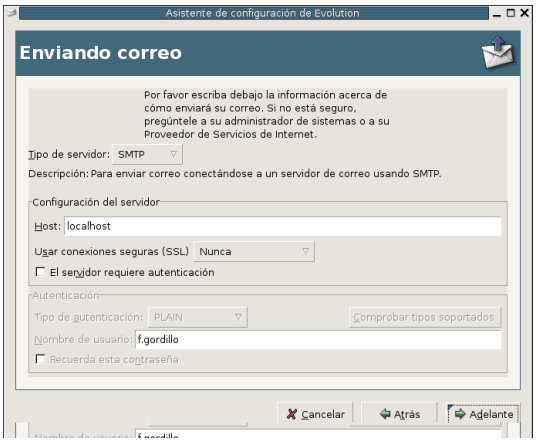
Seleccionamos la opción **Cuentas de correo**, a la derecha se muestran las cuentas de correo ya configuradas. Para añadir una nueva cuenta pulsamos **Añadir**: Se inicia el asistente de configuración para cuentas de correo, que nos aparece al iniciarlo por primera vez y que nos guiará de una forma muy intuitiva para configurar nuestra cuenta de correo:

En la siguiente pantalla podemos configurar nuestra identidad, para ello proporcionaremos nuestro nombre y nuestra dirección de correo:



A continuación proporcionaremos la información relativa al servidor de correo de entrada. Indicamos si el acceso al servidor está disponible por IMAP o POP (IMAP), dónde se encuentra nuestro servidor de correo (localhost), cuál es nuestro usuario en el servidor (mileto) y qué tipo de contraseña emplearemos. Además, si hemos configurado un canal seguro para las comunicaciones, indicaremos que se emplee siempre SSL:

Configuramos el servidor de correo saliente, debemos indicar la ubicación (localhost), el nombre de usuario y el tipo de autenticación, dado que nuestro servidor requiere autenticación y además indicaremos que se usen conexiones seguras siempre:

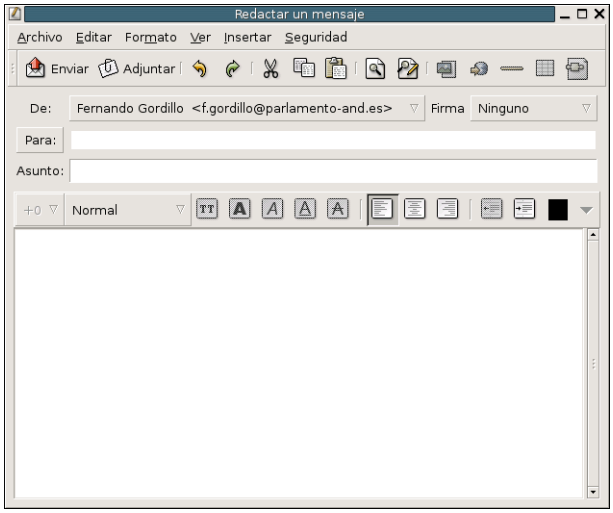


Finalmente, se nos pregunta acerca del nombre que emplearemos para identificar nuestra cuenta en el espacio de trabajo:

Para concluir, lo único que nos queda es suscribirnos a las carpetas que nos interesen de nuestro buzón de correo. Recordemos que cuando usamos IMAP, las carpetas y mensajes están directamente en el servidor. Imaginemos que de nuestra estructura de carpetas sólo nos interesa gestionar el contenido de alguna en concreto en el ordenador donde estemos trabajando, por ejemplo: si estamos en el ordenador del trabajo, quizás nos interese suscribirnos a la carpeta **Trabajo** donde almacenemos los correos de trabajo y en el ordenador de casa suscribirnos a la carpeta **Personal**. En nuestro caso aún no hemos creado carpeta por lo que nos suscribiremos a la carpeta por defecto INBOX. Seleccionamos el menú **Herramientas→Suscribirse a carpetas**...:

Seleccionamos el servidor, y la carpeta a la que deseamos suscribirnos y pulsamos [**Suscribir**]. A continuación pulsamos [**Actualizar**] para que los cambios se apliquen en el área de trabajo. Cuando nos suscribimos a una carpeta, sólo vemos el contenido de esa carpeta, no el de las carpetas hijas. Aunque en nuestra área de trabajo se muestre la estructura jerárquica, hasta dicha carpeta. Por ejemplo, si tenemos una carpeta Curso, que contiene dos carpetas Tema1 y Tema2, nos podremos suscribir a Curso, o a Tema1 o a Tema2. En cualquier caso sólo veremos el contenido de la carpeta a la que nos suscribamos, es decir, no por suscribirnos a la carpeta Curso tendremos acceso al contenido de Tema1 y Tema2.

Por último, comprobaremos que la configuración es correcta. Para ello nos enviamos a nosotros mismos un correo. En el menú seleccionamos **Nuevo**, se abre una ventana para que redactemos el correo:



Redactamos el correo destinado a nuestra cuenta y pulsamos **Enviar**. Antes de enviar, se nos solicita la contraseña del nuestro usuario en el servidor de correo, la escribimos y seleccionamos la opción de **Recuerda esta contraseña** si nos interesa.

Desde el área de trabajo pulsamos **Enviar/Recibir** para descargar el correo, que automáticamente aparecerá en nuestra bandeja **INBOX**.

18.5. Luchemos contra el SPAM: amavisd-new y spamassassin

Amavisd-new Amavisd-new[46] es una interfaz entre el MTA y uno o más filtros de contenidos, como puede ser un antivirus o un módulo antispam[47], como `SpamAssassin`. Está escrito en Perl y se comunica con el MTA vía (E)SMTP, LMTP, o mediante el uso de otros programas. No existen problemas de sincronización en su diseño que pudieran causar pérdidas de correos.

[46] Las secciones 18.5 y parte de la 18.6 están basadas en el tutorial que tenéis a vuestra disposición en `http://www.linuxsilo.net/articles/postfix.html`. Se han adecuado los contenidos a los objetivos del curso.
[47] O correo publicitario no deseado

Cuando está habilitado el uso de `SpamAssassin` (SA), se llama a SA una sola vez por mensaje (independientemente del número de destinatarios). En esta entrega veremos el funcionamiento con spamassassin y en una entrega posterior veremos cómo integrarlo con el antivirus clamav.

SpamAssassin SpamAssassin es un filtro de correo que trata de identificar el spam mediante el análisis del texto y el uso en tiempo real de algunas listas negras a través de Internet. A partir de su base de datos de reglas, utiliza un amplio abanico de pruebas heurísticas en las cabeceras y el cuerpo de los correos para identificar el spam, también conocido como correo electrónico comercial no solicitado. Una vez identificado, el correo puede ser opcionalmente marcado como spam y más tarde filtrado usando el cliente de correo del usuario. SpamAssassin normalmente identifica acertadamente entre un 95 y un 99 % del spam, dependiendo del tipo de correo que se reciba.

18.5.1. Instalación de SpamAssassin

Para instalarlo, los paquetes que necesitamos son[48] `spamassassin` y `spamc`:

```
# apt-get install spamassassin spamc
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes extras:
 libdigest-sha1-perl
Paquetes sugeridos:
 libnet-smtp-perl libmail-spf-query-perl razor libnet-ident-perl libio-
socket-ssl-perl  libdbi-perl dcc-client pyzor
Paquetes recomendados
 libnet-dns-perl
Se instalarán los siguientes paquetes NUEVOS:
 libdigest-sha1-perl spamassassin spamc
0 actualizados, 3 se instalarán, 0 para eliminar y 566 no actualiza-
dos.
Necesito descargar 847kB de archivos.
Se utilizarán 2683kB de espacio de disco adicional después de desempa-
quetar.
¿Desea continuar? [S/n]
```

Debido a que la configuración con la cual se ejecuta SpamAssassin al ser llamado desde Amavisd-new es la que se establece en el fichero de configuración de este último, no tendremos que tocar nada. En Debian[49] se trata del fichero `/etc/default/spamassassin`. Podemos observar que el demonio de SpamAssassin no se ejecuta por defecto[50], que es el comportamiento que nos conviene:

```
ENABLED=0
OPTIONS="-c -m 10 -a -H"
```

La ventaja principal de ejecutarlo como demonio sería su eficiencia, pues las comunicaciones se establecerían a través del puerto 783 en lugar de tener que arrancar un ejecutable cada vez que se tuviera que analizar un correo. En cambio, se correrían ciertos riesgos de seguridad, pues el paquete Debian nos deja una configuración por defecto que hace que se ejecute como root (en la

[48] Sólo el primero en Fedora

[49]

Fedora: `/etc/sysconfig/spamassassin` y en este caso sí se ejecuta por defecto como demonio.

[50] Si deseamos que se ejecute como demonio optaremos por:

```
ENABLED=1
```

documentación se explica cómo cambiarlo para que se ejecute como un usuario no privilegiado). Entonces, una posible vulnerabilidad a causa de un error en el código podría darnos permisos de root. En cambio, debido a que se usará SpamAssassin a través de Amavisd-new, éste será llamado a través del módulo de Perl `Mail::SpamAssassin`, manteniendo Perl el motor de reglas siempre cargado en memoria y consiguiendo la misma eficiencia que con el demonio. De hecho, éste es el comportamiento por defecto de los paquetes Debian de estas dos aplicaciones.

Si se desean utilizar los filtros bayesianos del SpamAssassin, y es muy recomendable hacerlo si se quiere tener un alto porcentaje de acierto, será preciso entrenarlo. Según el manual, varios miles de mensajes deben ser proporcionados a SpamAssassin, tanto de spam como de *ham* (que es el correo bueno, el que no es spam). Para ello se usa la herramienta `sa-learn` . Con

sa-learn –spam <directorio>

lo instruimos para que recoja información de correos que sabemos con certeza que son spam, y con

sa-learn –ham <directorio>

lo instruimos para que recoja información de correos que sabemos con certeza que no son spam. Asimismo, `sa-learn` tiene una opción que permite pasarle un fichero que contenga una lista de directorios, uno en cada línea, en los cuales buscará el tipo de correo que le especifiquemos. Este parámetro, `--folders=file`, es muy útil si queremos recoger una lista de buzones de usuarios que sabemos con seguridad que sólo guardan spam o ham y utilizarlos para continuamente mejorar nuestros filtros desde un job del cron, pues esta herramienta mantiene una lista de los correos que ya ha analizado y se los salta cada vez, haciendo este proceso bastante eficiente. Es importante tener en cuenta que la base de datos bayesiana se encuentra en `/var/lib/amavis/.spamassassin`, pues SpamAssassin es llamado a través de Amavisd-new (módulo `Mail::SpamAssassin` de Perl). Por lo tanto, cuando queramos usar la herramienta `sa-learn` deberemos hacerlo siempre con el usuario amavis.

18.5.2. Instalación de Amavisd-new

Para instalarlo en Debian, tan sólo es necesario instalar un paquete[51], como root:

```
root@guadalinex:/home/mowgli# apt-get install amavisd-new
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes extras:
libarchive-tar-perl libarchive-zip-perl libcompress-zlib-
perl libconvert-tnef-perl libconvert-uulib-perl libio-multiplex-
perl libio-string-perl libio-stringy-perl libio-zlib-perl libmailtools-
perl libmime-perl libnet-server-perl libtimedate-perl libunix-syslog-
perl Paquetes sugeridos: zoo nomarch apt-listchanges libmail-audit-
perl libio-socket-ssl-perl
Se instalarán los siguientes paquetes NUEVOS:
amavisd-new libarchive-tar-perl libarchive-zip-perl libcompress-zlib-
perl libconvert-tnef-perl libconvert-uulib-perl libio-multiplex-
perl libio-string-perl libio-stringy-perl libio-zlib-perl libmailtools-
perl libmime-perl libnet-server-perl libtimedate-perl libunix-syslog-
```

[51] El paquete para Fedora lo podemos encontrar en `http://dag.wieers.com/packages/amavisd-new/`. Si lo bajamos y deseamos instalarlo con el comando rpm presenta múltiples problemas de dependencias. Así que en este caso, lo más sencillo es usar los repositorios de la distribución anterior e instalarlo después. Para eso, añadiremos la línea

```
rpm http://apt.sw.be fedora/3/en/i386 dag
```

al fichero `/etc/apt/source.list` de Fedora y después ejecutamos:

```
#apt-get update
#apt-get install amavisd-new
```

```
perl 0 actualizados, 15 se instalarán, 0 para eliminar y 562 no actua-
lizados. Necesito descargar 1385kB de archivos.
Se utilizarán 4592kB de espacio de disco adicional después de desempa-
quetar.
¿Desea continuar? [S/n]
```

El fichero de configuración de Amavisd-new es bastante largo, pero tan sólo es necesario realizar unos pocos cambios a la configuración que viene por defecto. A continuación se presentan las líneas del fichero `/etc/amavis/amavisd.conf` que necesitan ser modificadas, en el formato definitivo (es decir, con las modificaciones ya realizadas):

```
$mydomain = 'midominio.com';
$myhostname = 'mail.midominio.com';
# @bypass_spam_checks_acl = qw( . );
$final_spam_destiny = D_PASS;
$warnbannedsender = 1;
$warnbadhsender = 1;
# $virus_quarantine_to = 'virus-quarantine';
$virus_quarantine_to ="virus-quarantine\@$mydomain";
# $sa_spam_subject_tag = '***SPAM*** ';
$banned_filename_re = new_RE(
# qr'^UNDECIPHERABLE$',
  qr'\.[^.]*\.(exe|vbs|pif|scr|bat|cmd|com|dll)$'i,
  qr'[{}]',
#qr'.\.(exe|vbs|pif|scr|bat|cmd|com)$'i,
qr'.\.(ade|adp|bas|bat|chm|cmd|com|cpl|crt|exe|hlp|hta|inf|ins|isp
js|jse|lnk|mdb|mde|msc|msi|msp|mst|pcd|pif|reg|scr|sct|shs|shb|vb|
vbe|vbs|wsc|wsf|wsh)$'ix,
  qr'.\.(mim|b64|bhx|hqx|xxe|uu|uue)$'i,
# qr'^\.(zip|lha|tnef|cab)$'i,
  qr'^\.exe$'i,
  qr'^application/x-msdownload$'i,
  qr'^application/x-msdos-program$'i,
  qr'^message/partial$'i,
  qr'^message/external-body$'i,
);
```

Estas modificaciones realizadas al fichero dejan una configuración específica para que amavisd-new se comporte de la manera que nosotros queremos, pero lo más habitual es que deba personalizarse para cada caso. Acto seguido se resumen los porqués de los cambios realizados:

- `bypass_spam_checks_acl`: comentamos esta línea para que Amavisd-new use SpamAssassin (por defecto viene deshabilitado su uso).
- `final_spam_destiny`: dejamos pasar los correos identificados como spam, aunque siguen siendo marcados como tales mediante cabeceras en el correo. De este modo, los destinatarios seguirán recibiendo toda su correspondencia pero podrán filtrarla fácilmente usando el cliente de correo y las cabeceras que Amavisd-new habrá añadido al mensaje.
- `warnbannedsender`: activamos el envío de un mensaje de aviso al remitente de un mensaje que contuviera algún fichero adjunto con una de las extensiones prohibidas que más abajo se detallan.
- `warnbadhsender`: igual que el anterior para ficheros con cabeceras mal formadas.

- `virus_quarantine_to`: activamos la cuarentena de los correos con virus. De este modo, cualquier correo que contenga un virus detectado, será redirigido a la cuenta especificada. Así, podremos revisarlos y decidir qué hacer con ellos.
- `sa_spam_subject_tag`: al comentar esta sentencia se desactiva la modificación del asunto del mensaje, pues con las cabeceras que se han añadido es suficiente para que nuestro cliente de correo filtre adecuadamente.
- `banned_filename_re`: rechazamos correos que contengan ficheros adjuntos con alguna de las extensiones mencionadas en esta variable (únicamente se permiten ficheros comprimidos), principalmente ejecutables y scripts.

Tras esto ya podemos reiniciar el servicio mediante el comando

```
#/etc/init.d/amavisd-new restart
```

y observar su carga en el log `/var/log/mail.log`, donde se informa de todos los módulos cargados al iniciar.

Con la instalación de amavisd vista en este apartado será inmediato instalar el antivirus clamav en una entrega posterior.

18.5.3. Modificaciones en Postfix

Las modificaciones a realizar en Postfix son muy sencillas.

- Fichero `/etc/postfix/master.cf`: Editamos el fichero y le añadimos las siguientes líneas:

```
127.0.0.1:10025 inetn   -   n   -   -   smtpd
-o content_filter=
-o local_recipient_maps=
-o relay_recipient_maps=
-o smtpd_restriction_classes=
-o smtpd_client_restrictions=
-o smtpd_helo_restrictions=
-o smtpd_sender_restrictions=
-o smtpd_recipient_restrictions=permit_mynetworks,reject
-o mynetworks=127.0.0.0/8
-o strict_rfc821_envelopes=yes
-o smtpd_error_sleep_time=0
-o smtpd_soft_error_limit=1001
-o smtpd_hard_error_limit=1000
smtp-amavis unix - - n - 2 lmtp
-o lmtp_data_done_timeout=1200
-o lmtp_send_xforward_command=yes
```

- Fichero `/etc/postfix/main.cf`: En este fichero debemos añadir la siguiente línea:

```
content_filter = smtp-amavis:[127.0.0.1]:10024
```

Con estas modificaciones, lo que conseguimos es añadir un filtro de contenido a Postfix, el cual redirigirá el tráfico al puerto 10024 de la interfaz loopback. Una vez amavisd-new haya finalizado su trabajo, devolverá el mensaje a Postfix a través del puerto 10025, donde hemos habilitado un smtpd.

Ahora tan sólo queda reiniciar el servidor Postfix:

```
# /etc/init.d/postfix restart
Stopping mail transport agent: Postfix.
Starting mail transport agent: Postfix.
```

Para comprobar que todo está funcionando correctamente, podemos enviar un correo a un usuario de nuestro servidor y comprobaremos que se está filtrando el contenido a través de amavisd-new, observando el fichero de log `/var/log/mail.log`.

Una medida de seguridad que podemos tomar es utilizar otros puertos diferentes. Tendremos que ir afinando la configuración de amavisd en lo que respecta a las políticas de filtrado, una primera forma de comenzar es desechando los virus y permitiendo, pero con una modificación de las cabeceras, el spam y los correos no permitidos ("banned").

18.6. Gestores de listas de correo: Mailman

18.6.1. ¿Qué es una lista de correo?

Una lista de correo es simplemente una dirección de correo que contiene un grupo de direcciones a las cuales se envía la misma información. En el caso de las listas de correo electrónico, se usa una lista de direcciones de correo electrónico de gente interesada en escuchar o discutir sobre un tema determinado.

Entre los tipos más comunes de listas de correo electrónico están las listas de anuncios y las listas de discusión.

Las listas de anuncios sirven para que una o más personas puedan informar a un grupo más numeroso de personas.

Una lista de discusión permite a un grupo de personas, debatir temas entre ellos mismos, pudiendo cada uno enviar mensajes a la lista y hacer que se distribuyan a todos los integrantes del grupo. Esta discusión también se puede moderar, de manera que sólo los mensajes a los cuales el administrador les haya dado el visto bueno serán distribuidos a la lista. También es posible hacer que sólo a ciertas personas se les permita enviar mensajes a la lista. Dando lugar a la división entre las listas abiertas, en las que cualquiera puede enviar, y las listas cerradas, en las que sólo sus integrantes pueden enviar información.

Algunos términos comunes son:

- Un "envío" denota un mensaje que se envía a una lista de correo.
- A las personas que son parte de una lista de correo electrónico normalmente se las llama "suscriptores" de la lista.
- "Los administradores de las listas" son personas encargadas de mantener esas listas. Las listas pueden tener uno o más administradores.
- Una lista puede tener también personas encargadas de leer los mensajes enviados a la lista y decidir si éstos deberían ser distribuidos a todos los suscriptores. A estas personas se les llama moderadores.
- A menudo varias listas de correo electrónico utilizan el mismo software. A la persona que mantiene el software gracias al cual funcionan las listas se le llama el "administrador del sitio." A menudo el administrador del sitio también administra listas individuales.

18.6.2. Mailman

GNU Mailman es un programa que permite administrar listas de correo electrónico, con soporte para un rango amplio de tipos de listas de correo, tales como listas de discusión general y listas de sólo anuncios. Mailman tiene características para los suscriptores, tales como: facilidad en la suscripción y desuscripción, opciones de privacidad, y capacidad de detener temporalmente la recepción de los envíos a la lista.

Mailman también tiene muchas características para facilitar la tarea a los administradores de listas y a los administradores de sitio.

Instalación y Configuración

Para la instalación de mailman vamos a seguir los pasos en Guadalinex 2004 y posteriormente comentaremos las diferencias con Fedora.

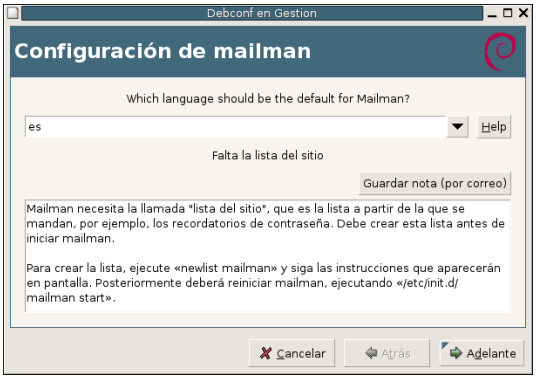
Antes de realizar la instalación de mailman debemos tener en cuenta las dependencias de dicho paquete y el gestor de correo que se va a utilizar. En este caso, y siguiendo el orden del curso, se va a utiliar como MTA el software postfix, supuestamente ya instalado.

En el momento de escribir esta documentación, la versión que utiliza guadalinex para mailman es la 2.1.5-1, si realizamos la instalación:

```
# apt-get install mailman
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes extras:
  pwgen
Paquetes sugeridos:
  python2.2-korean-codecs python2.3-korean-codecs python-japanese-
codecs
Se instalarán los siguientes paquetes NUEVOS:
  mailman pwgen
0 actualizados, 2 se instalarán, 0 para eliminar y 119 no actualizados.
Necesito descargar 6618kB de archivos.
Se utilizarán 32,4MB de espacio de disco adicional después de desempa-
quetar.
¿Desea continuar? [S/n] s
```

Mailman dependende además de Apache, nosotros partiremos de la premisa de que Apache 2 ya está instalado.

Durante la instalación nos aparecerán las pantallas de postconfiguración para el idioma, que dará igual lo que seleccionemos porque se instalará en inglés, y posteriormente nos indica que debemos crear la lista mailman antes de arrancar.



Tras la instalación del paquete, es conveniente leer la documentación disponible en `/usr/share/doc/mailman`, principalmente en `/usr/share/doc/mailman/README.Debian.gz` y en `/usr/share/doc/mailman/README.POSTFIX.gz`. En ella se detallan las modificaciones necesarias en Apache para el correcto funcionamiento de la interfaz web, así como la forma de integrar Postfix y Mailman.

A continuación debemos modificar ligeramente el fichero `/etc/mailman/mm_cfg.py` a fin de que Mailman sepa que se está trabajando con Postfix como *Mail Transport Agent*, para eso descomentamos la siguiente línea:

```
MTA = 'Postfix'
```

La configuración por defecto de Postfix nos deja la directiva `alias_maps` apuntando a `/etc/aliases`. Ya que no nos interesa estar modificando este fichero y ejecutando el comando newaliases de Postfix cada vez que creemos o borremos una lista, utilizaremos el fichero de alias propio de Mailman, que es automáticamente actualizado por los comandos `newlist` y `rmlist`.

El primer paso será generarlo:

```
# cd /var/lib/mailman
# bin/genaliases
```

A continuación añadiremos ese fichero de alias a la directiva `alias_maps` del `/etc/postfix/main.cf`, además de otras directivas necesarias:

```
alias_maps=hash:/etc/aliases,hash:/var/lib/mailman/data/aliases
mailman_destination_recipient_limit = 1
unknown_local_recipient_reject_code = 550
owner_request_special = no
recipient_delimiter = +
```

Algunas de estás opciones estarán ya definidas previamente.

Y solicitaremos a Postfix que recargue la configuración:

```
#/etc/init.d/postfix reload
```

El tercer paso de la instalación de Mailman nos avisa de que es necesario crear una *sitelist* llamada mailman y que hasta que no la creemos, el demonio del Mailman no arrancará. Ahora es el momento de crearla y, para ello, ejecutamos el siguiente comando:

```
# newlist mailman
Enter the email of the person running the list: mileto@midominio.com
Initial mailman password:
Hit enter to notify mailman owner...
```

Nótese que Mailman no nos muestra la lista de alias que nos requiere que añadamos a nuestro fichero de alias. Esto es debido a la configuración realizada más arriba, eliminándose de esta manera este tedioso paso. Podemos, por lo tanto, pulsar enter y pasar a iniciar el demonio de Mailman mediante el comando

```
#/etc/init.d/mailman start
```

Una vez iniciado el servicio, recibiremos el correo que nos notifica la creación de la lista en la dirección de correo que hayamos especificado (mileto@midominio.com en este ejemplo).

Las listas que creemos en el futuro tampoco nos solicitarán que añadamos manualmente la lista de alias. Cuando se añada o quite una lista, el fichero `/var/lib/mailman/data/aliases.db` será automáticamente actualizado, pero no se ejecutará automáticamente un

```
#/etc/init.d/postfix reload
```

Esto es debido a que es necesario ser root para ejecutar este comando y los scripts suid-root no son seguros. El único efecto de esto es que le llevará aproximadamente un minuto a Postfix darse cuenta de los cambios y actualizar sus tablas, si bien esto se puede considerar una inconveniencia menor.

Por último nos queda confirmar que se puede acceder vía web. Este paso dependerá del apache instalado y la configuración que tenga. Los scripts de mailman se encuentran en `/usr/lib/cgi-bin/mailman/`, también se necesita acceso a los archivos de la lista en `/var/lib/mailman/archives/public/` y a las imagenes `/usr/share/images/mailman/`. Según el apache que se esté utilizando, habilitaremos la ejecución de cgi's mediante `a2enmod cgi` (en la versión apache2). Es posible definir alias para que el acceso sea más sencillo, aquí la gestión la haremos utilizando el path por defecto.

Gestión de las listas

Normalmente será el administrador del site el que utilice la línea de comandos para gestionar la creación, propiedades y borrado de las listas. Para esto, utilizará los comandos del directorio `/usr/lib/mailman/bin/` entre los cuales están:

newlist para generar una nueva lista

add_members con el que se añaden miembros a una lista

mmsitepass para modificar passwords, por ejemplo -c para el creador de listas

list_* serie de comandos para listar los elementos listas, miembros,...

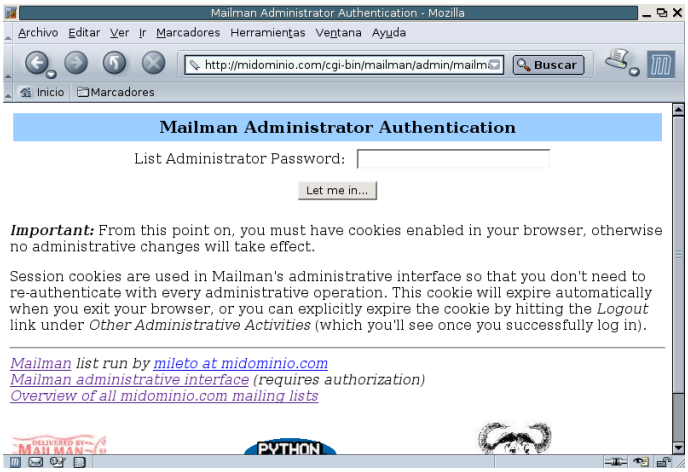
remove_members borrar miembros de listas

rmlist borrar listas

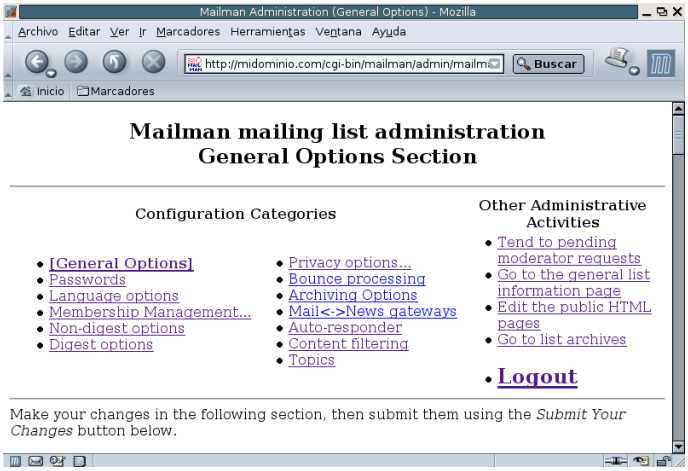
Los ficheros de log se encuentran en `/var/log/mailman/` y los de configuración en `/etc/mailman` y `/var/lib/mailman`. Además del administrador del site que gestionará los ficheros y comandos anteriores, existen otros roles con privilegios como el administrador de una lista que controla todo lo referente a dicha lista, el moderador de una lista que aprueba los mensajes para su envío y el creador de listas que puede crear listas aunque no gestionarlas.

Todos los roles al igual que los usuarios de las listas podrán utilizar el correo electrónico para algunas operaciones así como utilizar el interfaz web. En nuestro caso, vamos a ver con mayor detalle el interfaz web y dado que nuestro dominio lo hemos puesto como midominio.com vamos a hacer que en `/etc/hosts` ese dominio apunte al localhost o bien modificamos el fichero `/etc/mailman/mm_cfg.py` para indicar la URL del Host. De esta forma accedemos a

`http://localhost/cgi-bin/mailman/admin/nombre_lista`



Con la contraseña de administración, accedemos al menú donde podemos configurar todos los parámetros de la lista.



Para crear nuevas listas se aconseja no habilitar el uso del creador de listas por motivos claros de seguridad, es aconsejable crearla en la línea de comandos y posteriormente configurarla vía web.

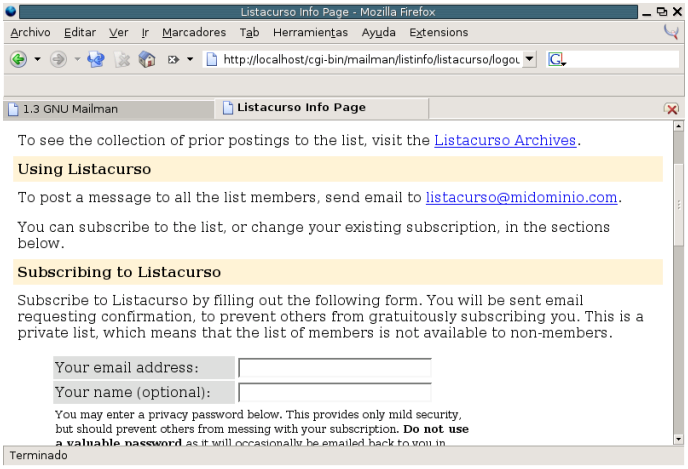
Ejemplo de ello:

```
#newlist listacurso
```

Desde

`http://midominio.com/cgi-bin/mailman/admin/listacurso` configuamos la lista y desde

`http://midominio.com/cgi-bin/mailman/listinfo/listacurso` permite que el usuario se de alta y de baja en la lista



Con más detalle tenemos:

Página de información de la lista (listinfo) Usualmente se encuentra en `http://SERVIDORWEB/mailman/listinfo/NOMBRELISTA`

La página listinfo es el punto de inicio del interfaz del suscriptor. Como se podría asumir por el nombre dado, esta página contiene información acerca de la lista NOMBRELISTA. A partir de esta página se puede llegar al resto de páginas del suscriptor, así que realmente sólo se necesita conocer esta dirección.

Página de opciones del suscriptor Usualmente se encuentra en
`http://SERVIDOR/mailman/options/NOMBRELISTA/CORREO`

También se puede acceder a esta página yendo a la página listinfo y escribiendo su dirección de correo en el cuadro de texto junto al botón etiquetado "**Opciones de Edición y Desuscripción**" (éste está cerca del final de la página).

La página de opciones de suscriptor le permite entrar/salir y cambiar la configuración de sus opciones, así como también desuscribirse u obtener una copia de su contraseña por correo electrónico.

Para acceder a su página de opciones de suscriptor: Si aún no ha entrado, encontrará un cuadro de texto cerca de la parte superior de la página para introducir la contraseña. Escriba su contraseña en el cuadro de texto mencionado y haga clic en el botón "**Cambiar**".

Una vez dentro, podrá mirar y cambiar toda la configuración personal de su lista.

Archivos de la Lista Usualmente los encontrará en
`http://SERVIDORWEB/pipermail/NOMBRELISTA`
si la lista se archiva públicamente, y
`http://SERVIDORWEB/mailman/private/NOMBRELISTA`
si la lista se archiva en forma privada.

Las páginas de los archivos de la lista disponen de una copia de los mensajes enviados a la lista, normalmente agrupados por mes. En cada grupo mensual, los envíos se indexan por autor, fecha, hilo y asunto.

Toda lista de correo tiene un conjunto de direcciones de correo electrónico a las cuales se pueden enviar los mensajes, incluyendo, una dirección para enviar los mensajes a la lista, una dirección destinada a recibir mensajes devueltos y direcciones para procesar órdenes de correo.

Para una lista de correo ficticia llamada *listacurso@midominio.com*, nos encontraremos estas direcciones:

listacurso@midominio.com ésta es la dirección de correo para enviar mensajes a la lista.

listacurso-join@midominio.com enviando un mensaje a esta dirección, un nuevo miembro puede solicitar suscripción a la lista, pero si se hace, Mailman ignora tanto la cabecera de Asunto: como el cuerpo de tal mensaje.

listacurso-leave@midominio.com enviando un mensaje a esta dirección un miembro puede solicitar que se le dé de baja de la lista. Igual que con la dirección -join, se ignora la cabecera Asunto: y el cuerpo del mensaje.

listacurso-owner@midominio.com Esta dirección permite contactar con el propietario o moderador de la lista. Esta es la dirección que deberá utilizar cuando necesite contactar a la persona o personas encargadas de la lista.

listacurso-request@midominio.com Esta dirección está asociada a un robot de correo que procesa órdenes de correo electrónico que se pueden usar para definir las distintas opciones de los suscriptores, así como también para procesar otras órdenes.

listacurso-bounces@midominio.com Esta dirección se usa para el procesamiento automático de los mensajes devueltos de Mailman.

listacurso-confirm@midominio.com Esta dirección se usa para procesar mensajes de confirmación de las solicitudes de suscripción y desuscripción.

También hay una dirección -admin que permite contactar a los administradores de las listas. Esta dirección sólo existe por compatibilidad con las versiones más antiguas de Mailman.

Para cambiar las opciones, se usa la dirección `NOMBRELISTA-request`.

Para un mayor conocimiento de mailman, se recomienda la lectura de la guía de usuario y de administrador, así como posibles cambios, en `http://www.gnu.org/software/mailman`

Mailman en Fedora Core 3

Para Fedora Core 3 podemos realizar la instalación mediante rpm o bien utilizar el apt-get, con este último obtendríamos lo siguiente:

```
#apt-get install mailman
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
Se instalarán los paquetes NUEVOS siguientes:
mailman
0 upgraded, 1 newly installed, 0 removed and 156 not upgraded.
Need to get 7979kB of archives.
After unpacking 27,8MB of additional disk space will be used.
Get:1 http://ayo.freshrpms.net fedora/linux/3/i386/updates mailman
3:2.1.5-30.fc
3 [7979kB]
Fetched 7979kB in 10s (754kB/s)
Committing changes...
Preparing... ##########################################[100%]
1:mailman ##########################################[100%]
Done.
```

Una vez instalado, es conveniente leer la documentación de `/usr/share/doc/mailman-version/README.POSTFIX` para ver que los pasos de la instalación son similares a los indicados en el apartado anterior. Destacar algunos cambios que, aunque poco significativos, se encuentran en esta distribución:

`/etc/mailman/sitelist.cf` Este fichero contiene parámetros de configuración por defecto de las listas de correo

`/etc/httpd/conf.d/mailman.conf` En este fichero se encuentra la configuración de apache para acceder vía web.

`/usr/lib/mailman/` Directorio en el que se encuentran los ficheros de configuración, binarios y plantillas en varios idiomas.

`/usr/share/doc/mailman-2.1.5` Directorio de documentación donde podremos leer sobre las diferentes instalaciones e integraciones.

18.7. Correo Web: SquirrelMail

¿Qué es exactamente SquirrelMail? Se trata de un interface, o cliente, de correo escrito en PHP4. Se ha diseñado para permitir acceso al correo a través de su servidor desde cualquier parte del mundo empleando la "web".

18.7.1. Instalación

Squirrelmail es un potente sistema de correo web. ¿Cuántas veces cuando estamos fuera de casa o del trabajo sentimos la necesidad de acceder a nuestro correo con todas sus funciones? Leer correo, enviarlo, acceder a la libreta de direcciones... Sin más programas que un navegador y acceso a Internet, podremos acceder desde cualquier sitio[52] a nuestro correo, en el caso de que tengamos nuestro servidor accesible desde Internet.

SquirrelMail es un programa de webmail escrito en PHP4 que proporciona toda la funcionalidad que esperamos de un cliente de correo, incluyendo ficheros adjuntos, libreta de direcciones y uso

[52] Desde casa, el trabajo, un cibercafé, el ordenador de un amigo...

de carpetas. Soporta los protocolos IMAP y SMTP y todas las páginas se generan en HTML 4.0 (sin Javascript) para obtener la máxima compatibilidad con los navegadores.

Las conexiones a las carpetas de usuario se realizan mediante el protocolo IMAP, que permite la creación y manipulación de carpetas en el servidor. Así que partimos de que Apache, PHP4 e IMAP ya funcionan correctamente.

Hay que activar un servidor `imap`, aunque hay varios disponibles (Cyrus, uw-imapd...), nos decantaremos por dovecot, que funciona sin problemas tanto en Fedora como Guadalinex. Lo podemos instalar en ambos, si no lo está ya, con

```
#apt-get install dovecot
```

En Fedora

Para instalarlo, utilizamos `apt-get`. Él se encargará de instalar los paquetes necesarios.

```
[root@linux root]# apt-get install squirrelmail
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
Se instalarán los paquetes NUEVOS siguientes:
squirrelmail
0 upgraded, 1 newly installed, 0 removed and 51 not upgraded.
Need to get 2585kB of archives.
After unpacking 8194kB of additional disk space will be used.
Get:1 http://ayo.freshrpms.net fedora/linux/1/i386/core squirrelmail 1.4.0-1 [2585kB]
Fetched 2585kB in 1m39s (25,9kB/s)
Committing changes...
Preparing... ########################################### [100%]
1:squirrelmail ########################################### [100%]
Done.
```

En Debian

Partimos de que se han instalado los paquetes de apache2 y php4. Después ejecutaremos:

```
root@guadalinex:~/curso-linux/3# apt-get install squirrelmail
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes extras:
  libapache2-mod-php4 libzzip-0-12 php4-common squirrelmail-locales
Paquetes sugeridos:
  php4-pear phpdoc imap-server
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-php4 libzzip-0-12 php4-
common squirrelmail squirrelmail-locales
0 actualizados, 5 se instalarán, 0 para eliminar y 608 no actualizados.
Se necesita descargar 1805kB/6377kB de archivos.
Se utilizarán 22,3MB de espacio de disco adicional después de desempa-
quetar.
¿Desea continuar? [S/n]
Des:1 http://ftp.fi.debian.org sarge/main php4-common 4:4.3.10-
9 [164kB]
Des:2 http://ftp.fi.debian.org sarge/main libapache2-mod-php4 4:4.3.10-
9 [1642kB]
Descargados 1805kB en 37s (48,1kB/s)
Seleccionando el paquete libzzip-0-12 previamente no seleccionado.
```

```
(Leyendo la base de datos ...
112469 ficheros y directorios instalados actualmente.)
Desempaquetando libzzip-0-12 (de .../libzzip-0-12_0.12.83-
4_i386.deb) ...
Seleccionando el paquete php4-common previamente no seleccionado.
Desempaquetando php4-common (de .../php4-common_4%3a4.3.10-
9_i386.deb) ...
Seleccionando el paquete libapache2-mod-
php4 previamente no seleccionado.
Desempaquetando libapache2-mod-php4 (de .../libapache2-mod-
php4_4%3a4.3.10-9_i386.deb) ...
Seleccionando el paquete squirrelmail-
locales previamente no seleccionado.
Desempaquetando squirrelmail-locales (de .../squirrelmail-
locales_1.4.4-20050122-1_all.deb) ...Seleccionando el paquete squirrel-
mail previamente no seleccionado.
Desempaquetando squirrelmail (de .../squirrelmail_2%3a1.4.4-
3_all.deb) ...
Configurando libzzip-0-12 (0.12.83-4) ...
Configurando php4-common (4.3.10-9) ...
Configurando libapache2-mod-php4 (4.3.10-9) ...
Forcing reload of web server: Apache2.
Configurando squirrelmail-locales (1.4.4-20050122-1) ...
Configurando squirrelmail (1.4.4-3) ...
Installing default squirrelmail config.
Run /usr/sbin/squirrelmail-
configure as root to configure/upgrade config.
Para configurarlo, podemos ver la sección siguiente. También exis-
te una utilidad de configuración:
#/usr/sbin/squirrelmail-configure
```

para conseguir el mismo entorno que en el gráfico 18.6 en la página 382.

18.7.2. Configuración

Empecemos la configuración. Para ello, veamos los ficheros implicados[53]:

Como ya sabemos, apache leerá los ficheros[54] del directorio `/etc/httpd/conf.d`. En este caso `squirrelmail.conf`

```
[root@linux images]# more /etc/httpd/conf.d/squirrelmail.conf
#
# SquirrelMail is a webmail package written in PHP.
#
Alias /webmail /usr/share/squirrelmail
```

Lo que obtenemos de él es que apuntando nuestro navegador a `http://localhost/webmail`[55] accedemos a la entrada del correo web, como vemos en la figura. En realidad, los ficheros necesarios están ubicados en `/usr/share/squirrelmail`.

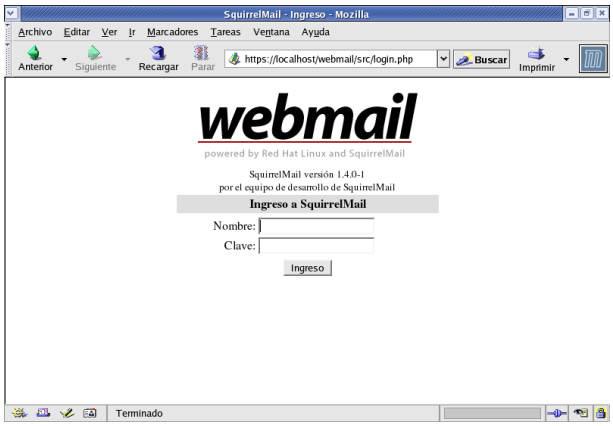
[53] Lo haremos sobre Fedora, trasladarlo a Debian es similar.

[54] En Debian, tendremos que añadir la línea

```
Alias /webmail /usr/share/squirrelmail
```

al fichero de configuración de Apache: `/etc/apache2/apache2.conf`,
O hacer el siguiente enlace: #ln -s /usr/share/squirrelmail /var/www/webmail

[55] O `http://www.midominio.com` y, lo mejor por SSL: `https://www.midominio.com`



La configuración de la herramienta se encuentra en el fichero `/etc/squirrelmail/config.php`. Veamos sus características principales:

```
<?php
/**
* SquirrelMail Configuration File
* Created using the configure script, conf.pl
*/
global $version;
$config_version = '1.4.0';
$config_use_color = 1;
$org_name = "SquirrelMail";
$org_logo = SM_PATH . 'images/sm_logo.png';
$org_logo_width = '308';
$org_logo_height = '111';
$org_title = "SquirrelMail $version";
$signout_page = ' ';
$frame_top = '_top';
$provider_uri = 'http://www.squirrelmail.org/';
$provider_name = 'SquirrelMail';
$motd = "";
//$squirrelmail_default_language = 'en_US';
$squirrelmail_default_language = 'es_ES';
```

Cambiamos a `es_ES` para que nos aparezca en castellano.

```
$domain = 'localhost';
$imapServerAddress = 'localhost';
$imapPort = 143;
$useSendmail = true;
$smtpServerAddress = 'localhost';
$smtpPort = 25;
$sendmail_path = '/usr/sbin/sendmail';
$pop_before_smtp = false;
$imap_server_type = 'other';
```

Configuramos el dominio de correo y los servidores IMAP y SMTP y puertos a los que se conecta. Con la configuración por defecto, funcionaría directamente sobre la misma máquina en la que está el servidor web, aunque podría ser otro servidor distinto.

```
$invert_time = false;
```

```
$optional_delimiter = '/';
$default_folder_prefix = 'mail/';
$trash_folder = 'Trash';
$sent_folder = 'Sent';
$draft_folder = 'Drafts';
$default_move_to_trash = true;
$default_move_to_sent = true;
```

Contiene opciones de las carpetas por defecto. Existen más opciones, pero normalmente no las tocaremos.

También podremos configurar nuestro sistema SquirrelMail mediante menús con el comando[56]

`/usr/share/squirrelmail/config/conf.pl`

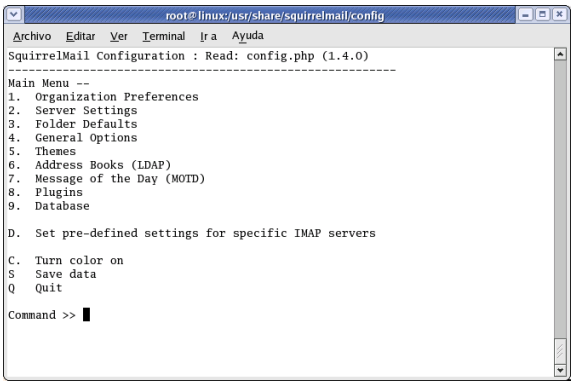


Figura 18.6: Configuración Squirrelmail

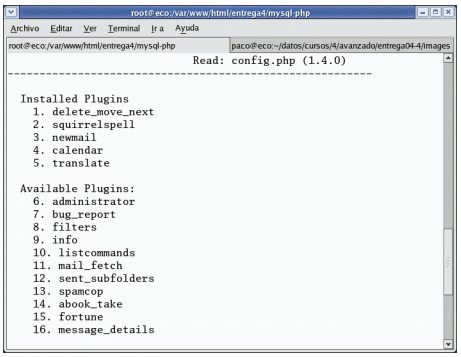
Por ejemplo, si pulsamos 1, podremos establecer el nombre de nuestro sitio, el idioma por defecto, el logo, ...:

```
Organization Preferences
1. Organization Name : Mileto
2. Organization Logo : ../images/logo.jpg
3. Org. Logo Width/Height : (500/100)
4. Organization Title : SquirrelMail $version
5. Signout Page :
6. Default Language : es_ES
```

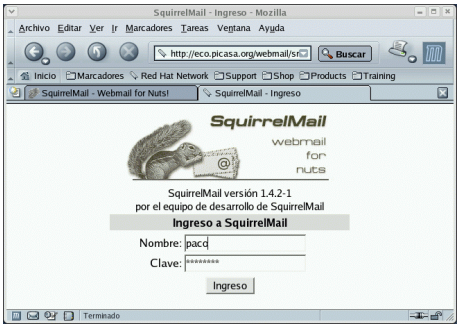
Si en el menú principal pulsamos sobre 8, accederemos a la posibilidad de añadir funcionalidades añadidas a la aplicación, por ejemplo, calendarios, filtros, etc

[56] En Debian

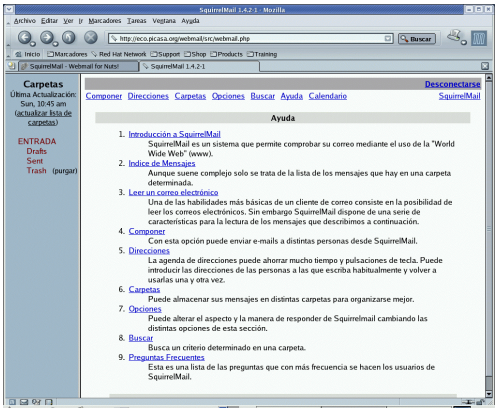
```
#/usr/sbin/squirrelmail-configure
```



Una vez que nuestro sistema está configurado a nuestro gusto, no tenemos más que conectarnos[57] e introduciendo el usuario y contraseña del correo, accedemos a éste.



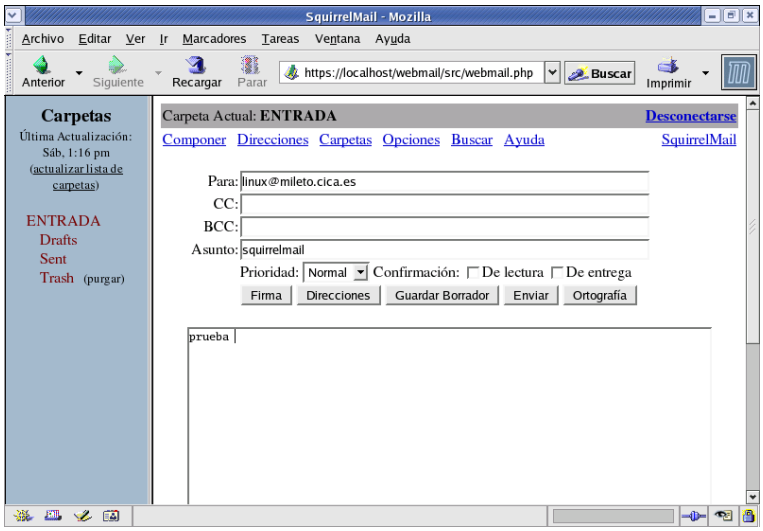
Comprobamos que las carpetas se sitúan a la izquierda y en la derecha tenemos en la parte superior el menú principal con las opciones: **Componer**, **Direcciones**, **Carpetas**, **Opciones**, **Buscar** y **Ayuda**. Si pulsamos sobre está última accederemos a la completa ayuda (en castellano) que nos permite conocer todos los aspectos relacionados con el uso del programa.



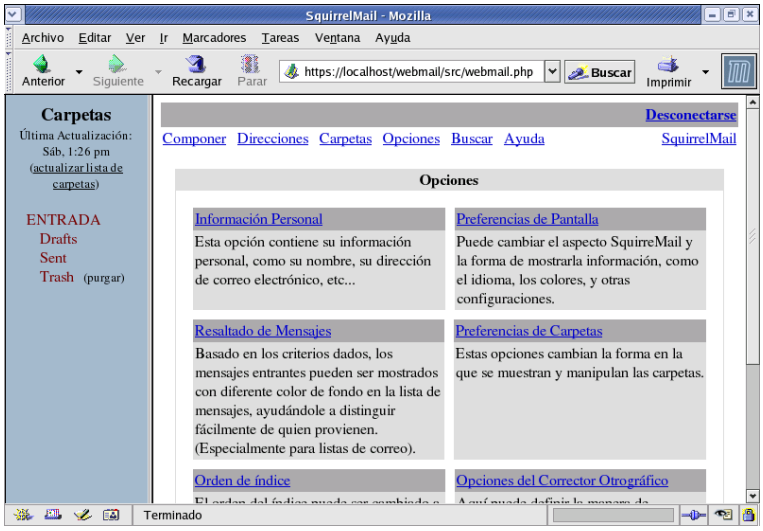
A partir de aquí, el cliente es muy intuitivo. Podemos enviar correo nuevo con la opción **Componer**:

57

`http://localhost/webmail`



Las opciones de personalización del cliente son variadas para ponerlo a nuestro gusto, y con la ventaja de que accedemos desde el lugar que queramos.



Prácticas

Tipo I

E3-I-1

Realizar correctamente todas las preguntas del cuestionario que para esta entrega se ha habilitado en Moodle:

1. ¿Qué comando hay que ejecutar en Debian para añadir el módulo de apache "comod-on"?

   a) a2enmod comod-on
   b) a2dismod comod-on
   c) apache2ctl comod-on
   d) a2ensite comod-on
   e) a2dissite comod-on

2. Por defecto, el nombre del subdirectorio que hay en cada $HOME de usuario para ubicar los archivos HTML que pueden ser servidos si activamos el módulo de apache (`mod_userdir` en Fedora, `userdir` en Debian) es

   a) public_html
   b) apache2-default
   c) www
   d) index.html
   e) userdir

3. En el fichero de configuración de apache hemos añadido la siguiente directiva para un directorio situado dentro del `DocumentRoot` de Apache.

   ```
   Options +Indexes +FollowSymlinks -ExecCGI
   ```

   Señala qué afirmación de las que siguen es falsa

   a) Si en él no hay `DirectoryIndex`, Apache devolverá un listado formateado del directorio.
   b) El servidor seguirá los enlaces simbólicos en ese directorio.
   c) Se permite ejecutar scripts cgi.
   d) La directiva `Options` se usa para establecer características para directorios.

4. Una Autoridad Certificadora, como tercera parte de confianza de usuarios en Internet, se encarga de firmar digitalmente:

   a) Las claves privadas de los usuarios de dicha CA

b) La clave pública de los usuarios junto con información de la identidad

c) Los certificados de las Autoridades de Registro

d) Las claves simétricas de los usuarios

5. Si en un servidor web, que sirve las páginas del dominio `www.curso.org`, se realiza la instalación por defecto de webalizer ¿en qué url se consultan las estadísticas?:

   a) `http://www.curso.org/webalizer/`

   b) `http://www.curso.org:8080/stats/`

   c) `http://www.curso.org:8080/usage/`

   d) `http://www.curso.org/usage/`

6. Si dirigimos un correo electrónico a la dirección `persona@organismo.org`, ¿a qué servidor se dirigirá el correo para su entrega al dominio `organismo.org`?

   a) Al servidor `mi.maquina.com`, si la respuesta del DNS es: `organismo.org. 1800 MX 10 mi.maquina.com`.

   b) Al servidor `mail.organismo.org`, porque esa es la convención

   c) Al servidor `organismo.org`

   d) Al servidor `persona.organismo.org`

7. Si en un servidor postfix queremos permitir el reenvío de correo a todas las máquinas de la LAN (10.1.1.0/24):

   a) Configuramos `mynetworks=10.1.1.0` siempre que `mynetworks_style=subclass`

   b) Configuramos `mynetworks=10.1.1.0/24` e ignoramos `mynetwork_style`

   c) Configuramos `mynetworks=10.1.1.1` y `mynetworks_style=subnet`

   d) Configuramos `mynetworks=10.1.1.1 - 10.1.1.254`

8. ¿Cuál de las siguientes respuestas es correcta?

   a) El protocolo LDAP permite recuperar los correos desde un agente de usuario

   b) El protocolo SMTP permite recuperar los correos desde un agente de usuario

   c) El protocolo POP permite enviar correos desde un agente de usuario

   d) El protocolo IMAP permite recuperar los correos desde un agente de usuario

9. Cuál es la secuencia correcta para analizar el spam (donde :xx representa el puerto TCP)

   a) origen - postfix - amavisd:1024 - spamassassin - amavisd - postfix:1025 - destino

   b) origen - postfix - spamassassin - amavisd - postfix:1025 - destino

   c) origen - postfix - amavisd:1024 - spamassassin - postfix:1025 - destino

   d) origen - amavisd - spamassassin - amavisd - postfix - destino

10. Para el gestor de listas de correo mailman, una url del tipo `http://midominio.org/cgi-bin/mailman/listinfo/listacurso` permite:

    a) Ver los archivos de mensajes de la lista

    b) Darse de alta y de baja en la lista "listacurso"

    c) Administar todos los parámetros de la lista

    d) Borrar la lista

E3-I-2 Apache

Pretendemos montar un servidor web en nuestro centro con control de acceso por grupos (serían los Dptos didácticos). Para ello necesitamos:

1. Montar dos hosts virtuales: `matematicas.micentro.org` y `lenguaje.micentro.org`
   a) El `DocumentRoot` de cada uno será de la forma `/var/www/matematicas` y `/var/www/-lenguaje`
      1) No es necesario especificar los cambios en los DNS, con hacerlo sobre `/etc/hosts` es suficiente.
   b) Controlar el acceso en base a grupos por medio de Apache. Concretamente, existirán dos grupos

      `mate` con usuarios *thales*, *mileto* y *pitagoras*

      `lengua` con usuarios *quevedo* y *cervantes*

      A la dirección `http://matematicas.micentro.org` solamente podrán conectarse los usuarios del grupo `mate` y a `http://lenguaje.micentro.org` solamente podrán hacerlo los usuarios del grupo `lengua`. A continuación damos algunas indicaciones sobre cómo conseguirlo:
      - Tenemos que crer un fichero de nombre `dptos` que puede estar situado en el mismo directorio en que guardemos el archivo de contraseñas de apache. Ese fichero debe contener dos líneas, en cada una definimos uno de los grupos de las práctica según se ejemplifica el analizar la directiva `AuthGroupFile`.
      - Cuando creemos las contraseñas de acceso, serán paran cada uno de los usuarios. Aunque trabajemos con un grupo el que se autentica es el usuario de ese grupo.
      - En el punto **??** de la entrega aparece un fichero de ejemplo para la autentificación. En este caso tener en cuenta que:
        - Debemos escribir de forma adecuada la directiva `AuthGroupFile` apuntando al fichero en el que hemos creado los dos grupos.
        - La directiva `Require` ahora será de la forma:

          ```
          Require group nombre_grupo
          ```

2. Mostrar una captura de la pantalla del navegador solicitando el nombre del usuario para entrar en la dirección `http://matematicas.micentro.org`

   El nombre del fichero será `e3-i-2.sxw`. Además de la captura, en el fichero debéis explicar los pasos que habéis seguido y ficheros utilizados para conseguir esta configuración.

Tipo II

E3-II-1 Servidor Seguro SSL

Instalar un servidor seguro usando SSL. El servidor debe responder a las peticiones `https://www.iesvirtual.org` La página principal debe mostrar "Servidor seguro SSL de www.iesvirtual.org"

Debe enviarse un fichero de nombre `e3-ii-1.sxw` que contenga una captura de la página principal de acceso a dicho servidor y algunas líneas del log de accesos.

E3-II-2 Squirrelmail

Esta práctica consiste en instalar la aplicación de correo web squirrelmail. Previamente, debéis haber realizado la instalación de apache y de un servidor de correo (Agente de transporte) en vuestro servidor. Una vez instalado debéis mandar a través de él un correo al usuario root de vuestro servidor con el asunto: "prueba de webmail". En el fichero `e3-ii-2.sxw` debéis enviar las capturas de pantalla de envío y recepción de dicho correo y un breve comentario sobre el agente de transporte y servidor imap utilizados, así como cualquier otro software adicional utilizado.

Bibliografía

[1] *Sendmail*, 2nd Edition, Bryan Costales & Eric Allman 2nd Edition January 1997 ISBN: 1-56592-222-0

[2] *TCP/IP Network Administration*, 2nd Edition, Craig Hunt 2nd Edition December 1997 ISBN: 1-56592-322-7

[3] *Guía de Administración de Redes con Linux* Olaf Kirch & Terry Dawson, Editado por O'Reilly (printed version) (c) 2000 O'Reilly & Associates Proyecto LuCAS por la traducción al español (c) 2002 HispaLiNUX

[4] `Bibliographyhttp://www.networkcomputing.com/unixworld/tutorial/008/008.txt.html`

[5] `http://www.catb.org/~esr/fetchmail/fetchmail-man.html`

[6] *How to set up SMTP AUTH* `http://www.jonfullmer.com/smtpauth/`

[7] *Proyecto de traducción de la documentación de Apache al español.*`http://quark.fe.up.pt/ApachES/`

[8] *The Apache Software Foundation*`http://www.apache.org/`

[9] *Servidor Apache.* Rich Bowen & Ken Coar. Prentice Hall

[10] *Servidor Apache 2.* Mohammed J. Kabir. Anaya Multimedia.

[11] Capítulos 9 y 10 del *Manual de referencia de Red Hat Linux* `http://europe.redhat.com/documentation/rhl9/rhl-rg-es-9/ch-httpd.php3`

[12] Capítulos 19 y 20 del *Manual de personalización de Red Hat Linux* `http://europe.redhat.com/documentation/rhl9/rhl-cg-es-9/`

Parte IV

Contenido dinámico

Capítulo 19

Páginas PHP

PHP, acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje "Open Source" interpretado de alto nivel, especialmente pensado para desarrollos web y el cual puede ser embebido en páginas HTML. La mayoría de su sintaxis es similar a C, Java y Perl, y es fácil de aprender. La meta de este lenguaje es permitir escribir a los creadores de páginas web, páginas dinámicas de una manera rápida y fácil, aunque se pueda hacer mucho más con PHP. `http://www.php.net/manual/es/preface.php`

19.1. Introducción

Hace algún tiempo[1], la información de la Web era relativamente estática. El cliente/navegador hacía una petición HTTP al servidor y éste le servía las páginas HTML que alguien había elaborado.

Este modelo se queda pequeño y tiene varios inconvenientes:

1. El mantenimiento de los enlaces y la información es tedioso. Cuando cambia algún dato, debemos recorrer todas las páginas para actualizar todos los lugares en los que aparece.

2. No enlaza con las bases de datos y aplicaciones que podamos tener en nuestra organización, con lo que perdemos una gran cantidad de información o lo que es peor, debemos duplicarla con el riesgo de desactualización.

3. El usuario se encuentra atado a la hora de seleccionar la información que desea. Solamente puede acceder a la información de la forma en la que ha previsto quien ha realizado la página web, pero no puede pedir por ejemplo que se ordene por edad, si está ordenada por apellidos.

En definitiva, podemos añadir más valor a la información si ésta es dinámica y se genera a petición del usuario.

Con el paso del tiempo, se han inventado varios métodos para conseguir que la información se genere de foma dinámica.

Los primeros en nacer fueron los *Server Side Includes* (SSI) y el interfaz *Common Gateway Interface* (CGI).

Una página SSI es una página HTML con comandos incluidos que se ejecutan en el servidor web. Por ejemplo, integrado con el código HTML se encuentra el SSI que nos detecta una cadena e imprimirá un valor dependiendo de ésta:

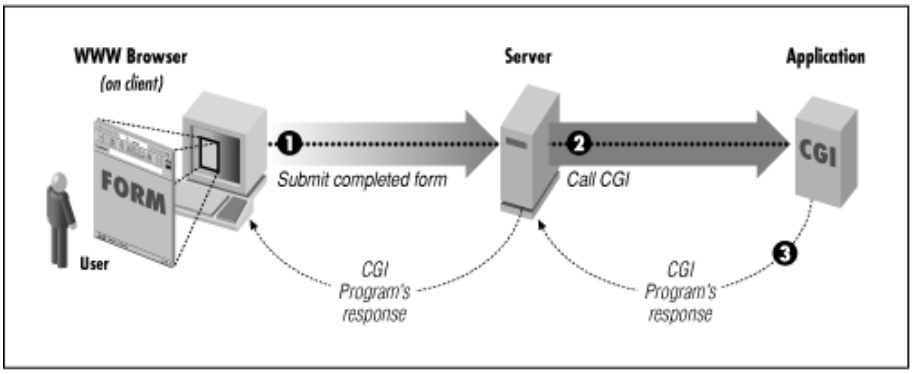
```
<html>
...
 <body>
```

[1] En la escala temporal de Internet, varios años dan para mucho.

```
...
<!--#if expr="\"$DOCUMENT_URI\" = \"/linux/file.html\"" -->
En Linux
<!--#elif expr="\"$DOCUMENT_URI\" = \"/windows/file.html\"" -->
En Windows
<!--#else -->
Ni en Windows ni en Linux
<!--#endif -->
...
</body>
</html>
```

Es un avance, pero su potencia es muy limitada y complejo para sitios medianos y grandes.

CGI es un mecanismo estándar de comunicación entre un programa o script[2] y un servidor web. La limitación es que deben comunicarse de una manera determinada y poco eficiente. La entrada del cliente web se pasa al programa a través del servidor web, según el mecanismo establecido. El programa recibe la información, la procesa y devuelve el resultado al servidor web, que compone la página definitiva. El mecanismo se presenta en la siguiente imagen:



Sin embargo, este mecanismo es muy rígido y al tener que invocar a un programa externo al servidor web, es lento y consume muchos recursos tanto de memoria como de proceso.

Existen otros mecanismos para generar contenido dinámico, como los servidores de aplicaciones Java[3], ColdFusion o ASP. Nuestro objetivo será conocer PHP, un lenguaje especialmente diseñado para generar páginas web dinámicas, y fácil de aprender y ejecutar.

Haciendo un poco de historia[4], PHP comenzó en 1994 cuando Rasmus Lerdorf quiso dar un paso más allá de los CGI. En los siguientes años evolucionó y comenzó a utilizarse en muchos sitios web. Un gran salto se produjo en 1997 cuando Zeev Suraski y Andi Gutmans lo reescribieron dando lugar a PHP 3.

En la actualidad disponemos de dos versiones estables de PHP, la 4 y la 5[5]. Ambas se basan en un motor denominado Zend (la 5 en Zend II). La principal diferencia entre ambas es que la versión 4 basa su programación en procedimientos y la 5, en cambio, está basada en objetos (POO `http://es.wikipedia.org/wiki/POO`). En esta entrega, así como el curso trabajaremos con la versión 4.

PHP no solamente podemos utilizarlo en sistemas Linux/Unix, sino que está disponible en multitud de plataformas, incluidas las Windows, siendo un gran competidor de ASP.

Datos recientes de Netcraft, estiman su uso en 18.455.683 dominios y 1.317.871 direcciones IP[6].

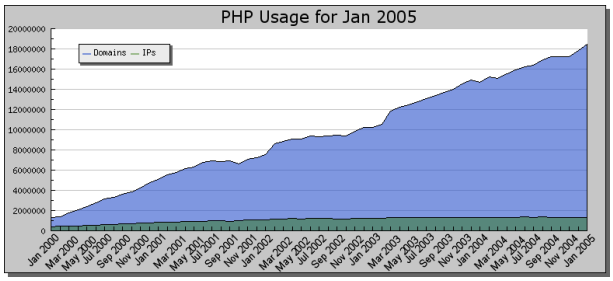
[2]Puede estar escrito en varios lenguajes: C, Perl...

[3]El estándar J2EE, está alcanzando su madurez y es adoptado por muchas empresas y organismos, pero su complejidad es alta.

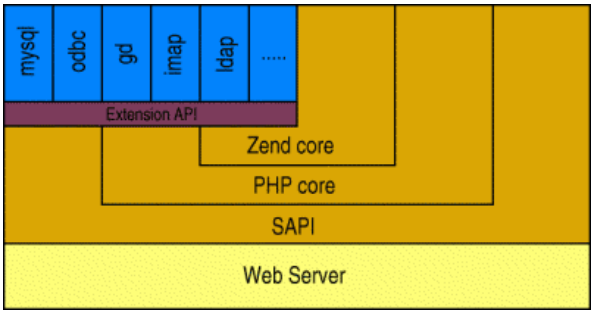
[4]Para conocer mejor esta historia se puede consultar `http://es.wikipedia.org/wiki/PHP`

[5]La primera versión estable vio la luz en julio de 2004

[6]Seguramente os llamen la atención estos datos. Como ya sabéis, una IP puede dar soporte a múltiples dominios virtuales.



La siguiente figura presenta la arquitectura de PHP. El interfaz SAPI permite integrarlo con la mayoría de los servidores web existentes. Por supuesto con Apache, pero también con IIS, Zeus, Netscape iPlanet, Java servlet, AOLServer, o Roxen.



Se pueden integrar módulos para conectarlo con bases de datos, servidores de directorios, servidores de correo...

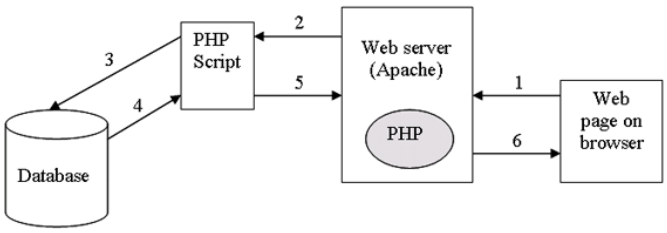
La sintaxis de PHP es similar a C, aunque hay características que ha tomado "prestadas" de Perl, C++ y Java.

PHP se ejecuta totalmente por el servidor web y al navegador le llega solamente código HTML. Aunque es compatible con lenguajes de script como javascript que se ejecutan en el navegador, con lo que podemos combinar ambas técnicas, si lo deseamos.

El siguiente código es una mezcla de HTML y PHP.

```
<html>
 <? echo date("M d, Y H:i:s", time()); ?>
</html>
```

El circuito que se forma viene dado por la siguiente figura. El navegador (1) solicita una página al servidor web. Éste ve que es una página con código PHP y la pasa al motor de PHP (2). Si es necesario, interacciona con bases de datos u otros módulos (3 y 4). Cuando la página está generada, se pasa al servidor web (5), que la traslada al cliente (6). Todo ello de una forma muy eficiente.



La salida HTML que obtendrá el navegador será la siguiente:

```
<html>
Apr 1, 2005 23:00:05
</html>
```

Resumiendo, PHP es un lenguaje de script que se ejecuta en el servidor web y permite que las páginas sean dinámicas. Nos permite conexiones a bases de datos y muchas utilidades más. La comparación con ASP de Microsoft es inmediata, pero, PHP es mucho mejor.

- El manual por excelencia: `http://www.php.net/manual/es/`
- Páginas interesantes:
  - `http://www.rinconastur.com/php/`
  - `http://jips.bankhacker.com/linux/apache/ssl/urlsphp.phtml`
  - `http://www.desarrolloweb.com/directorio/programacion/php/`

19.2. Instalación

Debian

Nos centraremos sólo en el módulo de php4 para Apache2[7]: `libapache2-mod-php4`

```
# apt-get install libapache2-mod-php4 php4-pear php4-cgi phpdoc
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes extras:
  php4-cli php4-common
Paquetes sugeridos:
  php4-dev
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-php4 php4-cgi php4-cli php4-common php4-pear phpdoc
0 actualizados, 6 se instalarán, 0 para eliminar y 119 no actualizados.
Necesito descargar 6876kB de archivos.
Se utilizarán 31,3MB de espacio de disco adicional después de desempa-
quetar.
¿Desea continuar? [S/n]
```

De los tres paquetes instalados, sólo el primero es imprescindible, los otros son:

`php4-pear` contiene archivos PEAR para php4. Esto es, numerosas clases para manejo más sencillo y limpio de, p.ej., bases de datos.

`php4-cgi` suministra CGI. Los módulos compilados son `bcmath`, `calendar`, `curl`, `dba`, `exif`, `filepro`, `ftp`, `mm`, `sockets`, `wddx`, `xml`, `yp` y `zlib`.

`phpdoc` documentación para PHP4.

Una vez instalado, reiniciamos el servidor con:

```
#apache2ctl restart
```

7

- Es el paquete "importante" ya que nos va a permitir trabajar con código php embebido en html. No podremos realizar scripts en línea de comandos con php, para eso es necesario instalar el intérprete "completo"

  ```
  # apt-get install php4
  ```

  pero en este caso, nos instalará a su vez Apache 1.3. Así que no os lo recomendamos.

Red Hat/Fedora

Necesitamos el paquete[8] `php`.

```
# apt-get install php
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
Se instalarán los siguientes paquetes extras:
  php-pear
Se instalarán los paquetes NUEVOS siguientes:
  php php-pear
0 upgraded, 2 newly installed, 0 removed and 178 not upgraded.
Need to get 1642kB of archives.
After unpacking 4652kB of additional disk space will be used.
Quiere continuar? [S/n]
```

y

```
#apachectl restart
```

Hola Mundo

Ya tenemos php instalado, ejecutemos nuestro primer script:

- Creemos un fichero de contenido

```
# cat prueba.php
 <?php
  //Esto es un comentario de una ílnea
  //Iniciamos un script php con <?php
  //Mostramos el texto encerrado entre comillas
  //con \n introducimos un retorno de ílnea
   echo "Hola Mundo \n";
  // La ófuncin getcwd() nos devuelve el directorio desde
  // donde se ejecuta
   echo getcwd(), "\n";
  // Fin del script
 ?>
```

- Ejecutemos el intérprete con[9]

```
# php -q prueba.php
Hola Mundo
/root
```

[8] Viene en los CDs de Fedora

[9]

- Con `-q` se suprimen las cabeceras HTTP:

```
Content-type: text/html
X-Powered-By: PHP/4.3.4
```

- Otra opción para ejecutarlo es dotar al archivo de los permisos adecuados (`chmod a+x prueba.php`) y añadir al fichero la línea

```
#!/usr/bin/php -q
```

En este caso sólo hay que escribir:

```
./prueba.php
```

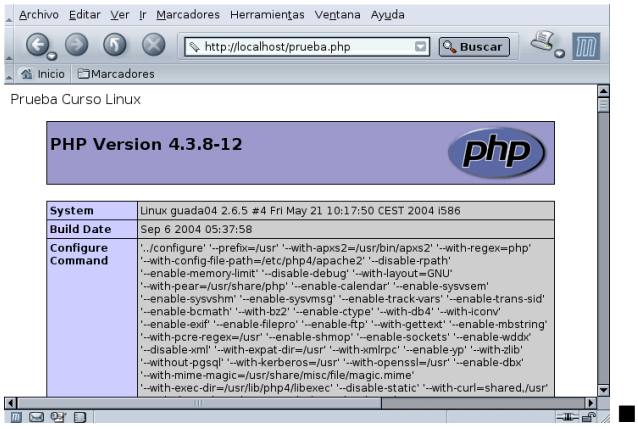
Pero no es la mejor forma de usarlo, ¿verdad?, veamos cómo usarlo dentro de una página web:

➥ **Para practicar:** Debian y Fedora

Podemos probar que nuestro servidor web funciona con el módulo PHP poniendo una página llamada `prueba.php` en el raíz del servidor Web con el contenido siguiente:

```
<html>
 <head>
   <title>PHP Test</title >
 </head>
 <body>
   <?php echo "Prueba Curso Linux<P>"; ?>
   <?php phpinfo(); ?>
 </body>
</html>
```

y apuntando nuestro navegador a `http:/localhost/prueba.php`



Merece la pena pararse un poco en la salida del fichero anterior. En ella se nos informa del estado del intérprete y del valor de las distintas directivas de configuración.

19.2.1. Configuración

php.ini

El fichero de configuración de php es

Fedora `/etc/php.ini`

Debian `/etc/php4/apache2/php.ini`

con él podemos controlar el comportamiento por defecto del intérprete. Nos centraremos en analizar el que se instala por defecto en Debian. A partir de él, comprender el de Fedora no presenta dificultad.

Las directivas de este fichero mantienen las convenciones en cuanto a la sintaxis:

- Las directivas de configuración son de la forma:

  `directiva = valor`

donde *valor* puede ser una cadena (`error_log=syslog`), un número (`precision=14`), una constante de PHP (`error_reporting=E_ALL`), una de las constantes[10] INI (`On`, `Off`, `True`, `False`, `Yes`, `No` y `None`) o una expresión de constantes `INI` operadas con `|` (`OR`), `&` (`AND`), `~` (`NOT`) , `!` (`FALSE`) o paréntesis ().

- Para introducir un comentario o comentar una directiva de configuración se le antepone un punto y coma.
- Los nombres de las directivas son sensibles a las mayúsculas y minúsculas.
- Las cabeceras de sección se indican encerrando el texto entre corchetes (por ejemplo `[MySQL]`).

No es necesario conocer todas las directivas de este fichero ni para escribir script PHP ni para seguir el resto de la entrega. El motivo de que se incluya es que os sirva de referencia por si hay que ajustar el intérprete a casos particulares. De hecho, casi con toda seguridad no tendremos que cambiar prácticamente ninguna directiva del fichero de configuración en un uso normal de PHP. Por tanto: desde este punto hasta la sección **Apache y php** (19.2.1 en la página 404) sólo habría que dar una lectura por encima para conocer las posibilidades de configuración.

Analicemos algunas directivas "importantes". El fichero se inicia con la sección principal de configuración del intérprete

```
[PHP]
```

en ella merece la pena comentar:

Directivas generales

`engine = On` activa el intérprete de PHP como módulo de Apache.

`short_open_tag = On` si está en `On`, activa la posibilidad de que los scripts php se delimiten usando etiquetas abreviadas `<? ... ?>`. Si es `Off` hay que usar `<?php ... ?>` o `<script>`.

`asp_tags = Off` si está en `On` permite el uso de etiquetas estilo ASP `<% ... %>`.

`precision = 12` número de dígitos significativos mostrados cuando trabajamos con números en coma flotante.

`y2k_compliance = On` hace cumplir la conformidad con el año 2000 (Causará problemas con navegadores que no cumplan con esto).

...

`allow_call_time_pass_reference = On` fuerza a que se pasen las variables de las funciones por referencia

`safe_mode = Off` si está en `On`, activa el modo seguro de PHP[11].

...

`;open_basedir =` limita las operaciones que php puede realizar sobre archivos a los directorios especificados (separados por `:`). Tal cual está, permite abrir todos los archivos. No depende del valor de `safe_mode`

[10] Para activar una directiva booleana es equivalente usar: 1, `On`, `True` o `Yes`.
Para desactivarla, podemos usar: 0, `Off`, `False` o `No`

[11] Para ampliar sobre este tema: `http://www.phpbuilder.com/manual2/manual/es/features.safe-mode.php`

`disable_functions =` permite desactivar determinadas funciones de PHP (separadas por ;). Tampoco depende de si está activo el modo seguro.

`;highlight.string = #DD0000` colores para resaltar la sintaxis de PHP. Los valores aceptables para la etiqueta `<font color="??????">` se pueden poner aquí.

`;highlight.comment = #FF9900`

`;highlight.keyword = #007700`

`;highlight.bg = #FFFFFF`

`;highlight.default = #0000BB`

`;highlight.html = #000000`

`expose_php = On` en `On` se incluye una cadena en la cabecera http del servidor que indica qué versión de PHP está instalada.

Límites de recursos

`max_execution_time = 30` tiempo máximo (en segundos) de ejecución de un script.

`max_input_time = 60` tiempo máximo (en segundos) que un script puede invertir en analizar los datos requeridos.

`memory_limit = 8M` tamaño máximo de memoria que puede consumir un script[12].

Gestión y registro de errores

Podemos trabajar con varios niveles de registro de errores con

`error_reporting = E_ALL & ~E_NOTICE` muestra todos los errores y avisos (`E_ALL`), excluyendo (`~`) las advertencias en tiempo de ejecución (`E_NOTICE`) del intérprete. Podemos usar los parámetros recogidos en la tabla:

Cuadro 19.1: Errores

Nombre	Significado
E_ALL	todos los errores y avisos
E_ERROR	errores fatales en tiempo de ejecución
E_WARNING	avisos en tiempo de ejecución
E_PARSE	errores en tiempo de compilación
E_NOTICE	advertencias en tiempo de ejecución
E_CORE_ERROR	errores fatales ocurridos el iniciarse PHP
E_CORE_WARNING	avisos ocurridos al iniciarse PHP
E_COMPILE_ERROR	errores fatales en tiempo de compilación
E_COMPILE_WARNING	avisos en tiempo de compilación
E_USER_ERROR	mensajes de error generados por el usuario
E_USER_WARNING	avisos de error generados por el usuario
E_USER_NOTICE	advertencia generada por el usuario

↬ Ejemplo

```
error_reporting = E_COMPILE_ERROR|E_ERROR|E_CORE_ERROR
```

[12] Si se piensa instalar Moodle (ya lo conocéis, ¿verdad?) es mejor poner este valor a 16M.

muestra sólo errores.

`display_errors = On` en `On` muestra los errores en pantalla como parte de la salida HTML generada.

`display_startup_errors = Off` si se cambia a `On` muestra los errores al iniciar el intérprete. No le afecta que la anterior esté en `On`.

`log_errors = Off` no guardamos los log de error de PHP en un archivo de registro del servidor.

`log_errors_max_len = 1024` tamaño máximo del archivo `error_log`. Por defecto es de 1024. Si se pone a 0 no se impone restricción de tamaño.

Manipulación de datos

`;arg_separator.output = "&"` separador usado por PHP para separar argumentos en las urls generadas. Por defecto es `&`.

`;arg_separator.input = ";&"` lista de separadores usados por PHP para analizar una URL y obtener las variables pasadas en ella. Por defecto es `&`.

`variables_order = "EGPCS"` establecemos el orden (de izquierda a derecha) en que PHP registra las variables: **G**ET, **P**OST, **C**ookie, **E**ntorno[13] y variables internas (**S**). Tal cual está, sería: Entorno, Get, Post, Cookie y Server.

`register_globals = Off` si se mantiene en `Off` impedimos que se creen automáticamente variables globales con los nombres de las variables pasadas como argumentos al script. Por defecto está desactivada.

`register_argc_argv = On` le dice a PHP que declare las variables[14] `argv` y `argc`.

`post_max_size = 8M` tamaño máximo permitido de los datos usando el método POST

`magic_quotes_gpc = On` el estar en `On`, añade una barra invertida antes de la comilla sencilla (’), comilla doble (“), barra invertida (\) y los NUL en operaciones entrantes GET, POST o Cookie.

`magic_quotes_runtime = Off` si se pone en `On`, se activa la conversión automática de comillas para los datos generados en tiempo de ejecución (datos generados por una base datos externa, lectura de archivos de texto, etc).

`auto_prepend_file =` permite añadir el archivo especificado al final de todo fichero php ejecutado. Se incluye como si fuese invocado por una llamada mediante la función `include()`. Si se ponen a `none,` se desactiva la adición automática de archivos.

`auto_append_file =` permite añadir el archivo especificado al principio de todo fichero php ejecutado. Se incluye como si fuese invocado por una llamada mediante la función `include()`.Si se ponen a `none,` se desactiva la adición automática de archivos.

`default_mimetype = "text/html"` tipo mime por defecto para la salida de datos.

`;default_charset = "iso-8859-1"` juego de caracteres usado por defecto.

`;always_populate_raw_post_data = On` si está en `On` se crea la variable `$HTTP_RAW_POST_DATA` aunque el tipo MIME sea conocido (por defecto sólo se crea cuando es desconocido).

[13] *Environment*

[14]

`argv` array de argumentos pasados en el script

`argc` número de argumentos (parámeros) pasados en el script

Rutas y directorios

`;include_path = ".:\php\includes"` lista de directorios (separados por :) en los que las funciones `require()`, `include()` y `fopen_with_path()` buscarán los archivos requeridos. Por defecto es el directorio "actual" (.).

`doc_root =` el raíz desde donde se sirven las páginas. Usado si no está vacío.

`user_dir =` directorio desde el que php abre los scripts para un usuario (`/~usuario`). Si está vacío, no se usa.

`extension_dir =/usr/lip/php4` directorio desde donde cargar las extensiones dinámicas de los módulos.

`enable_dl = On` activa la posibilidad de emplear la carga dinámica de extensiones de php, útil si se trabaja con PHP como módulo de Apache.

`;cgi.force_redirect = 1` necesario para proporcionar seguridad cuando ejecutamos PHP como un CGI. Por defecto está en `on`. Puedes ponerlo en `off` "bajo tu responsabilidad" (para IIS sí puede estar en `off`).

`;cgi.redirect_status_env = ;` si el anterior está en `on` y no estamos trabajando bajo servidores web Apache o Netscape, podemos necesitar un nombre de variable de entorno donde PHP comprobaría si está `OK` para continuar la ejecución. Puede originar problemas de seguridad.

`;cgi.rfc2616_headers = 0` informa a PHP del tipo de encabezados a usar cuando envía código de respuesta HTTP. Si está a 0 (valor por defecto) envía *Status*: encabezado soportado por Apache. Si está a 1, PHP enviará un encabezado RCF2616.

Subir ficheros

`file_uploads = On` permite subir archivos HTTP

`upload_tmp_dir =` directorio temporal para los archivos HTTP que hemos subido (si no se especifica, usará el del sistema por defecto)

`upload_max_filesize = 2M` tamaño máximo permitido a los archivos a subir

Directivas relacionadas con fopen

`allow_url_fopen = On` permite tratar URLs (como http:// o ftp://) como archivos.

`;from="john@doe.com"` define la contraseña para ftp anónimo (su e-mail)

`;user_agent="PHP"` define la cadena `User-Agent`

`default_socket_timeout = 60` tiempo por defecto para la disponibilidad de socket (en segundos)

Extensiones dinámicas

`;extension=modulename.extension` debemos usarlas si deseamos cargar extensiones automáticamente. Por ejemplo:

`;extension=msql.so` sólo debe estar el nombre del módulo, no es necesaria ninguna información del directorio, ya que debemos especificar la localización de la extensión con la directiva `extension_dir`. **Nota:** para Red Hat/Fedora la extensión de paquete de los módulos se carga a través de los archivos `ini` en el directorio `/etc/php.d`

Configuración de módulos de PHP

Sólo vamos a comentar las directamente relacionadas con el curso:

`[Syslog]`

`define_syslog_variables = Off` para definir o no las diversas variables `syslog` (ej. `$LOG_PID`, `$LOG_CRON`). Es una buena idea ponerlo en `off`. En tiempo de ejecución podemos definir estas variables con la llamada `define_syslog_variables()`.

`[MySQL]`

`mysql.allow_persistent = On` permite o previene enlaces persistentes.

`mysql.allow_persistent = -1` número máximo de enlaces persistentes (-1 equivale a: sin límite).

`mysql.max_links = -1` número máximo de enlaces (+número persistentes, -1 es sin límite)

`mysql.default_port =` número de puerto por defecto para `mysql_connect()`. Si no se determina, usará `$MYSQL_TCP_PORT` o la entrada en `/etc/services` o el valor de tiempo de compilación definido en `MYSQL_PORT` (en ese orden).

`mysql.default_socket =` nombre de socket por defecto para las conexiones locales MySQL. Si no se especifica, usa el que tiene por defecto MySQL.

`mysql.default_host =` host por defecto para `mysql_connect()` (no se aplica en modo "safe")

`mysql.default_user =` usuario por defecto para `mysql_connect()` (no se aplica en modo "safe")

`[bcmath]`

`bcmath.scale = 0` número de cifras decimales para todas las funciones `bcmath`.

`[browscap]`

`;browscap = extra/browscap.ini` contiene información sobre las cadenas de identificación que usa cada navegador.

`[Session]`

`session.save_handler = files` manipulador usado para guardar/recuperar datos.

`;session.save_path = "N;/path"` argumento pasado a `save_handler`. En el caso de archivos, éste es el path donde se guardan los archivos de datos. Desde PHP 4.0.1 se define como "`N;/path`" donde `N` es un entero. En lugar de guardar todos los archivos de sesión en `/path` guardará los datos en subdirectorios de `N` niveles de profundidad (útil para servidores que manejan gran cantidad de sesiones).
Nota: Php no crea esta estructura de directorios automáticamente. Para esto se puede usar el script en ext/session dir. En Fedora está descomentada y vale `/var/lib/php/-session`

`;session.save_path = /var/lib/php4`

`session.use_cookies = 1` para usar cookies

`session.use_only_cookies = 1` esta opción permite a los administradores proteger a los usuarios de ataques derivados de pasar la identidad de sesión en URLs, por defecto es 0.

`session.name = PHPSESSID` nombre de la sesión (usado como nombre de cookie).

`session.auto_start = 0` inicia la sesión ante peticiones de arranque.

`session.cookie_lifetime = 0` tiempo de vida, en segundos, de las cookies o, si es 0, hasta que el navegador es reiniciado.

`session.cookie_path = /` path para el que es válida la cookie.

`session.cookie_domain =` dominio para el que la cookie es válida.

`session.serialize_handler = php` manipulador para serializar los datos (php es el estándar para PHP).

`;session.gc_probability = 1` probabilidad de que el proceso de "recolección de basura" comience en cada inicialización de sesión. Se calcula usando `gc_probability/gc_divisor` (ej. 1/100 significa que hay un 1 % de posibilidades de que el proceso arranque en cada petición).

`session.gc_divisor = 100`

`session.gc_maxlifetime = 1440` tiempo en segundos tras el que los datos pueden ser limpiados por el proceso de recolección de basura. **Nota**: si está usando la opción subdirectory para grabar los archivos de sesión la recolección de basura no se hará de forma automática, necesitará un shell script, entrada `cron` u otro método.

`session.bug_compat_42 = 1`

`session.bug_compat_warm = 1` PHP 4.2 y versiones anteriores tienen un bug no documentado que permite inicializar una variable de sesión de alcance global aunque `register_globals` está deshabilitado. PHP 4.3 y posteriores avisan si se usa esta característica. Puede deshabilitar la característica y el aviso de forma separada. Por ahora, el aviso sólo aparece si `bug_compat_42` está habilitado.

`session.referer_check =` chequea HTTP Referer para invalidar las URLs externas que contienen identidades.

`session.entropy_length = 0` bytes a leer desde el archivo.

`session.entropy_file =` para especificar la session id.

`session.cache_limiter = nocache` poner en {`nocache`, `private`, `public`,} para determinar los aspectos de caché HTTP, o dejarlo vacío para permitir enviar encabezados anti-caché.

`session.cache_expire = 180` el documento expira después de `n` minutos.

`session.use_trans_sid = 0` desactivado por defecto. Su uso puede poner en peligro la seguridad de sus usuarios. Se debe usar esta opción con precaución.

Por último, comentar que podemos enviar correo a través de php.

Apache y php

La interacción entre Apache y PHP se configura a través de:

Fedora `/etc/httpd/conf.d/php.conf` en él hay una serie de directivas que tienen que ver con su instalación, y que comentaremos. En la siguiente línea se carga el módulo de php con la orden `LoadModule`:

```
LoadModule php4_module        modules/libphp4.so
```

Con

```
AddType application/x-httpd-php .php
```

conseguimos que los archivos de extensión `.php` sean manejados por el intérprete de PHP, y con la siguiente directiva, decimos qué páginas de un directorio[15] pueden ser consideradas de inicio (si no se especifica una página concreta). En este caso: `index.php` se añade a la lista que ya existiera (`index.html, index.htm, index.shtml`...).

```
DirectoryIndex index.php
```

Debian el cometido del fichero anterior lo comparten los archivos:

```
/etc/apache2/mods-available/php4.conf
/etc/apache2/mods-available/php4.load
```

y los enlaces simbólicos a ellos (para que se active) del directorio `/etc/apache2/mods-enabled/`

Con `php4.load` cargamos el módulo y con el fichero

```
$cat /etc/apache2/mods-available/php4.conf
AddType application/x-httpd-php .php
AddType application/x-httpd-php-source .phps
```

conseguimos que los archivos de extensiones `.php` o `.phps` sean manejados por el intérprete de PHP.

➥ **Para practicar**

Un ejemplo que simplemente esboza lo que podemos hacer con PHP es el siguiente, formado por los ficheros `prueba.html` y `accion.php`

```
$cat prueba.html
 <html>
  <body>
   <form action="accion.php" method="POST">
    Su nombre: <input type=text name=nombre>
    Su edad: <input type=text name=edad>
    <input type=submit>
   </form>
  </body>
 <html>
```

```
$cat accion.php
Hola <?php echo $nombre?>. Tienes <?php echo $edad?>años.
```

Para poder trabajar con PHP tal cual aparece en el ejemplo hay que modificar una directiva del fichero de configuración de php: `php.ini`. Esta directiva cambió su valor por defecto a partir de la versión 4.2.0 de php (con versiones anteriores estaba en ON). Se trata de hacer que `register_globals = ON`, además hay que reiniciar apache:

Debian `apache2ctl restart`

Fedora `apachectl restart`

para que los cambios sean efectivos. Si se mantiene en OFF impedimos que se creen variables globales con los nombres de las variables pasadas como argumentos al script.

Si ponemos esos ficheros en el raíz del servidor Web y desde el navegador llamamos a `prueba.html` veremos el resultado de mezclar formularios y php. ■

[15] Cuando especificamos un directorio sin decir la página concreta (`http://localhost` o `http://localhost/pruebas`, siendo el primer caso el directorio raíz de los documentos y el segundo, un directorio llamado pruebas, dentro de él)

19.3. Primeros pasos con php

En este apartado, vamos a introducir algunos conceptos básicos de PHP de forma resumida, ya que este apartado por sí sólo podría ser un curso.

PHP y HTML

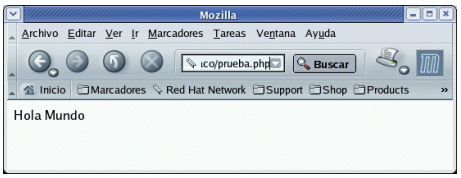
Como ya hemos visto en los ejemplos anteriores el modo estándar[16] de incrustar código PHP en html es

```
<?php ... ?>
```

Por ejemplo con

```
<html>
<body>
  <?php echo "Hola Mundo"; ?>
</body>
</html>
```

obtendríamos de salida el texto encerrado entre comillas (es que el dichoso `Hola Mundo` no puede faltar)



Como es el servidor el que ejecuta el código PHP, el usuario nunca llega a ver el código PHP, solamente el resultado en formato HTML, como podremos observar si en el navegador seleccionamos la opción de "Ver código fuente".

- Como ya hemos dicho, el código PHP se inserta en las páginas HTML entre las etiquetas `<?php ?>`; para insertar instrucciones HTML dentro de estas etiquetas se utiliza la instrucción `echo`. Hace que la cadena entrecomillada (`echo "Hola Mundo";`) o variable (`echo $variable;`) escrita a continuación se escriba en el documento HTML.
- Las instruciones de PHP se cierran con un punto y coma[17]. Uno de los errores más comunes (y que más dolores de cabeza da) es que se nos olvide poner el punto y coma final.

[16] En realidad existen cuatro estilos diferentes de etiquetas para hacerlo:

Estilo XML el que hemos comentado.

Estilo corto `<? ... ?>`

Estilo script `<script language='php'>... </script>`

Estilo ASP `<% ...%>`

[17] Si el código PHP consta de una sola instrucción, no es necesario ponerlo.

Cuadro 19.2: Tipos de datos

Tipo	Descripción
Integer	Números enteros
Double	Números en coma flotante
String	Cadenas de caracteres
Boolean	valor lógico: TRUE o FALSE
Array	Matrices de datos del mismo tipo
Object	Tipo especial de dato complejo

Comentarios

Disponemos de varias formas de introducir comentarios que aclaren el código, se trata de

```
// Esto es un comentario de una sola línea

# este es otro comentario de una sola línea
/*Comentario de varias lineas,
varias líneas,
varias líneas */
```

Variables

Se pueden definir las variables como puntos de almacenamiento en la memoria del ordenador. Durante la "vida útil" de una variable siempre se puede acceder a los valores que ésta almacena, siendo posible modificarlos.

Con PHP no es necesario declarar las variables (aunque es conveniente hacerlo), cuando se utiliza una variable por primera vez, ésta se considera declarada. El nombre de cualquier variable ha de empezar siempre con un signo de dólar (`$`). Además, la primera letra del nombre de la variable no puede ser un número.

PHP soporta varios tipos de datos (véase la tabla 19.2) que se reconocen y asignan automáticamente. Los básicos son: `string` (texto), `integer` (número entero) y `double` (número de coma flotante).

```
$cadena="Hola Mundo"; //string
$entero = 8; //entero
$decimal = 8.0; #float
$decimal = $entero; //ahora la variable decimal es un entero
```

Cuidado, que PHP discrimina entre mayúsculas y minúsculas en los identificadores de las variables. De esa forma, las dos variables siguientes son diferentes:

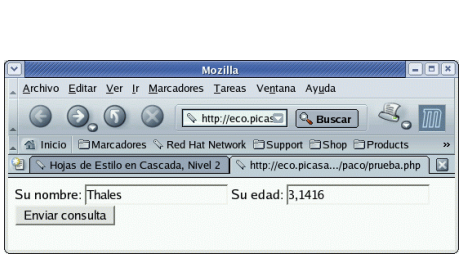
```
$mivariable=1;
$Mivariable='uno';
```

Se pueden concatenar *strings* y variables, bien escribiéndolas directamente unas a continuación de otras o, mediante el operador de concatenación que es el punto (`.`). Es más aconsejable este último método.

```
echo $cadena.' cruel';
```

¿Qué se obtendría?

Variables de formularios Si la variable `register_globals` del archivo de configuración de PHP está en `On`, desde PHP tendremos acceso directo a los valores introducidos en los formularios de una página Web (véase la práctica en la página 405). El resultado que obtenemos al ejecutar esa práctica es:



Es decir, hemos pasado al script `accion.php,` de forma directa, las variables `$nombre` y `$edad`, que provienen de los campos de tipo texto del formulario definido en el fichero `prueba`.

Como ya comentábamos en el ejemplo, para que eso sea posible, hemos tenido que modificar el fichero de configuración de PHP. Pero podemos hacerlo de otra forma.

Hay tres formas de acceder a las variables de un formulario:

- Estilo corto, es el comentado antes. Para eso es necesario poner `register_globals` en[18] `On.`
- Estilo intermedio[19]. Surge a partir de la versión 4.1 de PHP. Consiste en recuperar las variables del formulario en alguna de las matrices `$_POST` (si hemos usado el método POST), `$_GET` (si lo enviamos por el método GET) y `$_REQUEST` (para ambos).

  ↬ Con este método el fichero `accion.php` sería de la forma:

  ```
  Hola <?php echo $_POST['nombre'] ?>.  Tienes <?php echo
  $_POST['edad']?> años.
  ```

- Estilo largo. Es el más "engorroso" pero funciona siempre. En este caso, las variables del formulario las podemos recuperar a través de las matrices `$HTTP_POST_VARS` y `$HTTP_GET_VARS`

  ↬ Con este método el fichero `accion.php` sería de la forma:

  ```
  Hola <?php echo $HTTP_POST_VARS['nombre'] ?>.  Tienes <?php echo
  $HTTP_POST_VARS['edad']?> años.
  ```

Arrays

"Un array es un conjunto de variables o registros del mismo tipo que puede estar almacenados en memoria principal o en memoria auxiliar. Los arrays de 1 dimensión se denominan también vectores, los de 2 o más dimensiones se denominan matrices. La forma de acceder a los elementos del array es directo, es decir, el elemento deseado es obtenido a partir de su índice" `http://es.wikipedia.org/wiki/Array`

A "grosso modo", y sin entrar en más tecnicismos, podemos entender un array (table o matriz) como un nombre que permite almacenar varios datos del mismo tipo, que son accesibles usando el lugar que ocupan (comenzando desde el cero) o mediante una clave que los identifica (son los arrays asociativos). En la página 418 vemos más ejemplos sobre su uso.

↬ Por ejemplo,

- Un array que almacena los días de la semana se puede definir con

[18] En los dos que siguen no es necesario hacerlo.
[19] Es con el que trabajaremos en el ejemplo en la página 413.

```
$semana= array("Lunes","Martes","Miércoles","Jueves","Sábado","Domingo")
```

o equivalentemente,

```
$semana[0]="Lunes";
$semana[1]="Martes";
$semana[2]="Miércoles";
$semana[3]="Jueves";
$semana[4]="Viernes";
$semana[5]="Sábado";
$semana[6]="Domingo";
```

Por ejemplo, para visualizar el 4º dato almacenado escribiremos

```
echo $semana[3]
```

- Podemos usar un array asociativo para almacenar los datos de una persona[20]:

```
$alumno = array(nombre=>"Thales", apellido=>"de Mile-
to", edad=>"16");
```

y accederemos a los datos con:

```
echo $alumno["nombre"];
echo $alumno["apellido"];
echo $alumno["edad"];
```

Constantes

Una constante (como su nombre indica) no puede modificar su valor, se definen usando la palabra *define*

```
define['iva',16];
```

Operadores

Para conocer los operadores de PHP, os remitimos a la información disponible en la página `http://www.php.net/manual/es/language.operators.php`. Son los habituales en cualquier lenguaje de programación. Aquí tenéis un pequeño ejemplo de algunos de ellos.

```
$numero = 8 + 2; //tomaría el valor 10
$cuadrado = $numero * $numero; //valdría 100
$numero += 1; //equivale a $numero = $numero + 1;
```

Funciones

PHP trae una gran cantidad de funciones que nos pueden facilitar de forma considerable la realización de scripts (véase `http://es2.php.net/manual/es/funcref.php`). Además, podemos definir funciones. La idea de función surge de la necesidad de no repetir el mismo código[21] varias veces. La forma general de definir una función es:

```
<?php
 function foo ($arg_1, $arg_2, ..., $arg_n)
  {
    echo "Función de ejemplo.\n";
```

[20] Es equivalente a usar:

```
$alumno[nombre]="Thales";
```

y así, con el resto de valores

[21] También para facilitar su legibilidad

```
    return $retval;
  }
?>
```

↪ Por ejemplo, veamos una función que nos calcula el `iva` del valor pasado como argumento[22]:

```
function iva ($cantidad, $iva)
{
   //cálculo del iva, el valor del iva se pasa en%
   return $cantidad*$iva/100;
}
```

Para ejecutarla sólo hemos de llamarla con los parámetros adecuados, por ejemplo, con

```
echo iva(50,16);
```

obtendríamos de salida 8.

Básicamente, PHP permite trabajar con dos tipos de variables: *locales* y *globales*. Una variable es *local* cuando su uso está restringido al ámbito[23] de una función, es decir, sólo son accesibles dentro del cuerpo de la función. Una variable global es visible en toda la secuencia de comandos. Las variables globales se declaran anteponiendo la palabra `global`. Por ejemplo:

```
global $mysql_server;
```

↪ En la sección 20.5.2 en la página 453 sobre PHP y MySQL hay un script en el que es necesario usar estas últimas (en él, además, hacemos uso de un par de funciones).

Reutilización de código

PHP dispone de dos instrucciones que permiten la reutilización del código, se trata de

`include` cuando el intérprete llega a esta instrucción, sustituye el nombre del fichero que aparece entrecomillado por el código contenido en él. Existe otra instrucción parecida, que es `require`.

`require` se evalúa antes de procesarse el fichero. Es decir, PHP recorre el fichero antes de ejecutarlo, y donde encuentra un `require`, lo sustituye por el fichero "requerido" en cuestión, ejecutándose luego el código. `Include` en cambio, sólo se sustituye cuando el intérprete llega a la instrucción.

↪ En la sección 19.4.1 en la página 413 veremos un ejemplo sobre su uso.

19.3.1. Estructuras condicionales

La instrución If

Se utiliza para tomar una decisión. La sintaxis es la siguiente:

```
if (condicion) {comandos_si_verdadera_condicion;}
else {comandos_falsa_condicion;}
```

Se evalúa la condición y en caso de ser verdadera, se ejecutan `comandos_si_verdadera_condicion`. Si la condición no es verdadera y, existe la claúsula `else`, se ejecutarían los `comandos_falsa_condicion`. Por ejemplo[24]:

[22] En PHP5 todas las variables se pasan por referencia. Para conseguir que sea así en PHP4, hay que añadirles el símbolo `&`

[23] Con ámbito nos referimos a las zonas de código en las que es visible (y utilizable) una variable dada.

[24] Se analiza un ejemplo un poco más complejo en la página 417.

```
if ($numero==10) && ($nombre=="pepe") {echo 'Pepe tiene 10 años' }
```

 Un par de notas:

- Se pueden anidar.
- También existe la estructura `if` y `elseif`:

```
if(condición){si ocurre condición;}
elseif(otracondicion){si ocurre otracondición;}
else{si no ocurre ninguna de las dos anteriores;}
```

- Notar que las expresiones relacionadas con estructuras `if`, funciones o bucles se suelen agrupar mediante un par de llaves `{}`.

switch

Permite que se ejecute un bloque de instrucciones en función de un valor que tome una expresión. Es similar a una instrucción `if.` pero permitiendo que la condición tome más de dos valores.

La sintaxis es la siguiente:

```
switch (expresión)
  {
  case resultado1:
    instrucciones;
    break;
  case resultado2:
    instrucciones2;
    break;
    ....
  default:
    instrucciones por defecto;
  }
```

La sentencia `switch` ejecutará selectivamente los comandos que correspondan con la primera coincidencia. Esto se ve mejor con un sencillo ejemplo:

Si el valor de `$vehiculo` coincide con automóvil, se ejecutará el comando `echo "cuatro ruedas"` e irá al final. El otro patrón es similar, pero en el caso de que no coincida con ninguno, se ejecutarán los comandos de la entrada `default`.

```
echo "El ".$vehiculo." tiene ";
switch ($vehiculo)
{
case 'automovil':
  echo "cuatro ruedas";
  break;
case 'motocicleta':
  echo "dos ruedas";
  break;
default:
  echo "no es un vehículo válido";
}
```

19.3.2. Bucles

for (y foreach)

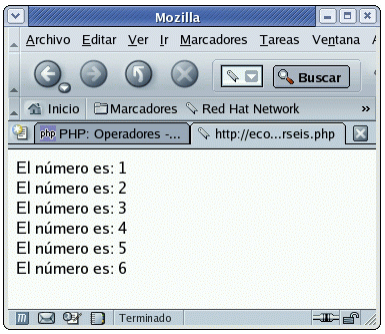
Su sintaxis es:

```
for (inicialización; condición; actualización)
  {
    instrucciones;
  }
```

Su funcionamiento es como sigue: se evalúa la expresión *inicialización* (en este parámetro se suele establecer el valor inicial del contador). Después se evalúa *condición*. Si el resultado es falso, se abandona el bucle. Si es cierto, se ejecuta el bloque de `instrucciones` y se evalúa la expresión `actualización` (en esta expresión se suele ajustar el valor del contador).

↬ Por ejemplo

```
<?php
 for ($i = 1; $i <=6; $i++ )
 {
  echo "El n&uacute;mero es: ".$i."<br>";
 }
?>
```



La sentencia `foreach` permite recorrer todos los elementos de una matriz de una forma sencilla, su sintaxis es

```
foreach (NombreArray as Variable)
  {
    instrucciones;
  }
```

se puede ver un ejemplo de su uso en la página 418.

while y do...while

Usando un bucle while podemos ejecutar un bloque de instrucciones, mientras que la condición sea verdadera. Se usan cuando no se sabe de antemano el número de iteraciones que se deben ejecutar, sino bajo qué condición se ejecutan. La sintaxis básica es

```
while (condición) {
instrucciones;
}
```

↬ El mismo ejemplo de la instrucción `for` quedaría

```
<?php
$i=1;
while ($i <=6)
{
  echo "El n&uacute;mero es: ".$i."<br>";
  $i++;
}
?>
```

El bucle `do...while` es similar al generado con la instrucción `while`. La diferencia estriba en que si la condición es falsa, con el bucle `while` no ejecutamos ninguna instrucción, mientras que con el `do...while` siempre se ejecuta al menos una vez.

↪ De nuevo el ejemplo anterior, pero con `do...while`

```
<?php
$i=1;
do
{
  echo "El n&uacute;mero es: ".$i."<br>";
  $i++;
}while ($i <=6);
?>
```

19.4. Ejemplos

19.4.1. Un ejemplo de Web con PHP

Como hemos comentado en la introducción, PHP es un sencillo lenguaje de secuencias de comandos. Normalmente se escribe directamente en las páginas HTML y es el servidor Web quien lo interpreta.

Para ver de una forma muy sencilla algunas de las cosas que se pueden hacer con PHP analizaremos el código de los siguientes ejemplos que componen un sitio web[25]:

- `index.php`
- `inicio.php`
- `pagina1.php`
- `pagina2.php`
- `cabecera.php`
- `menu-izq.php`

Además, vamos a usar la siguiente hoja de estilo (`estilo.css`) que se lista a continuación.

```
H2 {text-align: center}
H3 {text-align: center}
#cabecera{      margin-bottom: 10px;
                background-color: #000099;
                color: #ffcc99;
                padding: 3px;
                text-align: center;
                border-bottom-style: solid;
                border-top-style: solid;
        }

#izquierda{     padding: 5px;
                margin: 0px;
```

[25] Pondremos todos los ficheros en el servidor, en un fichero comprimido de nombre `ejemplo-php.tgz`

```
                width: 10%;
15              float: left;
                border-color: #000099;
                border-style: solid;
                border-width: 1px;
                background-color: #ffcc99;
20      }

#contenido{     padding: 10px;
                margin-left: 8%;
                float: left;
25              width: 70%;
                border-color: #000099;
                border-style: solid;
                border-width: 1px;
        }
```

Listado 19.1: estilo.css

Su interpretación no es demasiado compleja. Con ella redefinimos los encabezados de nivel 2 y 3 para que nos centre el texto y definimos tres estilos que nos van a permitir que las tres partes (cabecera, izquierda y derecha) que constituyen nuestra Web, tengan el aspecto deseado.

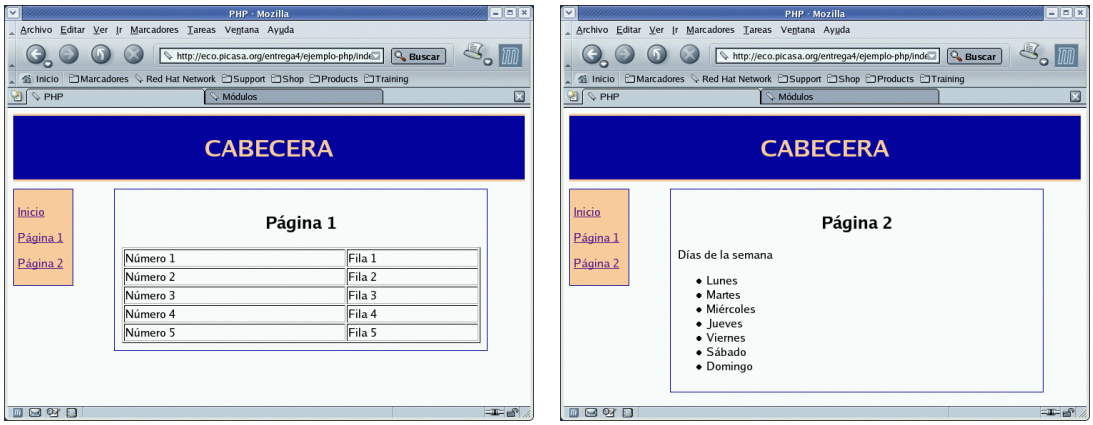
Cabecera	
Izquierda	Contenido

Si no conocemos las hojas de estilo y estamos interesados en ellas, en internet existen multitud de manuales que nos pueden ayudar sobre este tema. Una de las más interesantes es la Web de la especificación de las hojas de estilo `http://www.sidar.org/recur/desdi/traduc/es/css/cover.html`

Antes de analizar el código PHP de estas páginas, observaremos la página de inicio (`index.php`) a través de un navegador web donde podemos ver las distintas partes que componen la página Web y que se podrían reducir a:

- cabecera
- menú lateral izquierdo
- contenido de la página

Si accedemos a los distintos enlaces del sitio desde el menú lateral izquierdo, podemos observar que tanto éste, la cabecera y el estilo se mantienen, variando sólo el contenido central de las distintas páginas.



Estaría muy bien, por lo tanto, no tener que escribir en cada página los elementos que se repiten y escribir sólo lo que varía, que es el contenido central. Para ello podríamos pensar en generar una especie de esqueleto e insertar en él los distintos elementos de la página. Lo podemos hacer creando una página llamada `index.php`.

El código de las páginas php es el siguiente:

```
<html>
   <head>
      <title>PHP</title>
      <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1
          ">
      <link rel="stylesheet" type="text/css" href="estilo.css">
   </head>
   <body>
      <div id="cabecera"><?php include "cabecera.php"; ?></div>

      <div id="izquierda"><?php include "menu-izq.php"; ?></div>

      <div id="contenido">
         <?php
            /*Definimos la variable 'contenido' que ácontendr el nombre de
                la ápgina
            a incluir en esta parte de la web. En caso de que no se pase
                ninguna variable
            con la URL, la ápgina a cargar áser la de inicio.
            */
            $contenido;
            if (isset($_GET['pag'])){$contenido = $_GET['pag'].".php";}
               else {
                  $contenido = "inicio.php";
               }
            include $contenido;
         ?>
      </div>
   </body>
</html>
```

Listado 19.2: index.php

```
<h2>á
Pgina de inicio
</h2>
```

Listado 19.3: inicio.php

```
<h1>CABECERA</h1>
```

Listado 19.4: cabecera.php

```
<p>
   <a href="index.php" title="Ir a la ápgina principal">Inicio</a>
</p>
<p>
   <a href="index.php?pag=pagina1" title="Ir a la ápgina 1">áPgina 1</a>
</p>
<p>
   <a href="index.php?pag=pagina2" title="Ir a la ápgina 2">áPgina 2</a>
</p>
```

Listado 19.5: menu-izq.php

```
<h2>á
   Pgina 1
</h2>

<?php
   //Se crea una tabla
   echo "<table summary='' border='1' width='100%'>";

   for ($i=1; $i<6; $i++){
      /*Se van ñaadiendo filas mientras que el valor de la variable
      $i sea menor que el de la variable $conta (6), áincrementndose
      el valor de $i en una unidad ($i++) cada vez que se repite el bucle
      */
      echo "<tr><td>úNmero ".$i."</td><td>Fila ".$i."</td></tr>\n";
   }

   echo "</table>";//Se finaliza la tabla
?>
```

Listado 19.6: pagina1.php

```
<h2>á
   Pgina 2
</h2>í

Das de la semana

<?php
   $semana = array("Lunes", "Martes", "éMircoles", "Jueves", "Viernes", "
       áSbado", "Domingo");

   echo "<ul>\n";
   for ($i=0; $i<count($semana); $i++){
      /*Se van ñaadiendo items de úmen mientras que el valor de la variable
      $i sea menor que el únmero de elementos del array $semana,
          áincrementndose
      el valor de $i en una unidad ($i++) cada vez que se repite el bucle
      */
      echo "<li>".$semana[$i]."</li>\n";
   }
   echo "</ul>\n";
?>
```

Listado 19.7: pagina2.php

Como se puede observar, el código de esta página es similar al de una página HTML excepto por las etiquetas `<?php ?>` que aparecen en él[26].

Podemos ver que hay definidos tres elementos `DIV` que es donde van las distintas partes de la página.

Como primera parte del código a destacar, parémonos en los comentarios[27]. Recordemos que se pueden insertar comentarios de (al menos) dos formas:

```
// Esto es un comentario de una sola línea

/*Comentario de varias líneas,
varias líneas,
varias líneas */
```

En `index.php` podemos observar el siguiente código PHP entre el código HTML:

```
<?php
 $contenido;
 if (isset($_GET['pag'])){$contenido = $_GET['pag'].".php";}
 else {$contenido = "inicio.php";}
 include $contenido;
?>
```

- Si analizamos esta parte del código, vemos que en primer lugar se ha definido la variable `$contenido`, que almacena el valor de la página a incluir en este elemento de la web.
- Después utilizamos la estructura condicional, `if else,` para determinar el valor de la variable `$contenido`. Si se cumple una condición, la variable toma un valor. Si no se cumple (`else`) el valor que toma es `inicio.php`.

  En nuestro ejemplo la condición a cumplir es que esté definida la variable `$_GET['pag'];` para ello utilizamos la función `isset()`.

  `isset($variable)` comprueba si la variable se ha definido, devuelve la respuesta `true` (verdadero) en este caso o `false` en caso contrario.

  Veamos ahora de dónde sale la variable `$_GET['pag']`. Si observamos los enlaces de `menuizq.php` nos encontramos:

  ```
  <a href="index.php?pag=pagina1" title="Ir a la página 1">Pági-
  na 1</a>
  ```

  Junto al nombre de la página introducimos una variable (`pag`) con valor `= pagina1`.[28] En nuestro caso definimos una variable `$contenido.` La variable `$contenido` puede ser:

  - igual al valor de `pag` (que tomará los valores `pagina1` o `pagina2`) junto a la cadena ".php" en caso de estar definida `$_GET['pag']`,
  - o igual a "inicio.php" si dicha variable no está definida (como ocurre en el primer enlace del menú)[29].

[26] El código incluido en las etiquetas `<?php ?>` es el código PHP que el servidor web interpreta, terminando todas las instrucciones que aparecen entre estas dos etiquetas, en punto y coma.

[27] Los comentarios no afectan a la ejecución del programa.

[28] Se pueden adjuntar datos mediante la URL de esta forma, (`method="GET"`) o bien mediante un formulario (`method="POST"`). Se debe acceder a los datos adjuntos con la URL a través de la matriz `$_GET`.

[29] Recordemos que se pueden concatenar *strings* y variables, bien escribiéndolas directamente unas a continuación de otras, o mediante el operador de concatenación que es el punto (`.`)

A continuación incluimos el código de la página php mediante la instrucción `include`, el código incluido será distinto en función del valor de `$contenido`: puede ser `pagina1.php`, `pagina2.php` o `inicio.php`.

Si observamos el código de las páginas `pagina1.php` y `pagina2.php` veremos que hemos utilizado la instrucción php: `echo`.

```
echo "<table summary="" border='1' width='100%'>";
echo "<tr><td>Número ".$i."</td><td>Fila ".$i."</td></tr>\n";
echo "</table>";
```

Hemos añadido `\n` al final de cada fila para añadir un salto de línea y facilitar así la lectura del código HTML generado.

En la página `pagina2.php` definimos otro tipo de variables, los `array` o matrices, que como hemos visto, pueden almacenar un conjunto de valores.

```
$semana = array("Lunes", "Martes", "Miércoles", "Jueves", "Viernes",
                "Sábado", "Domingo");
```

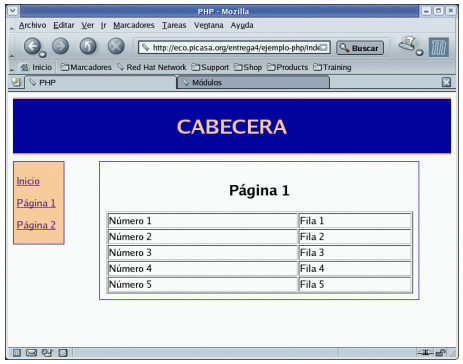
Se puede acceder a cualquiera de los elementos de una matriz[30] mediante `$semana[numero_orden]`. El primer elemento de una matriz ocupa el lugar 0 (`$semana[1]="Martes"`).

Tanto en `pagina1.php` como en `pagina2.php` utilizamos otro tipo de estructura PHP: el bucle `FOR`.

```
for ($i=1; $i<6; $i++){
echo "<tr><td>Número ".$i."</td><td>Fila ".$i."</td></tr>";
}
```

Se define una variable (`$i=1`), una condición (`$i<6`) y un patrón contador (`$i++`).

Las instrucciones que hay dentro del bucle se siguen realizando mientras que el valor de `$i` sea menor que 6, sumando 1 a la variable `$i` cada vez que se realiza el bucle (`$i++`), de esa forma construimos la tabla de la página 1.



Observar que en el fichero `pagina2.php`, para ejecutar el bucle, necesitamos contar primero el número de elementos del array usando la función `count($semana)` (daría como resultado 7)

```
for ($i=0; $i<count($semana); $i++){
  }
```

Para recorrer los elementos del array en el que se almacenan los días de la semana, también podemos usar la sentencia `foreach`, en ese caso quedaría:

[30] Se puede calcular el número de elementos de una matriz mediante la función `count()`.

```
<h2>á
   Pgina 2
</h2>í

Das de la semana

<?php
   $semana = array("Lunes", "Martes", "éMircoles", "Jueves", "
       Viernes", "áSbado", "Domingo");

   echo "<ul>\n";

   foreach ($semana as $dia){
      echo "<li>".$dia."</li>\n";
   }

   echo "</ul>\n";
?>
```

Listado 19.8: pagina2bis.php

19.4.2. Representación gráfica de funciones con PHP

Una de las ventajas que presenta el software libre es la cantidad de recursos que tenemos a nuestra disposición en Internet, y esto es más que manifiesto si se trata de usar aplicaciones (clases) creadas con PHP[31]. Vamos a ver en este ejemplo cómo usar una librería orientada a la creación de gráficos matemáticos usando PHP, se trata de phpplot `http://www.phplot.com/`. Nos centraremos en un uso básico de ella.

Para trabajar con ella, bajamos la última versión de la Web del programa y la descomprimimos en el raíz de Apache

```
root@eco:/var/www# tar -xzvf phplot-5.0rc2.tar.gz
```

Para facilitar las rutas, vamos a trabajar en el directorio creado (aunque no es obligatorio).

```
$ cd phplot
$ chmod 644 phplot.php
```

y nos garantizamos que los permisos sean los adecuados.

En él tenemos la clase phplot, la documentación del programa y ejemplo de uso.

Además de tener Apache y php, para que podamos trabajar adecuadamente con él necesitamos tener instaladas las librerías gd[32] y el módulo que permite trabajar con ellas con Apache[33]:

```
# apt-get install libgd2-xpm php4-gd
```

[31] Por ejemplo:

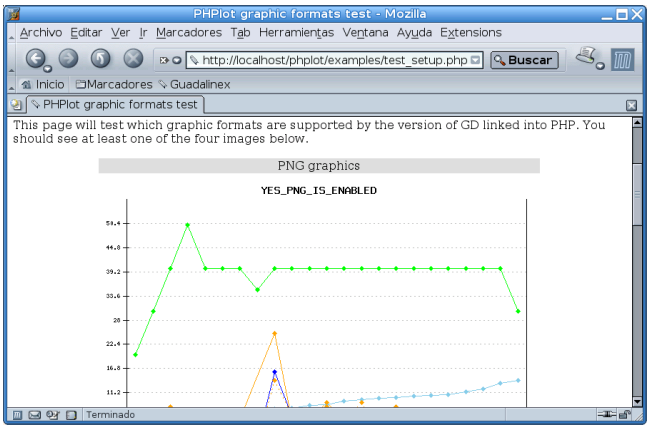
phppdflib `http://www.potentialtech.com/ppl.php` para generar pdfs dinámicamente en nuestras páginas Web

phpmailer `http://phpmailer.sourceforge.net/` se trata de una clase php para enviar emails

[32] Para poder generar los gráficos

[33] En Fedora:

```
#apt-get install php-gd
```



Documentación:

- La disponible en la Web del programa y que se instala al descomprimir el programa.
- Un tutorial sobre su uso `http://www.programacion.net/php/tutorial/phpsol_phplot/`

En nuestro ejemplo, vamos a crear una Web que nos va a permtir representar parábolas. Para conseguirlo necesitamos dos ficheros:

`formulario.html` lo usaremos para introducir los coeficientes de la parábola, dominio de definición y resolución gráfica.

`parabola.php` para obtener los datos de la parábola y pasar los datos a la librería phplot

Analicemos su contenido

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.
    w3.org/TR/html4/loose.dtd">
<html>
   <head>
      <title></title>
   </head>
   <body>
      <form action="parabola.php" method="post">
         Coeficientes de la áparbola
         <br>
         a= <input type="text" name="a" size="3" maxlength="3">
         b= <input type="text" name="b" size="3" maxlength="3">
         c= <input type="text" name="c" size="3" maxlength="3">
         <br>
         Dominio de ódefinicin:
         <br>
         Inicio= <input type="text" name="x_ini" size="3" maxlength="3">
         Final= <input type="text" name="x_fin" size="3" maxlength="3">
         <br>ú
         Nmero de puntos a evaluar (por defecto 50): <input type="text" name
             ="evalua" value="50" size="3" maxlength="3">
         <br>
         <center>
            <input type="submit" name="Representar">
         </center>
      </form>
   </body>
</html>
```

Listado 19.9: formulario.html

Del formulario hay poco que comentar (salvo que el formato de salida es manifiestamente mejorable -:)). Lo utilizamos para introducir las 6 variables que necesitamos:

- 3 para los coeficientes (`a`, `b` y `c`)
- 2 para el dominio de definición (`x_ini` y `x_fin`). Deberíamos usar JavaScript para controlar su valor y obligar a que no se puedan pasar valores incorrectos.
- 1 para el número de puntos a evaluar (`evalua`)

Y el script en php

```
<?php
   //incluimos la clase phplot
   include_once("phplot.php");

   //valor de inicio del eje x
   $inicio=$_POST["x_ini"];

   //valor final del eje x
   $final=$_POST["x_fin"];

   //numero de puntos a evaluar
   $evalua=$_POST["evalua"];

   //Incremento en cada paso.
   //Lo obtenemos dividiendo la amplitud del intervalo
   //entre el únmero de puntos a evaluar
   //No se controla su valor
   $paso=($final-$inicio)/$evalua;

   //áParmetros de la áparbola. No se testea el valor
   //de a, si es cero se ápintar una recta
   $a=$_POST["a"];
   $b=$_POST["b"];
   $c=$_POST["c"];

   //Valor inicial del contador
   $i=$inicio;

   //Definimos el array que contiene los valores a representar
   //el formato de cada dato es (etiqueta,x,y)
   while ($i<=$final){
      $datos=array("",$i,$a*$i*$i+$b*$i+$c);
      $data[]=$datos;
      $i=$i+$paso;
   }

   //Creamos una instancia de la clase PHPlot
   $grafico = new PHPlot;

   //Tipo de datos con los que se trabaja
   $grafico->SetDataType("data-data");

   //íTtulo del ágrfico
   //es mejorable el formato de salida
   $grafico->SetTitle("áGrfica de y=(".$a.")x^2+(".$b.")x+(".$c.")");

   //Tipo del ágrfico, en este caso "lines": une
```

```
    //los puntos con una ílnea
    $grafico->SetPlotType("lines");

    //Grosor de la ílnea
    $grafico->SetLineWidth(3);

    //Longitud de las rayitas de los ejes en ípxel
    $grafico->SetTickLength(1);

    //Incremento del eje y
    //En este ejemplo no lo usamos y dejamos que lo
    //represente de forma áautomtica
    //$grafico->SetVertTickIncrement(10);

    //Incremento del eje X
    //En este caso numeramos el eje X de uno en uno
    //a partir del valor inicial
    $grafico->SetHorizTickIncrement(1);

    //Etiquetas de los ejes
    $grafico->SetXLabel("Eje␣X");
    $grafico->SetYLabel("Eje␣Y");

    //Pasamos a PHPlot los datos del ágrfico
    $grafico->SetDataValues($data);

    //Abre la ágrfica en el navegador Web
    $grafico->DrawGraph();

?>
```

Listado 19.10: parabola.php

Comentemos mejor algunas líneas del fichero:

`include_once("phplot.php");` con esta línea incluimos la librería phplot.php en nuestro script. Es similar a la ya estudiada `include()`, sólo se diferencia en que si el código ha sido ya incluido, no se volverá a incluir. Notar que tal cual está, obligamos a que se encuentre en el mismo directorio que `parabola.php`.

```
while ($i<=$final){

        $datos=array("",$i,$a*$i*$i+$b*$i+$c);

        $data[]=$datos;

        $i=$i+$paso;

        }
```

Con este bucle construimos la matriz de datos que usaremos para representar nuestra parábola. Los datos que usa phpplot son un array cuyos datos son a su vez otro array. Cada uno de los datos tiene la forma (etiqueta, valor_eje_x,valor_eje_y). Por eso, lo que hacemos es recorrer el dominio de definición de la función `($i<=$final)` avanzando en cada iteración el valor de paso (obtenido al dividir la amplitud del intervalo entre el número de puntos a evaluar).

En cada punto a evaluar creamos un array de nombre datos en el que almacenamos los valores

- etiqueta= “” es decir, nada de etiqueta

- valor_eje_x=`$i`
- valor_eje_y=f(`$i`)=`$a*$i*$i+$b*$i+$c` es decir, la imagen de la función para el valor de x (que es `$i`)

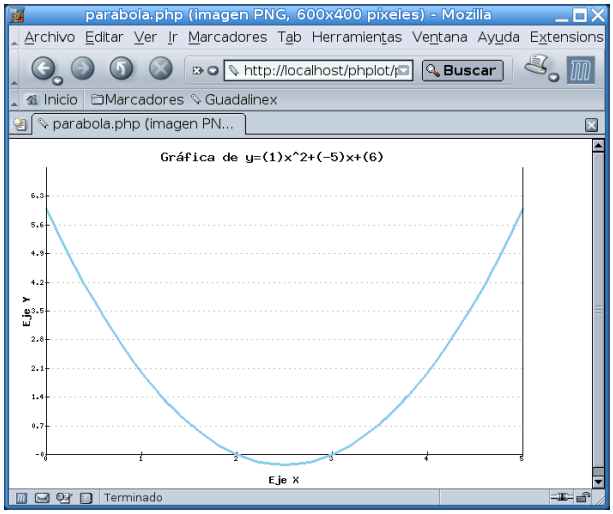
Después, añadimos al array de datos[34] (`$data`) el array creado y aumentamos el valor de x (`$i=$i+$paso;`)

`$grafico->SetDataType("data-data");` cuando trabajamos con programación orientada a objetos, la forma de apuntar a un método de una clase se consigue mediante el operador "`->`". En este caso, lo que hacemos es definir el tipo de gráfico con el que vamos a trabajar en nuestro caso data-data[35]: cada punto consta de una etiqueta, el valor de la x y el valor de y.

`$grafico->SetPlotType("lines");` nos permite definir el tipo de gráfico, permite: `bars`, `lines`, `linepoints`, `area`, `points` y `pie`.

El resto de opciones están comentadas en el propio fichero. Os remitimos a la documentación antes comentada para ampliar sobre su significado o posibilidades de mejora.

Para representar la gráfica de $f(x) = x^2 - 5x + 6$, escribiremos `http://localhost/phplot/formulario.html` en nuestro navegador y tras rellenar el formulario con los datos adecuados[36] obtendremos:



[34] Notar que podíamos usar una sola línea de la forma
`$data[]=array("",$i,$a*$i*$i+$b*$i+$c);`

[35] Las opciones son `text-data`, `data-data` y `data-data-error`. Para saber su significado: `http://www.phplot.com/doc/user_functions.html`

[36] Como dominio se ha tomado `[0, 5]`

Capítulo 20

MySQL

> A menudo la gente se pregunta "¿Cómo es que tengo que obtener una licencia para el servidor de MySQL simplemente porque estoy ejecutando Windows?" Es una pregunta razonable y tiene una respuesta razonable.
>
> ...
>
> Todo esto quiere decir que mientras los costes de desarrollo de sotware en UNIX son prácticamente cero, en WINDOWS pueden ser considerables. A los desarrolladores de MySQL les gusta trabajar en MySQL, pero no tanto como para que quieran pagar por ese privilegio. Los costes de desarrollo en Windows deben recuperarse de algún modo y el coste de la licencia es el método empleado.
>
> Además, los desarrolladores se han encontrado con que necesitan emplear mucho más tiempo para el desarrollo en Windows que en UNIX. (*MySQL*, Paul DuBois, Edit. Prentice Hall)

20.1. Introducción a las Bases de Datos Relacionales

MySQL es un Sistema Gestor de Bases de Datos Relacional (SGBDR[1]). Los Sistemas de Gestión de Bases de Datos nos aíslan de la complejidad del almacenamiento de los datos, sin importarnos en dónde se almacenan físicamente.

Hay tres características importantes inherentes a los SGBD: la separación entre los programas de aplicación y los propios datos, el manejo de múltiples vistas por parte de los usuarios y el uso de un catálogo para almacenar el esquema de la base de datos.

En 1975, el comité ANSI-SPARC (*American National Standard Institute - Standards Planning and Requirements Committee*) propuso una arquitectura de tres niveles para los SGBD, que resulta muy útil a la hora de conseguir estas tres características.

El objetivo de la arquitectura de tres niveles, es el de separar los programas de aplicación de la base de datos física. En esta arquitectura, el esquema de una base de datos se define en tres niveles de abstracción distintos:

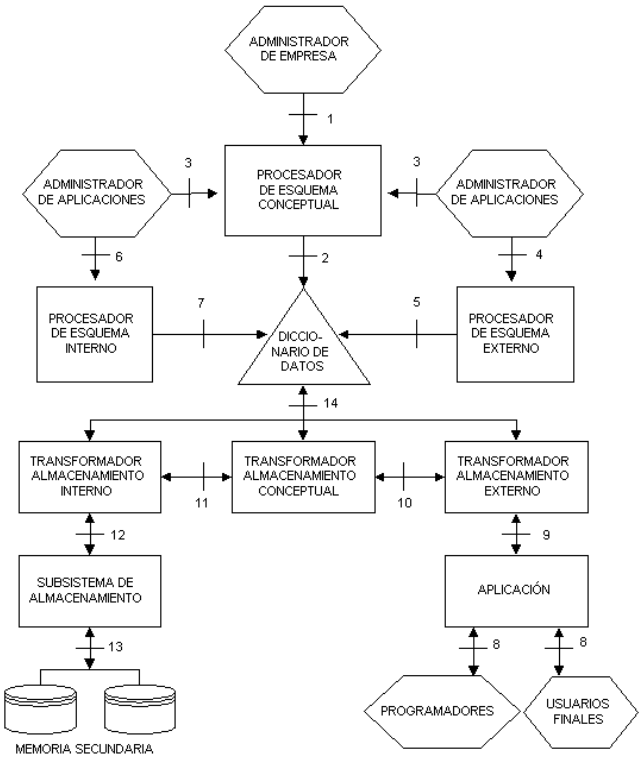
1. En el *nivel interno* se describe la estructura física de la base de datos mediante un esquema interno. Este esquema se especifica mediante un modelo físico y describe todos los detalles para el almacenamiento de la base de datos, así como los métodos de acceso.

2. En el *nivel conceptual* se describe la estructura de toda la base de datos para una comunidad de usuarios (todos los de una empresa u organización), mediante un esquema conceptual. Este esquema oculta los detalles de las estructuras de almacenamiento y se concentra en describir entidades, atributos, relaciones, operaciones de los usuarios y restricciones. En este nivel se puede utilizar un modelo conceptual o un modelo lógico para especificar el esquema.

[1] *Relational DataBase Management System* (RDBMS) en la lengua de Chespir ;-).

3. En el *nivel externo* se describen varios esquemas externos o vistas de usuario. Cada esquema externo describe la parte de la base de datos que interesa a un grupo de usuarios determinado y oculta a ese grupo el resto de la base de datos. En este nivel se puede utilizar un modelo conceptual o un modelo lógico para especificar los esquemas.

El siguiente gráfico muestra la arquitectura ANSI/X3/SPARC.



El modelo relacional fue propuesto originariamente por E.F. Codd en 1970 y se basa en la teoría de conjuntos y la lógica de predicados de primer orden. Sin entrar en demasiados formalismos, significa que un sistema relacional almacena y recupera datos que están en forma de relaciones, que más gráficamente las vemos como tablas.

A muchos os parecerá que la tabla es la manera natural de guardar los datos, pero esto no ha sido siempre así, ni dentro de algún tiempo será lo común. Estructuras en forma de árbol, jerárquicas o en red se han usado para representar los datos, y la tendencia actual es a guardar "objetos".

Un Sistema Gestor de Bases de Datos Relacionales se encarga de administrar múltiples Bases de Datos Relacionales. Una Base de Datos Relacional almacena los datos en tablas separadas, pero que se pueden relacionar entre sí.

Las siglas SQL que en parte dan nombre a MySQL, provienen de "*Structured Query Language*"[2]. SQL es un lenguaje de definición y acceso a las Bases de Datos Relacionales y está definido por el estándar ANSI/ISO SQL. Desde 1986 han surgido varias versiones del estándar. Las de uso más común son "SQL-92", referida a la versión surgida en 1992, "SQL:1999" y "SQL:2003" que es la versión actual del estándar.

Cada tabla representa a una entidad del universo que estamos modelando y consiste en una serie de filas (o tuplas) y de columnas (o atributos). Cada fila de una tabla tiene el mismo número de columnas y representa una *instancia* de esa entidad. Cada columna representa un atributo o *propiedad* de la entidad y es de un tipo determinado.

[2]Lenguaje de Consultas Estructurado, pero el peso de la historia hace que siempre hablemos de Ese-ku-ele.

La siguiente tabla, de nombre "CURSO", posee cuatro columnas denominadas: Número, Nombre, Apellidos y Fecha Nacimiento. Existen tres filas de datos, cada una correspondiente a un alumno de dicho curso, con sus atributos: número en la clase, nombre, apellidos y fecha de nacimiento. Cada atributo será de un tipo; por ejemplo, Nombre y Apellidos serán cadenas de caracteres, Número será un valor entero y Fecha Nacimiento será de un tipo fecha, si existe en nuestro sistema. La primera fila que representamos es una cabecera que contiene los nombres de las columnas y no son datos de la tabla.

Número	Nombre	Apellidos	Fecha Nacimiento
1	Juan	Pérez Gil	07/01/1992
2	Dolores	Fuertes Cabeza	12/03/1991
3	Carmelo	Cotón Rojo	22/10/1991

Una de las grandes ventajas del modelo relacional es que define un método de manipulación de los datos mediante el "álgebra relacional". Todas las manipulaciones posibles sobre las relaciones se obtienen gracias a la combinación de tan sólo cinco operadores: `SELECT`, `PROJECT`, `TIMES`, `UNION` y `MINUS`. Por comodidad, se han definido también tres operadores adicionales que de todos modos se pueden obtener aplicando los cinco fundamentales: `JOIN`, `INTERSECT` y `DIVIDE`. Los operadores relacionales reciben como argumento una relación o un conjunto de relaciones y devuelven una única relación como resultado.

Veamos brevemente estos ocho operadores:

SELECT: devuelve una relación que contiene un subconjunto de las tuplas de la relación a la que se aplica. Los atributos se quedan como estaban. También se denomina `RESTRICT`.

PROJECT: devuelve una relación con un subconjunto de los atributos de la relación a la que se ha aplicado. Las tuplas de la relación resultado se componen de las tuplas de la relación original, de manera que siguen siendo un conjunto en sentido matemático.

TIMES: se aplica a dos relaciones y efectúa el producto cartesiano de las tuplas. Cada tupla de la primera relación está concatenada con cada tupla de la segunda.

JOIN: se concatenan las tuplas de dos relaciones de acuerdo con el valor de un conjunto de sus atributos.

UNION: aplicando este operador a dos relaciones compatibles, se obtiene una que contiene las tuplas de ambas relaciones. Dos relaciones son compatibles si tienen el mismo número de atributos y los atributos correspondientes en las dos relaciones tienen el mismo dominio.

MINUS: aplicado a dos relaciones compatibles, devuelve una tercera que contiene las tuplas que se encuentran sólo en la primera relación.

INTERSECT: aplicado a dos relaciones compatibles restituye una relación que contiene las tuplas que existen en ambas.

DIVIDE: aplicado a dos relaciones que tengan atributos comunes, responde con una tercera que contiene todas las tuplas de la primera relación que se puede hacer que correspondan con todos los valores de la segunda relación.

Por ejemplo, la operación

```
SELECT CURSO donde FechaNacimiento > 05/05/1991
```

devolverá la relación:

Número	Nombre	Apellidos	Fecha Nacimiento
1	Juan	Pérez Gil	07/01/1992
3	Carmelo	Cotón Rojo	22/10/1991

en la que no aparece la tupla que no cumplía la condición de una fecha de nacimiento mayor del 05/05/1991. Al ataque con MySQL.

20.2. Instalación

Fedora

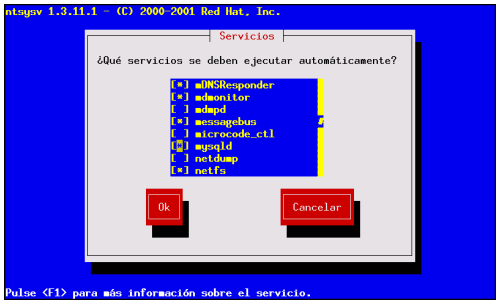
Instalemos los paquetes que permiten disponer de la base de datos MySQL[3]

```
# apt-get install mysql mysql-server mysql-devel
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
Se instalarán los siguientes paquetes extras:
  perl-DBD-MySQL perl-DBI
Se instalarán los paquetes NUEVOS siguientes:
  mysql mysql-devel mysql-server perl-DBD-MySQL perl-DBI
0 upgraded, 5 newly installed, 0 removed and 176 not upgraded.
Need to get 5915kB of archives.
After unpacking 11,8MB of additional disk space will be used.
¿Quiere continuar? [S/n] n
```

y activémosla:

```
# service mysqld start
```

Lo mejor es usar `ntsysv` para que se active en el arranque.



Como vamos a trabajar con el módulo que pemite a PHP disponer de soporte de base de datos MySQL, lo instalamos a su vez:

```
# apt-get install php-mysql
Leyendo listas de paquetes... Done
Construyendo árbol de dependencias... Done
Se instalarán los paquetes NUEVOS siguientes:
  php-mysql
0 upgraded, 1 newly installed, 0 removed and 69 not upgraded.
Need to get 33,2kB of archives.
After unpacking 44,4kB of additional disk space will be used.
```

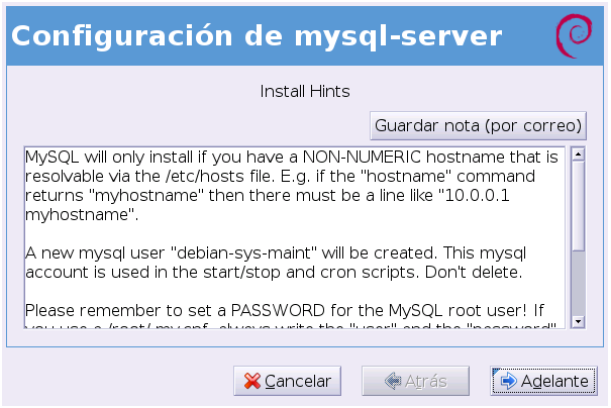
Debian

```
# apt-get install mysql-server mysql-client mysql-common mysql-doc
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes extras:
  libdbd-mysql-perl libdbi-perl libmysqlclient12 libnet-daemon-
perl libperl5.8 libplrpc-perl perl perl-base
  perl-modules
Paquetes sugeridos:
```

[3] El último no es estrictamente necesario. Todos ellos están en los CDs de Fedora

```
  dbishell libterm-readline-gnu-perl libterm-readline-perl-perl
Paquetes recomendados
  perl-doc
Se instalarán los siguientes paquetes NUEVOS:
  libdbd-mysql-perl libdbi-perl libnet-daemon-perl libplrpc-perl mysql-
client mysql-doc mysql-server
Se actualizarán los siguientes paquetes:
  libmysqlclient12 libperl5.8 mysql-common perl perl-base perl-modules
6 actualizados, 7 se instalarán, 0 para eliminar y 740 no actualizados.
Necesito descargar 15,7MB de archivos.
Se utilizarán 19,8MB de espacio de disco adicional después de desempa-
quetar.
¿Desea continuar? [S/n]
```

En el proceso de instalación se nos avisará de una serie de cuestiones: información sobre la necesidad de que nuestra máquina tenga un nombre en `/etc/hosts`, que se ha creado una nueva cuenta de usuario[4] y nos recuerda la necesidad de proteger el servidor asignando una password al root. También nos informa de que los accesos vía red están deshabilitados por defecto y que para activarlos hay que descomentar la opción skip-networking del fichero de configuración del servidor de bases de datos (`/etc/mysql/my.cnf`)[5]



Como nuestra intención es trabajar con PHP y MySQL instalemos, además, el módulo adecuado[6]:

```
#apt-get install php4-mysql
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes NUEVOS:
php4-mysql
0 actualizados, 1 se instalarán, 0 para eliminar y 730 no actualizados.
Necesito descargar 21,2kB de archivos.
Se utilizarán 119kB de espacio de disco adicional después de desempa-
quetar.
```

20.2.1. Configuración del servidor

El servidor de bases de datos se configura mediante el fichero[7]:

[4] Este usuario especial (`debian-sys-maint`) tiene privilegios similares a root de mysql.

[5] Podemos reconfigurarlo con `#dpkg-reconfigure mysql-server`

[6] Depende, entre otros de `apache2` y `libapache2-mod-php4`, así que si no están instalados, los instalará ahora.

[7] En Debian, en el directorio `/etc/mysql` hay varios ficheros más, usados en la configuración del servidor (`debian.cnf` y `debian-log-rotate.conf`). Para conocer su significado, mirar la documentación que acompaña al programa.

Debian `/etc/mysql/my.cnf`

Fedora `/etc/my.cnf`

Está dividido en secciones y su sintaxis es la habitual[8]:

`#` indica que se trata de un comentario

`[sección]` indica que se inicia una sección. Los parámatros incluidos dentro de una sección afectan sólo a ésta.

↬ `[mysqld]` Opciones para el servidor

`opcion = valor`

↬ `port=3306` puerto en el que escucha el servidor

`opcion` se trata de parámetros booleanos que se establecen al estar presentes en el fichero de configuración.

↬ `log-bin` si lo escribimos (descomentamos) registraremos las actualizaciones que realicemos en todas las tablas.

`set-variable = variable = valor` para establecer los valores de las variables

↬ `set-variable=write_buffer=2M` establecemos en 2MB el bufer de escritura.

En[9] `/usr/share/doc/mysql-server/examples` tenemos varios ficheros de configuración de ejemplo adecuados a la memoria dedicada (ojo, que no se trata de la memoria RAM instalada) a MySQL en nuestro ordenador. En la columna memoria hemos puesto los valores indicativos que no se deben interpretar de forma exacta, sino más bien como valores de referencia.

Fichero	Memoria para MySQL
`my-huge.cnf`	1 a 2 GB
`my-large.cnf`	512MB
`my-medium.cnf`	32MB
`my-small.cnf`	Sistemas con poca memoria RAM

Usando este fichero podremos, por ejemplo, optar por el tipo de tabla predeterminado o que los mensajes de error aparezcan en castellano. En general, para un trabajo normal no es necesario modificar nada en este fichero, salvo conseguir que los mensajesde error se muestren en castellano [10]

```
[mysqld]
....
language = /usr/share/mysql/spanish
....
```

[8] Para ampliar y si hemos instalado el paquete mysql-doc:
`file:///usr/share/doc/mysql-doc/manual_Using_MySQL_Programs.html#Program_Options`

[9] `/usr/share/doc/mysql-sever-x-x-x/` en Fedora

[10] Y, en su caso, permitir accesos vía red. Por defecto, sólo se permite accesos desde el bucle local:

```
bind-address = 127.0.0.1
```

Si deseamos conectar desde otra máquina, habrá que permitirlo haciendo que el servidor escuche en el interfaz de red. Por ejemplo, si nuestra IP local es 192.168.0.1 escribiremos

```
bind-address = 192.168.0.1
```

20.3. Inicio de MySQL

20.3.1. Aseguremos el servidor

Inicialmente, cualquiera puede conectarse al servidor MySQL y crear o modifcar las bases de datos. Así que lo primero que deberíamos hacer es asignar al root una contraseña de acceso. Podemos conseguirlo de varias formas, pero para ejemplificar la forma de conectar al servidor lo haremos con:

```
root@guada04:~# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5 to server version: 4.0.24_Debian-5-log
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql>
```

y nos aparecerá el prompt de introducción de comandos. Si no deseamos hacer nada y deseamos salir, podemos escribir[11]

```
mysql> quit
```

Pero todavía no vamos a salir. Veamos antes un par de ejemplos de cómo trabajar con él. En primer lugar listaremos las dos bases de datos que ya hay creadas, se consigue con el comando

```
mysql> SHOW DATABASES;
+----------+
| Database |
+----------+
| mysql    |
| test     |
+----------+
2 rows in set (0.00 sec)
```

si nos fijamos un poco en él podemos constatar que:

- En las sentencias, da igual que los escribamos en mayúsculas o no. No es así en los nombres de los campos, tablas, bases de datos, etc.
- Las sentencias se han de terminar en `;`
- Se mantiene un historial de comandos al que podemos acceder de la forma habitual, es decir, con los cursores. Pongamos la password del root[12]:

  ```
  mysql> SET PASSWORD=PASSWORD('palabra_acceso');
  Query OK, 0 rows affected (0.01 sec)
  ```

y salgamos del servidor

```
mysql> exit
Bye
```

[11] MySQL no es casesensitive (sensible a mayúsculas y minúsculas) en cuanto a los comandos, es decir, que podemos escribir `QUIT`, `Quit`, ...

[12] Podemos hacer esto desde la línea de comandos, ejecutando:

```
#mysqladmin password palabra_acceso
```

20.3.2. Un poco de comandos

Si ahora intentamos conectar como antes, comprobaremos que no es posible

```
sh-2.05b$ mysql
ERROR 1045: Access denied for user: 'root@localhost' (Using pass-
word: NO)
```

así que, avisemos al servidor MySQL de quién somos (`-u root`, es decir el usuario root) y que, además, vamos a introducir la contraseña (`-p`)

```
sh-2.05b$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8 to server version: 4.0.24_Debian-5-log
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql>
```

Pero sigamos. Demos un breve repaso a algunos comandos básicos. Creemos una base de datos[13] (que luego borraremos)

```
mysql> CREATE DATABASE prueba;
Query OK, 1 row affected (0.05 sec)
```

y comprobemos los cambios:

```
mysql> SHOW DATABASES;
+----------+
| Database |
+----------+
| mysql    |
| prueba   |
| test     |
+----------+
3 rows in set (0.00 sec)
```

Pero sólo hemos creado la base de datos, así que está vacía. Creemos una tabla (muy simple) que va a contener sólo dos campos: `nombre` y `apellidos`, serán de tipo texto de un máximo de 20 caracteres, longitud variable (`VARCHAR`) y no permitimos que ninguno de los dos quede vacío[14] (`not null`)

```
mysql> USE prueba;
Database changed
mysql> CREATE TABLE datos(nombre VARCHAR(20) not null, apellidos VAR-
CHAR(30) not null);
Query OK, 0 rows affected (0.06 sec)
```

En primer lugar, usamos la base de datos en la que crearemos la tabla, para después crearla. Comprobemos las tablas creadas con[15]:

[13] De forma equivalente, podemos usar:

```
#mysqladmin -u root -p create prueba
```

[14] Formalmente hablando, si especificamos que una columna es NOT NULL obligamos a que la columna no pueda contener valores NULL. Pero didácticamente es más sencilla y efectiva la idea de campo vacío o no vacío.

[15] Con:

`SHOW STATUS` se nos muestran las variables de estado del servidor

`SHOW VARIABLES` lista las variables del servidor

```
mysql> SHOW TABLES;
+------------------+
| Tables_in_prueba |
+------------------+
| datos            |
+------------------+
1 row in set (0.00 sec)
```

y analicemos su estructura con:

```
mysql> DESCRIBE datos;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| nombre    | varchar(20) |      |     |         |       |
| apellidos | varchar(30) |      |     |         |       |
+-----------+-------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

Es el momento de introducir un par de datos

```
mysql> INSERT INTO datos VALUES ("Pepe","Pinto Gorgori-
to"),("Pilar","Pan y Agua");
Query OK, 2 rows affected (0.00 sec)
Records: 2  Duplicates: 0  Warnings: 0
```

y mostrarlos por pantalla:

```
mysql> SELECT * FROM datos;
+--------+-----------------+
| nombre | apellidos       |
+--------+-----------------+
| Pepe   | Pinto Gorgorito |
| Pilar  | Pan y Agua      |
+--------+-----------------+
2 rows in set (0.02 sec)
```

Pero deseamos que el usuario Thales pueda manejar esta base de datos (tal cual está, sólo el root tiene acceso a ella). Así que lo autorizamos con:

```
mysql> GRANT ALL ON prueba.* TO thales@localhost IDENTIFIED BY 'pass-
word';
Query OK, 0 rows affected (0.00 sec)
```

➥ **Para practicar**

1. Comprobar que ahora el usuario al que hayamos dado permiso con la línea anterior, puede acceder a esta base de datos. Habrá que conectarse al servidor con:

   ```
   $mysql -u usuario -p
   ```

   e introducir la contraseña "password"

2. Si hemos permitido accesos vía red y deseamos que nuestro usuario Thales pueda conectar de forma remota desde cualquier IP (eso significa el %) escribiremos

   ```
   mysql> GRANT ALL ON prueba.* TO thales@'%' IDENTIFIED BY 'password';
   Query OK, 0 rows affected (0.00 sec)
   ```

   Si nuestro servidor de bases de datos está a la escucha en la IP 192.168.0.1, para conectar, usaremos el comando

```
$mysql -u thales -h 192.168.0.1 -p
```

Usando `GRANT`[16] y `REVOKE` podemos conceder y retirar, respectivamente, derechos a los usuarios[17] de cuatro maneras distintas: a nivel global, de base de datos, de tablas o de columnas.

Los objetos a los que se puede conceder cada privilegio son:

Cuadro 20.1: Privilegios para usuarios

Privilegios	Se aplica a	Permite al usuario
`SELECT`	tablas y columnas	seleccionar filas (registros)
`INSERT`	tablas y columnas	insertar filas
`UPDATE`	tablas y columnas	modificar valores de filas
`DELETE`	tablas	eliminar filas
`INDEX`	tablas	crear y eliminar índices
`ALTER`	tablas	modificar su estructura
`CREATE`	bases y tablas	crear bases de datos o tablas
`DROP`	bases y tablas	eliminar bases de datos o tablas

La sintaxis genérica es[18]:

```
GRANT privilegios [columnas] ON elementos TO usuario
    [IDENTIFIED BY 'contraseña'] [WITH GRANT OPTION]
```

donde

`privilegios` son los que aparecen en la primera columna de la tabla 20.1.

`columnas` es opcional, se trata de una lista de columnas separadas por comas a las que aplicar los privilegios.

`elemento` se trata de la(s) base(s) de datos o tabla(s) a las que aplicar los privilegios. Es posible usar comodines:

- `*.*` todas las bases y tablas
- `prueba.*` todas las tablas de la base de datos ***prueba***
- `prueba.datos` a la tabla ***datos*** de la base de datos ***prueba***

`usuario [IDENTIFIED BY 'contraseña']` usuario al que se conceden los privilegios y en su caso contraseña de acceso

`WITH GRANT OPTION` si se pone, permite que el usuario delegue sus privilegios en otros usuarios.

Sin que nuestro sufrido usuario se entere de que podía conectar, nos arrepentimos y le revocamos los privilegios concedidos con:

```
mysql> REVOKE ALL ON * FROM thales@localhost;
```

Listemos ahora sólo los apellidos con:

[16] En la sección 21.1.2 en la página 464, en la explicación sobre la instalación de Moodle, se ejemplifica la forma de trabajar con esta sentencia en modo gráfico usando phpMyAdmin.

[17] No vamos a analizar ni los privilegios para administradores, ni privilegios especiales.

[18] Como es usual, las opciones incluidas entre corchetes son opcionales

```
mysql> SELECT apellidos FROM datos;
+-----------------+
| apellidos       |
+-----------------+
| Pinto Gorgorito |
| Pan y Agua      |
+-----------------+
2 rows in set (0.00 sec)
```

mejor los ponemos ordenados alfabéticamente:

```
mysql> SELECT apellidos FROM datos ORDER BY apellidos;
+-----------------+
| apellidos       |
+-----------------+
| Pan y Agua      |
| Pinto Gorgorito |
+-----------------+
2 rows in set (0.00 sec)
```

El comando `SELECT` es quizás el más utilizado cuando trabajamos con MySQL. Sirve para recuperar las filas de una base de datos que cumplan con los criterios especificados. Su sintaxis es "potente y compleja", permite usar operadores de comparación y recuperar datos combinados de varias tablas. Más adelante, cuando estudiemos el programa phpMyAdmin (20.4) y la integración de PHP y MySQL ampliaremos sobre su uso.

Tras esta pequeña introducción dejemos las cosas (casi) como estaban, así que antes de seguir borremos la base datos[19]:

```
mysql> DROP DATABASE prueba;
```

El número de sentencias y posibilidades es muy grande, así que pensando en gente "desmemoriada" como nosotros, mejor si usamos: PHPMyAdmin

20.4. PHPMyAdmin

Se trata de un conjunto de scripts de php que permiten gestionar bases de datos MySQL usando un navegador web. Usándola, y sin conocer las sentencias de MySQL, es posible (entre otras cosas) gestionar los permisos de acceso de los distintos usuarios a las bases de datos o tablas, así como consultar, crear, borrar, modificar bases de datos o tablas.

La web principal es `http://www.phpmyadmin.net/home_page/` y la última versión estable la podemos bajar de `http://www.phpmyadmin.net/home_page/downloads.php`

20.4.1. Instalación

Si usamos Debian, podemos instalarlo con:

```
#apt-get update
#apt-get install phpmyadmin
```

[19] Con este comando podemos borrar, a su vez, una tabla.

El que para Fedora no exista el paquete en los repositorios oficiales, añadido a la facilidad de instalación si usamos el paquete comprimido, nos ha hecho optar por el método general común a ambas distribuciones.

Una vez descargado, lo instalamos con[20]:

```
# cp phpMyAdmin-2.6.1-pl3.tar.bz2 /var/www/
# cd /var/www/
# tar -xjvf phpMyAdmin-2.6.1-pl3.tar.bz2
```

Una vez desempaquetado, os recomendamos que le cambiéis el nombre al directorio o que, usando un enlace simbólico, facilitéis el acceso a él desde el navegador web tecleando menos

```
#ln -s /var/www/html/phpMyAdmin-2.6.1-pl3 /var/www/html/phpMyAdmin
```

En el directorio creado, tenemos dos ficheros que nos explican la forma de finalizar la instalación, se trata de `Documentation.txt` o `Documentation.html`. En resumen, los pasos a seguir son:

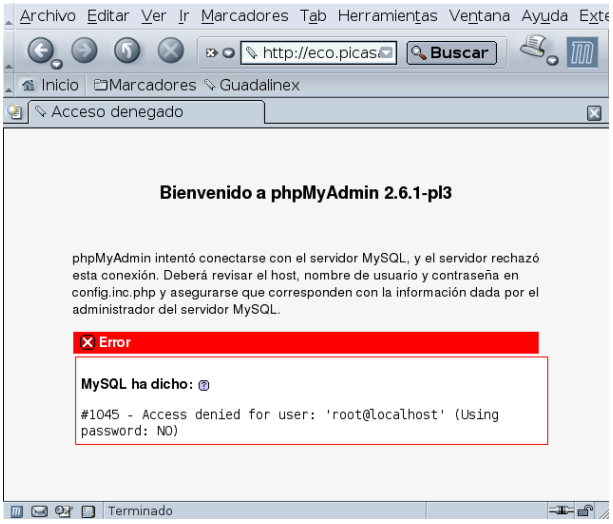
1. Editar el fichero `config.inc.php` y adecuarlo a nuestra máquina, los parámetros importantes a cambiar son[21]:

   ```
   $cfg['PmaAbsoluteUri'] = 'http://localhost/phpMyAdmin/';
   ```

   aquí escribiremos la ruta completa para acceder a phpMyAdmin, por ejemplo

   `http://localhost/phpMyAdmin` o `http://www.midominio.org/phpMyAdmin`

   Para acceder al programa, sólo es necesario ejecutar[22] `http://localhost/phpMyAdmin`



   Si hemos puesto contraseña para acceder al motor de base de datos, nos aparecerá el error y tendremos que configurar adecuadamente las variables:

[20] En Fedora hay que adecuar lo que sigue al `DocumentRoot` de Apache: `/var/www/html`

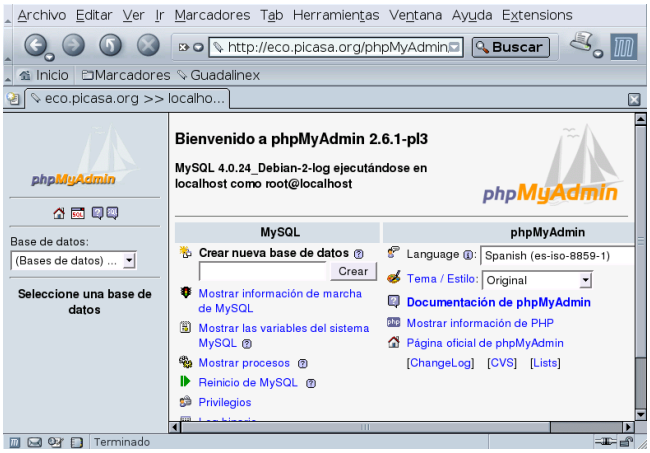
[21] Si no tenemos un nombre de máquina completamente cualificado, podemos optar por escribir `localhost`.

[22]

- Si no hemos puesto contraseña de acceso al gestor de base de datos, se nos avisará además de que:
  "Su archivo de configuración contiene parámetros (root sin contraseña) que corresponden a la cuenta privilegiada predeterminada de MySQL. Su servidor de MySQL está usando estos valores, que constituyen una vulnerabilidad. Se le recomienda corregir esta brecha de seguridad."
- Si nos aparece en inglés sólo hemos de optar por el idioma adecuado.

```
$cfgServers[$i]['user']= 'root';
$cfg['Servers'][$i]['password'] = 'contraseña';
```

con ellas, definimos el usuario y contraseña que usará el programa para iniciar la conexión con el servidor MySQL.



2. Crear un archivo `.htaccess` en el directorio `/var/www/phpMyAdmin-2.6.1-pl3` para que sólo nosotros podamos gestionar nuestras bases de datos (véase la entrega anterior para saber qué pasos hay que seguir). Por ejemplo, para restringir el acceso al usuario[23] `thales`

```
# cat .htaccess
AuthType Basic
AuthName "phpMyAdmin"
AuthUserFile /var/www/passwd/.htpasswd
AuthGroupFile /dev/null
require user thales

#htpasswd /var/www/passwd/.htpasswd thales
```

Actualizar: Para actualizar a una versión más moderna, sólo hay que descomprimir la nueva versión en el lugar adecuado (raíz de Apache) y copiar el fichero viejo `config.php.inc` a la nueva situación.

20.4.2. ➥Base de datos cursolinux

Vamos a ejemplificar el trabajo con una base de datos de nombre ***cursolinux*** en la que vamos a controlar las faltas de nuestros alumnos. El proceso lo vamos a comentar de dos formas:

- Con `phpMyAdmin`
- Usando comandos de MySQL: los hemos puesto a continuación y en recuadros de texto[24].

Mediante este ejemplo, introduciremos los conceptos básicos para trabajar con bases de datos.

La base de datos de ejemplo va a constar incialmente de dos tablas.

[23] En Fedora, `htpasswd` y ajustar el `DocumentRoot`

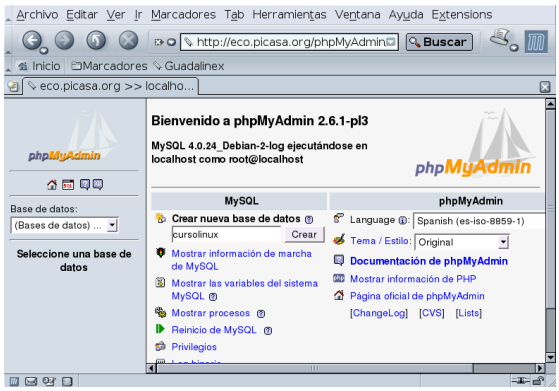
[24] Si bien aparecen en la capturas gráficas, hemos optado "por sacarlos" de ellas para facilitar su lectura y no tener que agrandar en exceso los gráficos.

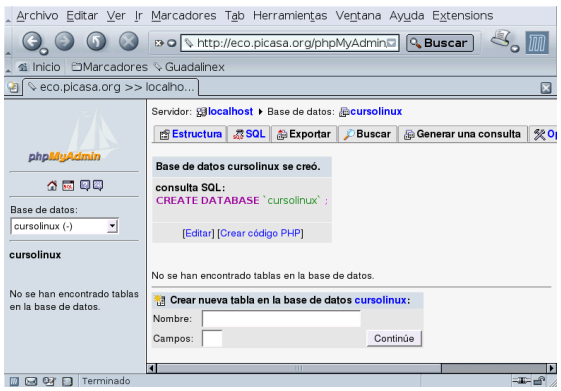
- La tabla `alumnos`, que constará de los campos: `codigo`, `nombre`, `apellido1`, `apellido2`, `curso` y `dni`

  El único campo extraño es `codigo`, va a ser la clave principal, la generará el propio MySQL y nos va a permitir (después) establecer una relación uno varios (un alumno puede tener varias faltas) entre ambas bases de datos.

- La tabla faltas, de estructura: `codigo`, `fecha`

Lo primero será crear la base de datos. Iniciamos la aplicación, escribimos el nombre de nuestra base de datos y pulsamos sobre **[Crear]**

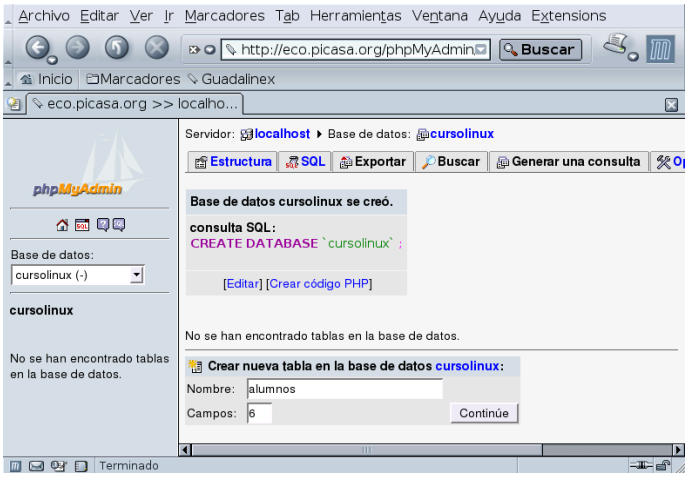




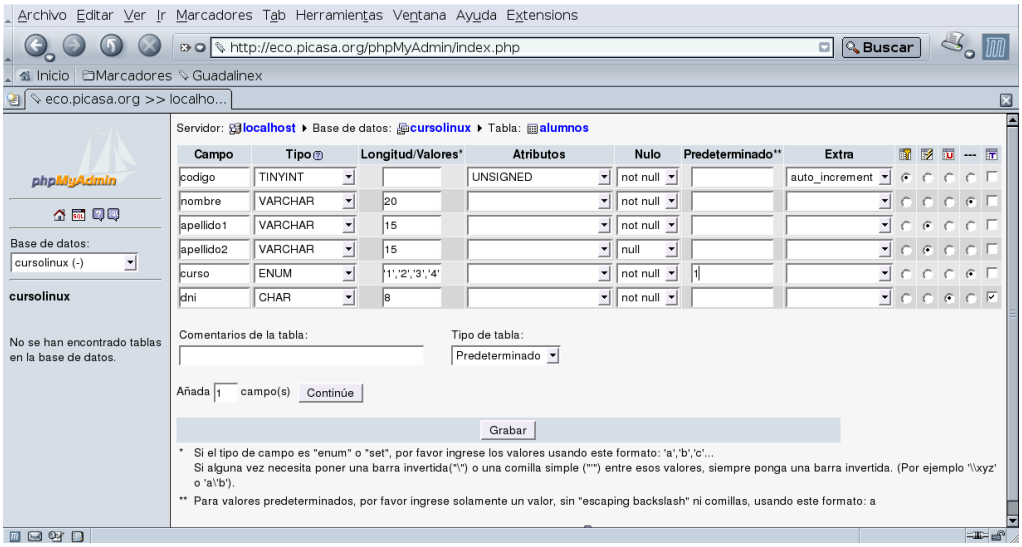
De esta forma, acabamos de crear un directorio en `/var/lib/mysql` de nombre `cursolinux`. El código equivalente en modo comando sería:

```
# mysql -u root -p
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 18 to server version: 4.0.18-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> CREATE DATABASE cursolinux;
```

Es el momento de añadir las tablas. En primer lugar introducimos el nombre de la tabla y número de campos:



para, a continuación, establecer el nombre y tipo de cada de uno de ellos:



Los valores posibles son:

Campo se trata del nombre del campo en el que vamos a almacenar los datos.

Tipo básicamente disponemos de tres tipos de datos en MySQL: numéricos, de cadena y fechas. Seleccionar el tipo y tamaño que mejor se adecúen a nuestra base de datos no es tarea sencilla, aunque para comenzar podemos permitirnos ciertos lujos que seguro que en la "facultad" serían gravemente penalizados. Las tablas 20.2 en la página siguiente, 20.4 en la página 441, 20.6 en la página 442 y 20.7 en la página 442, resumen los tipos soportados[25].

Longitud/Valores aquí introduciremos los tamaños máximos para nuestros campos o, en el caso de enumeraciones o conjuntos, los posibles valores a tomar.

↬ Por ejemplo, el campo `codigo` lo dejamos con el valor por defecto (de 0 a 255). El campo `nombre` tendrá una anchura máxima de 20 caracteres que si no rellenamos en su totalidad, no ocuparán espacio[26] (tipo `VARCHAR`). Sin embargo, el campo `dni` está limitado a 8 pero de tipo `CHAR`: aunque un `dni` tenga 5 dígitos se completará con espacios en blanco hasta llegar a los 8. El campo `curso` es una enumeración de posibles valores 1, 2, 3 y 4.

Atributos se puede dejar vacío o ser[27]: `BINARY` (si se opta por este atributo, hacemos distinción entre mayúsculas y minúsculas), `UNSIGNED` o `UNSIGNED ZEROFILL`.

Nulo si optamos por *`null`*, permitimos que ese campo se pueda dejar vacío. Si optamos por *`not null`* obligamos a que se tenga que rellenar de forma obligatoria.

↬ Observar que salvo la columna en donde se almacenará el segundo apellido (en previsión de inmigrantes que provengan de países en el que sólo usan el primer apellido) el resto de campos hay que rellenarlos de forma obligatoria.

Predeterminado nos permite introducir un valor predeterminado.

[25] No es una referencia exhaustiva. Para eso, consultar: `http://dev.mysql.com/doc/mysql/en/Column_types.html`

[26] Sólo se almacenan los caracteres introducidos más un byte para almacenar la longitud de la cadena. El tipo `CHAR`, pese a ocupar más espacio de almacenamiento es, en general, más eficiente.

[27] Para entender su significado, véanse las notas que siguen a las tablas con los tipos de datos

Cuadro 20.2: Tipos Numéricos

Tipo	Valor	Rango	Bytes
`TINYINT [(M)]`	Entero pequeño	-128 a 127 (-2^7 a 2^7-1)	1
`SMALLINT [ (M) ]`	Entero pequeño	-32.768 a 32.767	2
`MEDIUMINT [ (M) ]`	Entero mediano	-8.388.608 a 8.388.607	3
`INT [ (M) ]`	Entero	-2.147.483.648 a 2.147.483.647	4
`BIGINT [ (M) ]`	Entero grande	-2^{63} a $2^{63}-1$	8
`FLOAT [ (M,D) ]`	Decimal de precisión simple	3,402823466E+38 a -1,175494351E-38 y de 1,175494351E-38 a 3,402823466E+38	3
`DOUBLE [ (M, D) ]`	Decimal de precisión doble	-1,7976931348623157E+308 a -2,225073855072014+E308 y de -2,225073855072014+E308 a 1,7976931348623157E+308	4
`DECIMAL [ (M [, D]) ]`	Decimal almacenado como cadena	-1,7976931348623157E+308 a -2,225073855072014+E308 y de -2,225073855072014+E308 a 1,7976931348623157E+308	M+2

Notas: Todos los tipos admiten, como parámetros opcionales:

`M` indica el tamaño máximo mostrado

`UNSIGNED` en el caso de los tipos enteros, el rango es de 0 a `|valormínimo|+valormáximo` (es decir, $2^{8 \cdot bytes-1}$). Por ejemplo el tipo `TINYINT UNSIGNED` tendrá de rango de 0 a 255. Si se trata de un número de coma flotante (FLOAT o DOUBLE) se mantiene el valor máximo positivo y se impiden los valores negativos. Si el tipo es `DECIMAL`, `M` se usa para indicar el número total de dígitos (sin el signo ni punto decimal) y `D` es el número de decimales (por defecto 0).

`ZEROFILL` el resultado sería que rellenamos con ceros y no con espacios, hasta completar el valor.

↬ En el ejemplo, y como este año tenemos muchos primeros de ESO, hemos optado por poner un 1. Un caso un poco más real[28] puede ser: una base de datos en la que almacenar el domicilio de nuestos alumnos y optar por poner como valor predeterminado del campo "provincia" aquella en la que se sitúe el centro de enseñanza.

Extra si deseamos que el campo se autoincremente cada vez que introduzcamos un registro, optaremos por marcar `auto_increment`. Sólo puede usarse con tipos enteros. Los campos de este tipo se usan en general para las claves principales y sólo puede existir uno por tabla. Este tipo de columnas debe estar indexadas.

 **Primaria** si marcamos esta casilla, la columna se tratará como clave principal de la tabla y se indexará de forma automática. Las entradas en esta columna han de ser únicas.

 **Índice** usar índices es el mejor método para hacer las consultas más rápidas.

 **Único** con `UNIQUE` obligamos a que en esa columna no puedan existar datos repetidos.

↬ El `dni` es un buen ejemplo de ello. Podemos tener varias personas con el mismo nombre y apellidos pero no deberían existir dos con el mismo `dni`

[28] O al menos eso os deseamos -;)

Cuadro 20.4: Tipos cadena

Tipo	Descripción	Máx. caracteres	Bytes
`CHAR (M) [BINARY]`	Cadena de caracteres de longitud fija	De 0 a 255	M
`VARCHAR (M) [BINARY]`	Cadena de caracteres de longitud variable	De 0 a 255	L+1
`TINYBLOB`	Objeto binario largo pequeño	255 ($2^8 - 1$)	L+1
`TINYTEXT`	Objeto largo pequeño	255 ($2^8 - 1$)	L+1
`BLOB`	Objeto binario largo	65.535 ($2^{16} - 1$)	L+2
`TEXT`	Objeto largo	65.535 ($2^{16} - 1$)	L+2
`MEDIUMBLOB`	Objeto binario largo mediano	16.777.215 ($2^{24} - 1$)	L+3
`MEDIUMTEXT`	Objeto largo mediano	16.777.215 ($2^{24} - 1$)	L+3
`LONGBLOB`	Objeto binario largo grande	4.294.967.295 ($2^{32} - 1$)	L+4
`LONGTEXT`	Objeto largo grande	4.294.967.295 ($2^{32} - 1$)	L+4
`ENUM('valor1', 'valor2')`	Cadena de caracteres con uno solo de los valores especificados	65.535 valores	1 ó 2
`SET('valor1', 'valor2')`	Conjunto de caracteres formado por la unión de ninguno, uno o varios de los valores especificados.	64 valores	1,2,3,4 u 8

Nota: Los tipos binarios discriminan entre mayúsculas y minúsculas (`CHAR` y `VARCHAR` lo admiten como opción).

Los atributos `NULL` y `NOT NULL` se pueden especificar para cualquiera de los tipos (el predeterminado es `NULL`).

`L` es la longitud de la cadena y `M` el ancho fijo.

- - - Para desmarcar las tres últimas casillas.

Texto Completo si optamos por marcar esta casilla, crearemos un índice "especial" de tipo `FULLTEXT`, con un índice de este tipo (sólo se usa con tablas `MyISAM`[29]) para columnas del tipo `CHAR`, `VARCHAR` o `TEXT`. Con bases de datos muy grandes es mejor no usarlos.[30]

Al pulsar sobre **[Grabar]**[31] obtendremos una nueva ventana desde la que podremos modificar los valores de todos los campos introducidos hasta ahora, añadir nuevos campos o borrar algunos de los existentes.

[29] En realidad, esto no es una limitación seria ya que son las las tablas que usa MySQL desde la versión 3.23. Son las sustitutas de las tablas ISAM (Método de Acceso Secuencial Indexado).

[30] Para ampliar: `http://dev.mysql.com/doc/mysql/en/Fulltext_Search.html`

[31] Acabamos de crear tres ficheros en `/var/lib/mysql/cursolinux`, se trata de

```
# ls -al /var/lib/mysql/cursolinux
total 16
-rw-rw----  1 mysql mysql 8712 abr 1 17:58 alumos.frm
-rw-rw----  1 mysql mysql    0 abr 1 17:58 alumos.MYD
-rw-rw----  1 mysql mysql 1024 abr 1 17:58 alumos.MYI
```

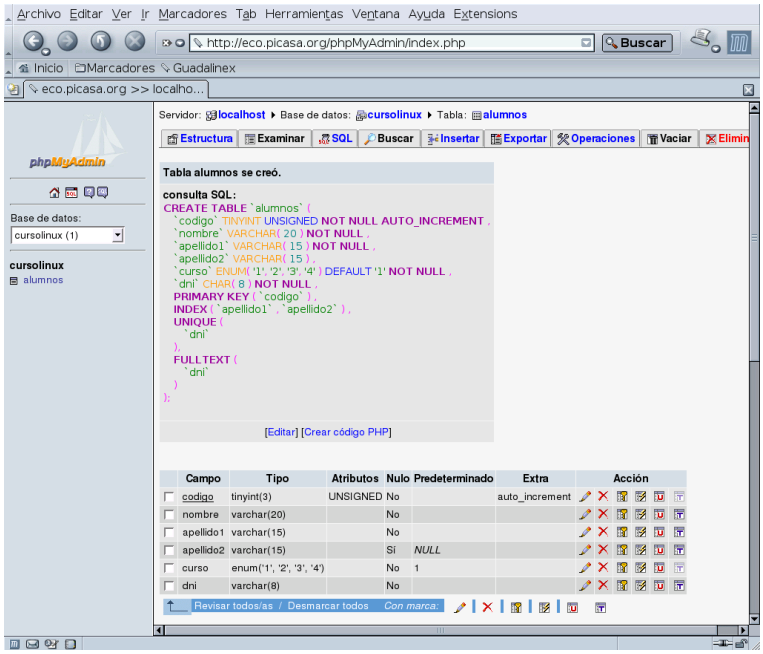
el primero (`.frm`) contiene la estructura de la tabla, el segundo (`.MYD`) contendrá los datos y el tercero (`.MYI`) los índices asociados a esa tabla.

Cuadro 20.6: Tipos fecha y hora

Tipo	Descripción	Rango	Bytes
DATETIME	AAAA-MM-DD HH:MM:SS	1000-01-01 00:00:00 a 9999-12-31 23:59:59	8
DATE	AAAA-MM-DD	1000-01-01 a 9999-12-31	3
TIMESTAMP	AAAAMMDDHHMMSS	19700101000000 a cualquier fecha del 2037	4
TIME	HH:MM:SS	-838:59:59 a 838:59:59	3
YEAR	AAAA	1901 a 2155	1

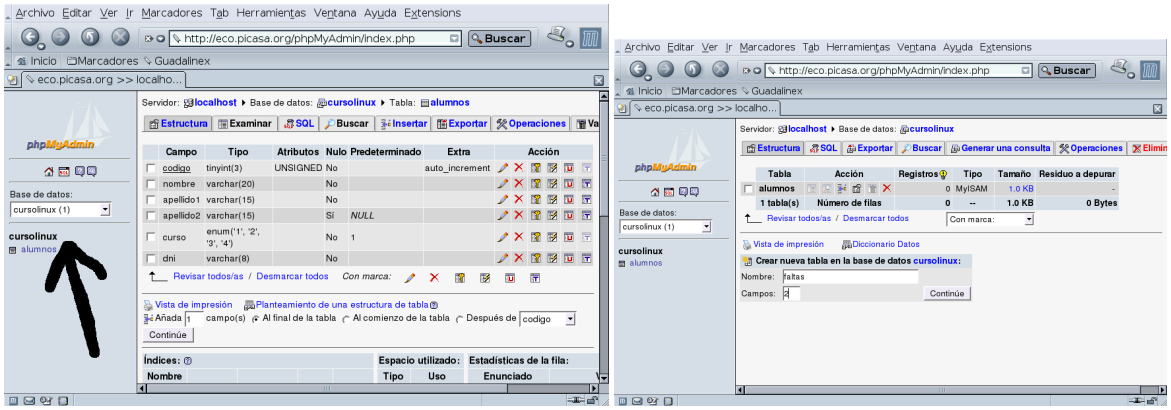
Cuadro 20.7: Tipos TIMESTAMP

Tipo	Formato pantalla
TIMESTAMP(14)	AAAAMMDDHHMMSS
TIMESTAMP(12)	AAMMDDHHMMSS
TIMESTAMP(10)	AAMMDDHHMM
TIMESTAMP(8)	AAAAMMDD
TIMESTAMP(6)	AAMMDD
TIMESTAMP(4)	AAMM
TIMESTAMP(2)	AA

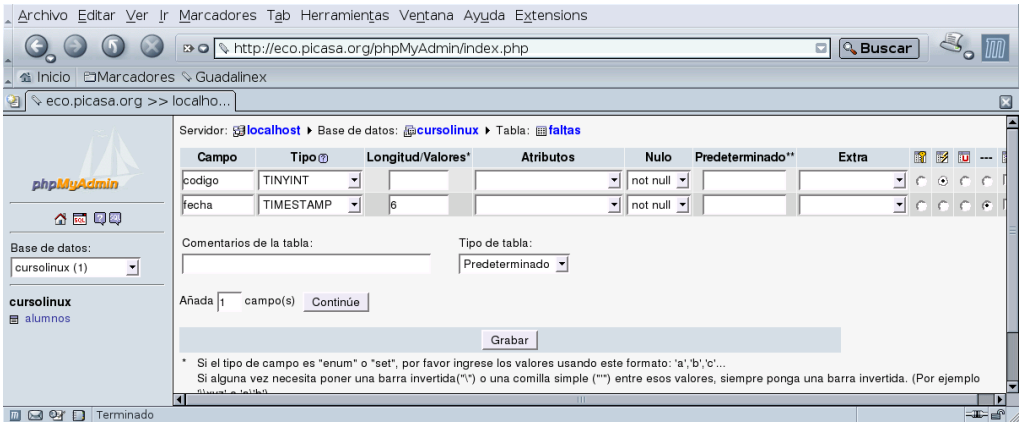


```
mysql> CREATE TABLE ‘alumnos‘ (
‘codigo‘ TINYINT UNSIGNED NOT NULL AUTO_INCREMENT ,
‘nombre‘ VARCHAR( 20 ) NOT NULL ,
‘apellido1‘ VARCHAR( 15 ) NOT NULL ,
‘apellido2‘ VARCHAR( 15 ) ,
‘curso‘ ENUM( ’1’, ’2’, ’3’, ’4’ ) DEFAULT ’1’ NOT NULL ,
‘dni‘ CHAR( 8 ) NOT NULL ,
PRIMARY KEY ( ‘codigo‘ ) ,
INDEX ( ‘apellido1‘ , ‘apellido2‘ ) ,
UNIQUE (
‘dni‘
),
FULLTEXT (
‘dni‘
)
```

Creemos ahora la segunda tabla. Para ello, en la parte izquierda del navegador, pulsemos sobre la base de datos sobre la que vamos a crear la nueva tabla (`cursolinux`) y en la página mostrada introduciremos el nombre de la tabla (`faltas`) así como el número de campos (2)



y [**Continúe**]. Es el momento de introducir los nombres de los campos y su tipo



Para volver a la página inicial, pulsaremos sobre **Página de Inicio**  (parte superior derecha de la ventana)

```
mysql> CREATE TABLE ‘faltas‘ (
‘codigo‘ TINYINT NOT NULL ,
‘fecha‘ TIMESTAMP( 6 ) NOT NULL ,
INDEX ( ‘codigo‘ )
);
```

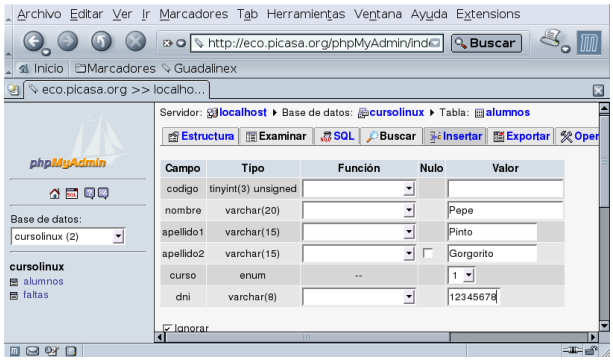
En este caso, acabamos de usar un nuevo tipo de columna. Se trata del tipo fecha `timestamp` de longitud 6, eso significa que nuestras fechas serán del tipo `AAMMDD` (véase la tabla 20.7 en la página 442). El resto no presenta problemas, indexamos por el código para optimizar las búsquedas y no permitimos campos vacíos.

Ya vimos que usando el comando `INSERT` (en la página 433) podíamos hacerlo, pero en general y en espera de usar php y formularios Web, la mejor forma de hacerlo es usando un archivo auxiliar que contenga los datos. En primer lugar, vamos a usar el interfaz Web de phpMyAdmin, pulsemos sobre la tabla `alumnos` para acceder a

Figura 20.1: Tablas y phpMyAdmin

y pulsemos en la pestaña [**Insertar**] para introducir nuestros alumnos:

- Pepe Pinto Gorgorito de 1º de ESO y dni 12345678
- Pilar Pan y Agua de 2º de ESO y dni 87654321

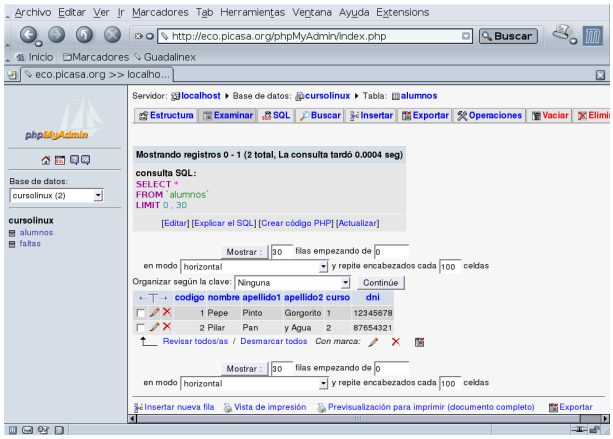



```
mysql> INSERT INTO ‘alumnos‘ ( ‘codigo‘ , ‘nombre‘ ,
‘apellido1‘ , ‘apellido2‘ , ‘curso‘ , ‘dni‘ )
VALUES (
'', ’Pepe’, ’Pinto’, ’Gorgorito’, ’1’, ’12345678’
);
```

Notar que:

1. Dejamos el campo código sin introducir
2. Marcamos la opción de **Insertar un nuevo registro**.

Cuando tengamos nuestros alumnos introducidos, pulsando sobre la pestaña [**Examinar**] podremos visualizarlos. Desde aquí podemos ordenar los registros, editarlos, ...



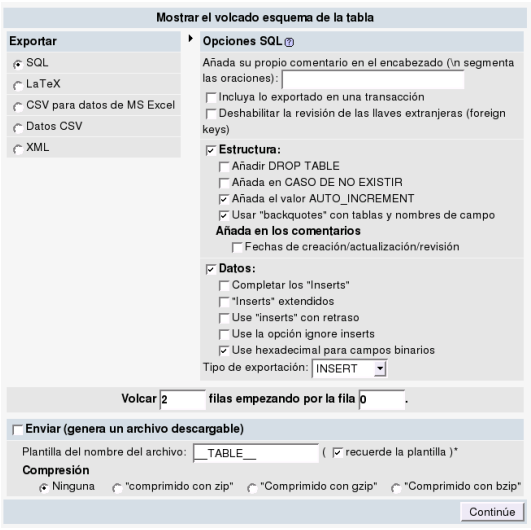
Desde esta página y si introducimos algunos registros más, podemos comprobar la potencia y versatilidad del comando `SELECT`. Las pestañas del gráfico 20.1 en la página anterior no comentadas aún, nos permiten[32]:

Estructura nos permite modificar la estructura de la tabla (es la captura gráfica 20.1)

SQL nos permite ejecutar sentencias en modo comando

Buscar para realizar búsquedas en nuestras tablas.

Exportar nos permite exportar los datos y la estructura de nuestra tablas a varios formatos estándar.



[32]Para bases de datos, las opciones son: **Estructura**, **SQL**, **Exportar**, **Buscar**, **Generar una consulta** y **Eliminar**. Su funcionalidad es similar a la que comentamos sobre tablas.

Operaciones para realizar distintas operaciones sobre ella: cambiarle el nombre, el orden, copiarla a otra base de datos, tabla, etc

Vaciar con esta opción podemos eliminar los registros de nuestra tabla.

Eliminar para borrar la tabla.

➥ **Insertar registros desde un fichero.** Vamos a realizar algunas funciones más con el par de alumnos que hemos introducido

1. Desde la pestaña **Exportar** exportemos los datos (no la estructura) a formato SQL enviándolos a un archivo descargable de nombre `datos_alumnos.sql`.

   Si lo visualizáis después de exportarlos, su contenido[33] será similar a:

   ```
   # phpMyAdmin SQL Dump
   # version 2.6.1-pl3
   # http://www.phpmyadmin.net
   #
   # Servidor: localhost
   # Tiempo de generación: 10-04-2005 a las 08:51:42
   # Versión del servidor: 3.23.58
   # Versión de PHP: 4.3.11
   #
   # Base de datos : ‘cursolinux‘
   #
   #
   # Volcar la base de datos para la tabla ‘alumnos‘
   #
   INSERT INTO ‘alumnos‘ VALUES (1, ’Pepe’, ’Pinto’, ’Gorgori-
   to’, ’1’, ’12345678’);
   INSERT INTO ‘alumnos‘ VALUES (2, ’Pi-
   lar’, ’Pan’, ’y Agua’, ’2’, ’87654321’);
   ```

   como siempre, lo importante son las dos últimas líneas, el resto son comentarios. Con este fichero podremos insertar datos en nuestra tabla `alumnos`.

2. Borremos los datos introducidos desde la pestaña **Vaciar** (la pestaña **Examinar** se pondrá “negra”)

3. Volvemos al modo comando, introduzcamos los datos anteriores ejecutando el comando

   ```
   # mysql -u root -p cursolinux < datos_alumnos.sql
   Enter password:
   ```

   De esta forma, estamos introduciendo datos desde un fichero de texto que podría estar preparado por nosotros previamente. Podemos comprobar que todo ha ido bien recargando phpMyAdmyn y tras llegar a la ventana **Examinar**, ver que tenemos de nuevo a nuestros dos alumnos.

4. Exportemos ahora de nuevo la tabla, pero esta vez dejando marcada la opción de exportar la **Estructura**. El resto igual que antes. El resultado final del fichero de nombre `alumos.sql` será[34]

[33] Si no se exporta a un fichero externo se muestra el resultado en el navegador.
[34] Aquí hemos eliminado algunos comentarios.

```
CREATE TABLE ‘alumnos‘ (
‘codigo‘ tinyint(3) unsigned NOT NULL auto_increment,
‘nombre‘ varchar(20) NOT NULL default '',
‘apellido1‘ varchar(15) NOT NULL default '',
‘apellido2‘ varchar(15) default NULL,
‘curso‘ enum(’1’,’2’,’3’,’4’) NOT NULL default ’1’,
‘dni‘ varchar(8) NOT NULL default '',
PRIMARY KEY (‘codigo‘),
UNIQUE KEY ‘dni‘ (‘dni‘),
KEY ‘apellido1‘ (‘apellido1‘,‘apellido2‘),
FULLTEXT KEY ‘dni_2‘ (‘dni‘)
) TYPE=MyISAM AUTO_INCREMENT=3 ;
#
# Volcar la base de datos para la tabla ‘alumnos‘
#
INSERT INTO ‘alumnos‘ VALUES (1, ’Pepe’, ’Pinto’, ’Gorgori-
to’, ’1’, ’12345678’);
INSERT INTO ‘alumnos‘ VALUES (2, ’Pi-
lar’, ’Pan’, ’y Agua’, ’2’, ’87654321’);
```

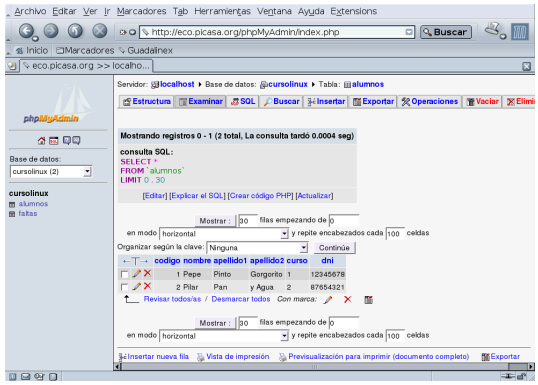
Usando este fichero podremos crear la tabla alumnos e insertar datos, todo de una sola vez. Veamos cómo.

5. Borremos ahora la tabla entera, datos y estructura pulsando sobre [**Eliminar**]. Tras recargar la aplicación, comprobaremos que sólo tenemos ya la tabla `faltas`.

6. Creemos la tabla `alumnos` e insertemos los dos datos usando el comando

```
# mysql -u root -p cursolinux < alumnos.sql
Enter password:
```

De nuevo, si recargamos phpMyAdmin comprobaremos que todo ha funcionado como debe.



20.5. PHP y MySQL: páginas web dinámicas.

20.5.1. Más sentencias de PHP

En este apartado, vamos a enumerar algunas de las sentencias de PHP que nos permiten conectar con MySQL. Sólo vamos a enumerar las que vamos a usar en el ejemplo que analizaremos después[35] (apartado 19.4.1 en la página 413). Os remitimos a la documentación añadida sobre

[35] De hecho la mayoría de los ejemplos se han tomado de él.

PHP para ampliar sobre todas ellas (`http://es2.php.net/manual/es/ref.mysql.php`) así como para conocer todas las que hay.

Conexión y desconexión con la base de datos

`mysql_connect(host, usuario, password)` Abre una conexión a un servidor MySQL[36]. Los tres argumentos son opcionales. Si falta alguno se toman los valores por defecto:

`host`	'localhost'
`usuario`	que conectará con la base de datos
`password`	vacía

↬ Ejemplo

```
<?php
//variables para almacenar los datos de la conexión
//se puede optar por declararlas o por escribir los datos di-
rectamente
$mysql_server="localhost";
$mysql_login="thales";
$mysql_pass="contraseña";
$c = mysql_connect($mysql_server,$mysql_login,$mysql_pass){
or die ("No he podido conectar");
}
```

Notar que hacemos uso de una función "añadida", se trata de la función `die()`. Se ejecuta en el caso de que falle la conexión (`or`). En ese caso, finaliza el script y se muestra en el navegador el mensaje especificado.

`mysql_select_db(basededatos, conexión)` selecciona la base de datos sobre la que se va a trabajar, asociada con el identificador de enlace especificado[37].

↬ Ejemplo

```
// Nombre de la base de datos que contiene todos los datos
// necesarios para la práctica, usamos una constante
//También se podría escribir el nombre directamente
define("base_de_datos", "cursolinux");
//Seleccionamos la base de datos con la que trabajar
mysql_select_db(base_de_datos,$c) or die("No se puede selec-
cionar la
                                                    base de da-
tos");
```

`mysql_close(conexión)` cierra el enlace con MySQL con la conexión especificada[38].

↬ Ejemplo: `mysql_close($c);`

[36] Devuelve un identificador de enlace positivo si tiene éxito, o falso si error.
[37] Si no se especifica un identificador de enlace, se toma como identificador de enlace el último que se ha abierto.
[38] Si no se especifica, se asume el último enlace.

Preparación de la consulta

`mysql_query(consulta, conexión)` ejecuta la sentencia de MySQL especificada en el primer parámetro, sobre el identificador de conexión del segundo parámetro[39].

↬ Ejemplo:

```
//usar las variables que siguen no es "obligatorio"
//pero no está mal parametrizar el código

//tabla en la que hacer la consulta
$tabla="alumnos";

//campos a mostrar, en este caso todos
$campos="*";

//cadena para la consulta
$consulta="SELECT ".$campos." FROM ".$tabla;

//ejecuta la consulta especificada
$resultado= mysql_query($consulta,$c);
```

Si no usamos variables, escribiríamos la sentencia equivalente[40]

```
//ejecuta la consulta especificada
$resultado= mysql_query(SELECT * FROM alumnos,$c);
```

Recuperación de resultados

`mysql_num_fields(cursor)` devuelve el número de campos de un resultado

↬ Ejemplo:

```
//se almacena en la variable el
//número de campos solicitado
$numero_campos=mysql_num_fields($resultado);
```

`mysql_num_rows(cursor)` devuelve el número de filas de un resultado

↬ Ejemplo:

```
//se almacena en la variable el número
//de filas obtenido en el resultado
$numero_filas=mysql_num_rows($resultado);
```

`mysql_field_name(cursor,índice)` Devuelve el nombre del campo de índice especificado en un resultado.

↬ Ejemplo: para obtener el nombre del segundo campo de una consulta podemos usar

```
mysql_field_name($resultado,2)
```

`mysql_fetch_row(cursor)` devuelve un array enumerado (el primer elemento se enumera con 0) en que se selecciona una fila de datos del resultado, o falso si no quedan más líneas. Con esta función y un bucle es fácil mostrar todos los campos (y registros) de una consulta.

[39] Si no se especifica un identificador de enlace, se trabaja con el último abierto.
[40] Notar que no es necesario escribir el ; final.

↬ Ejemplo:

```
//creamos una matriz enumerada que contiene los datos
$datos=mysql_fetch_row($resultado);
//número de campos que contiene
$numero_campos=mysql_num_fields($resultado);
//Mostramos los datos
for ($i=0;$i<$numero_campos;$i++){
    echo $datos[$i]." ";
}
```

`mysql_fetch_array(cursor,tipo_resultado)` devuelve un array asociativo en que se selecciona una fila de datos del resultado o falso si no quedan más líneas. Con `mysql_fetch_array` y un bucle es fácil mostrar todos los campos (y registros) de una consulta. Se trata de una versión extendida de la anterior ya que "además de guardar los datos en el índice numérico de la matriz, guarda también los datos en los índices asociativos, usando el nombre de campo como clave" [12]. El segundo parámetro es opcional e indica el tipo de array que devuelve[41].

↬ Ejemplo: Con el código siguiente, podemos recorrer varias filas de una consulta y mostrarlas en pantalla

```
# establecemos un bucle que recoge en un array
# cada una de las líneas del resultado de la
# usamos <<mysql_fetch_array>>  y para evitar duplicados
# optamos por el parámetro MYSQL_ASSOC
while ($registro= mysql_fetch_array($resultado,  MYSQL_ASSOC)){

   // insertamos un salto de línea para cada fila
   echo "<br>";

   # establecemos el bucle de lectura del ARRAY
   # con los resultados de cada línea
   foreach($registro as $clave => $valor){
          echo $valor." ";
    }
}
```

Notar que la condición del bucle `while` es cierta hasta que lleguemos a la última línea.

`mysql_free_result(cursor)` cierra la conexión establecida con `mysql_connect()`. Sólo puede ser interesante utilizarla si nos preocupa la posibilidad de que usemos demasiada memoria en la ejecución del script PHP, ya que toda la memoria del resultado especificado en `cursor` será automáticamente liberada.

↬ Ejemplo:

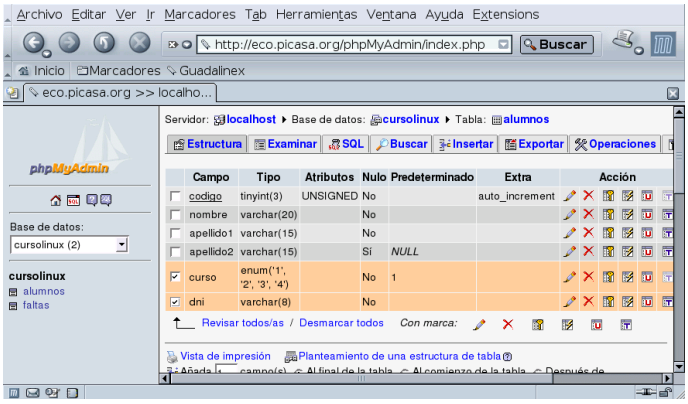
```
//libero los recursos de las consultas
mysql_free_result($resultado);
```

[41] Puede tomar los valores: `MYSQL_NUM`, `MYSQL_ASSOC` y `MYSQL_BOTH`.

20.5.2. Un ejemplo

Vamos a ver, de forma bastante simplificada, cómo podemos usar conjuntamente MySQL y PHP. Para esto vamos a echar mano de la Web creada en el capítulo sobre PHP (apartado 19.4.1 en la página 413) y de la base de datos `cursolinux` que tenemos creada (suponiendo que hemos introducido los datos previamente). En primer lugar vamos a "simplificarla" un poco eliminando algunos campos (`grupo` y `dni`) para centrarnos sólo en las cuestiones más elementales.

Eliminemos pues, los campos. Desde phpMyAdmin, nos situamos sobre estructura y tras marcar las casillas de verificación que hay a la derecha de `grupo` pulsaremos sobre 




```
mysql> ALTER TABLE ‘alumnos‘ DROP ‘curso‘;
Query OK, 2 rows affected (0.01 sec)
Registros: 2  Duplicados: 0  Peligros: 0
mysql> ALTER TABLE ‘alumnos‘ DROP ‘dni‘;
Query OK, 2 rows affected (0.01 sec)
Registros: 2  Duplicados: 0  Peligros: 0
```

Después lo haremos con el campo `dni`.

Pretendemos que en la página principal se nos muestre un listado de los alumnos de un grupo y que, seleccionando el que deseeemos, podamos introducir las faltas de ese alumno para la fecha en curso.

Realmente tendríamos que crear primero un formulario que nos permitiera ir introduciendo los datos de la base de datos, pero intentar abarcar todos los aspectos del funcionamiento conjunto de MySQL y PHP es algo que escapa de los objetivos de este curso y que sería materia suficiente para un curso específico.

Nuestra utilidad va a constar casi de los mismos ficheros[42] que cuando se estudió en el tema anterior:

- `estilo.css` es la misma hoja de estilo.
- Los ficheros `index.php` (fichero principal de la aplicación), `cabecera.php`, `menu-izq.php` e `inicio.php` se han mantenido prácticamente iguales

[42] Estarán a vuestra disposición en el servidor todos los ficheros, en un archivo comprimido de nombre `mysql-php.tgz`

sólo se han cambiado los textos que se muestran en pantalla para adecuarlos a la nueva situación. Si pulsamos sobre **Insertar** o **Listar** obtendremos una página de contenido similar, pero de funcionalidad final diferente.

- Con:

  Insertar se carga el fichero `pagina1.php` y nos permite poner la falta del día en curso a uno de los alumnos.

  Listar se carga `pagina2.php` y nos permite visualizar las faltas de asistencia del alumno seleccionado.

  El contenido de ambos ficheros sí que cambia, es:

```
<h2>
   Introducir faltas a un alumno
</h2>
<?php
   //variable para la óaccin a realizar
   $accion="introducir-faltas";

   //listado úcomn a la óaplicacin
   require "listar-grupo.php";
?>
```

Listado 20.1: pagina1.php

```
<h2>
   Listar las faltas de un alumno
</h2>
```

```php
<?php
   //variable para la óaccin a realizar
   $accion="listar-faltas";

   //listado úcomn a la óaplicacin
   require "listar-grupo.php";
?>
```

Listado 20.2: pagina2.php

Este par de ficheros tienen poco que comentar. Se declara la variable `$accion` que va a permitir el que con un solo fichero (`listar-grupo.php`) realicemos acciones diferentes dependiendo de la página desde la que se cargue (más adelante volveremos sobre ella). Es importante reseñar la ventaja que supone la posibilidad de reutilizar el código en ambos archivos.

Comencemos con el primer fichero "nuevo" y dejemos para después el fichero `listar-grupo.php`. Se trata del fichero que nos va a permitir conectar con la base de datos ***cursolinux*** y obtener o modificar los datos de las tablas que la componen. Se tendrá que incluir en los script PHP en los que conectemos con la base de datos.

```php
<?php
   //Variables para almacebar los datos de la óconexin
   //se puede optar por declararlos o por escribir los
   //datos directamente en los scripts php
   $mysql_server="localhost";
   $mysql_login="thales"; //este valor hay que adecuarlo a vuestro sistema
   $mysql_pass="ñcontrasea"; //poner íaqu la ñcontrasea de la base de datos

   // Nombre de la base de datos que contiene todos los datos
   // necesarios para la áprctica, usamos una constante
   define("base_de_datos", "cursolinux");

   //variable que recoge el identificador de óconexin
   $c;

   // óFuncin que realiza la óconexin con el gestor de base de datos
   // y selecciona la base de datos con la que vamos a trabajar
   // Anteponemos & a la variable para indicar que se pasa
   // por referencia
   function conectar_bd(&$c)
      {
      //las definimos como globales para trabajar con ellas en la ófuncin
      global $mysql_server, $mysql_login, $mysql_pass;

      // Nos conectamos con el gestor MySql y seleccionamos una BD
      $c = mysql_connect($mysql_server,$mysql_login,$mysql_pass) or die("No se puede conectar");
      mysql_select_db(base_de_datos,$c) or die("No se puede seleccionar la base de datos");
      return $c;
      }

   // óFuncin que nos desconecta del gestor de base de datos
   function desconectar_bd(&$c)
      {
      mysql_close($c);
      }
?>
```

Listado 20.3: comun.inc

Todo el código que aparece en este fichero se ha analizado ya en la subsección 20.5.1 en la página 448. Llega el momento de la verdad, analicemos el fichero `listar-grupo.php`

```
<script type="text/javascript">
<!--
   //ócdigo javascript necesario para la ventana emergente
   function abrir(pagina) {
      window.open(pagina,'ventanaFaltas','scrollbars=no,resizable=yes,width
          =200,height=280,status=no,location=no,toolbar=no');
   }
// -->
</script>

<?php
   //datos comunes
   require "comun.inc";

   //establecemos la conexion con el servidor
   conectar_bd($c);

   //tabla en la que hacer la consulta
   $tabla="alumnos";

   //campos a listar, en este caso todos
   $campos="*";

   //cadena para la consulta
   $consulta="SELECT␣".$campos."␣FROM␣".$tabla;

   //ejecuta la consulta especificada
   $resultado= mysql_query($consulta,$c);

   //únmero de campos solicitados
   $numero_campos=mysql_num_fields($resultado);

   //Aviso sobre ócmo proceder
   echo "<h3>Marca␣sobre␣el␣c&oacute;digo</h3>";

   // Creamos una cabecera de una tabla (ócdigo HTML)
   echo "<table␣align=center␣border=2>";

   //comienzo en cero para que se muestre el ócdigo
   for ($i = 0; $i < $numero_campos; $i++){
      echo "<td␣><b><center> ";
      echo ucfirst(mysql_field_name($resultado,$i));
      echo "</b></td></center> ";
   }
   echo "</tr>";

   # establecemos un bucle que recoge en un array
   # cada una de las ílneas del resultado
   # usamos mysql_fetch_array y para evitar duplicados
   # optamos por el áparmetro MYSQL_ASSOC
   while ($registro= mysql_fetch_array($resultado, MYSQL_ASSOC)){

      // insertamos un salto de ílnea en la tabla HTML
      echo "<tr>";

```

```
      # establecemos el bucle de lectura del ARRAY
      # con los resultados de cada ílnea
      # y encerramos cada valor en etiquetas <td></td>
      # para que aparezcan en celdas distintas de la tabla
60    foreach($registro as $clave => $valor){

         if ( $clave == "codigo" ) {
            echo "<td><center><a href=javascript:abrir('$accion.php?codigo=
                $valor');>".$valor."</a></center></td>";
            }
65       else
            {
            echo "<td><center>",$valor,"</center></td>";
         }
      }
70 }

   echo "</table>";

   //libero los recursos de las consultas
75 mysql_free_result($resultado);

   // cerramos la óconexin
   desconectar_bd($c);
?>
```

Listado 20.4: listar-grupo.php

La primera parte es código javascript. Nos permite abrir una ventana emergente en la que se nos informa del resultado y se ha puesto sólo con la intención de que no se pierda de vista la página principal.

Aparece una función[43] nueva de PHP en la línea:

```
echo ucfirst(mysql_field_name($resultado,$i));
```

`ucfirst('cadena')` pone en mayúsculas el primer carácter la cadena si es un carácter alfabético. Por ejemplo `ucfirst('thales')` daría como resultado "Thales". El bucle contiene un `if`

```
if ( $clave == "codigo" ) {
       echo "<td><center><a href=javascript:abrir(
          '$accion.php?codigo=$valor');>".$valor."</a></center></td>";
    }else{
       echo "<td><center>",$valor,"</center></td>";
    }
```

que nos permite diferenciar el primer campo del resto (campo código), de esa forma creamos el enlace sólo con él.

El enlace creado llama a la función javascript (`abrir`) que permite ver los resultados en la ventana emergente. Notar además que en el argumento de la llamada se introduce la variable `$accion`. De esa manera, cuando se ejecuta desde `pagina1.php`, `$accion=introducir-faltas` y sin embargo, si se hace desde `pagina2.php` su valor es `listar-faltas`. La consecuencia es que si bien el código es común, dependiendo de dónde se llame, ejecutaremos el script php adecuado. Es decir, si

`$accion=introducir-faltas` se ejecuta el script `introducir-faltas.php`

[43] El número de funciones para tratar cadenas de PHP es muy amplio: `http://es2.php.net/manual/es/ref.strings.php`

`$accion=listar-faltas` se ejecuta el script `listar-faltas.php`

Además, en ambos casos le pasamos al script una variable (`$codigo`) cuyo valor coincide con el código del alumno sobre el que pulsemos. El resto se reduce a cuestiones ya estudiadas, o a crear la tabla en que se muestran nuestros alumnos.

El script que se lista a continuación nos permite poner la falta de asistencia (la fecha del día en curso) al alumno sobre cuyo código hayamos pulsado:

```
<?php
   //datos comunes
   require "comun.inc";

   //tabla en la que introducir las faltas
   $tabla="faltas";

   //capturamos la fecha del sistema
   $fecha=date("ymd");

   //almacenamos la variable ócdigo
   $codigo=$_GET['codigo'];

   //ñAadimos el nuevo registro
   //Primero establecemos la óconexin con el servidor
   conectar_bd($c);
   $consulta="INSERT␣".$tabla."␣(codigo,fecha)␣VALUES␣('$codigo','$fecha')";
   mysql_query($consulta, $c);

   //comprobamos el resultado de la óinsercin
   //el ócdigo de error CERO significa NO ERROR
   if (mysql_errno($c)==0){
         echo "<p><br><p><br><center><h2>Registro␣a&ntilde;adido</b></H2></
             center>";
      }else{
         echo "<p><br><center><h2>Se␣ha␣producido␣un␣error<br></h2></center>
             ";
      }

   # cerramos la óconexin
   desconectar_bd($c);
?>

<!-- Formulario que permite cerrar la ventana -->
<center>
   <form>
      <font size=2 face="arial">
         <input type="button" value="Cerrar␣ventana" name="B1" onclick="
             window.close()">
      </font>
      </form>
</center>
```

Listado 20.5: introducir-faltas.php

Una nueva función (`date()`) en la línea

```
$fecha=date("ymd");
```

con ella capturamos la fecha del sistema. Además, al pasarle los parámetros "ymd" la almacenamos en la variable `$fecha` de la forma adecuada (`AAMMDD`) para insertarla en el campo `fecha` de la tabla `faltas`. Con

```
$consulta="INSERT ".$tabla." (codigo,fecha) VALUES ('$codi-
go','$fecha')";
mysql_query($consulta, $c);
```

creamos la cadena que contiene la sentencia de inserción. Recordar que el campo `codigo` es el que nos permite establecer la relación uno a muchos entre ambas tablas (`alumnos` y `faltas`). Como pasamos el código de alumno en la URL, sólo tenemos que recuperarlo en una variable y, junto a la fecha del sistema, insertamos ambos valores en la base de datos `faltas`.

Por último, otro añadido (que no es imprescindible):

```
//comprobamos el resultado de la inserción
//el error CERO significa NO ERROR
if (mysql_errno($c)==0){
 echo "<p><br><p><br><center><h2>Registro a&ntilde;adido</b></H2></center>";
}else{
 echo "<p><br><center><h2>Se ha produci-
do un error<br></h2></center>";
}
```

La función `mysql_errno($c)` nos devuelve el código de error para la última función llamada. Si bien aquí no está muy "explícito", su uso nos permite (usando ese código de error) detectar a qué puede ser debido el que un registro no se haya insertado correctamente.

Llegamos al último archivo, con él listamos todas las faltas del alumno seleccionado. De nuevo tendremos que pasar en la URL el valor del código de ese alumno que recuperaremos en una variable (`$codigo=$_GET['codigo'];`).

```
<?php
   //datos comunes
   require "comun.inc";

   //establecemos la óconexin con el servidor
   conectar_bd($c);

   //almacenamos la variable ócdigo
   $codigo=$_GET['codigo'];

   //tabla en la que hacer la consulta
   $tabla="alumnos";

   //campos a mostrar
   $campos="nombre,apellido1,apellido2";

   //cadena para la consulta
   $consulta="SELECT␣".$campos."␣FROM␣".$tabla."␣WHERE␣codigo=".$codigo;

   //establecemos el criterio de óseleccin
   $resultado= mysql_query($consulta,$c);

   //creamos una matriz enumerada que contiene los datos
   $datos=mysql_fetch_row($resultado);

   //tabla en la que hacer la consulta
   $tabla="faltas";

   //campos a mostrar, en este caso solo el campo fecha
   $campos="fecha";

   //cadena para la consulta
```

```
    $consulta="SELECT UNIX_TIMESTAMP(".$campos.") FROM ".$tabla." WHERE
        codigo=".$codigo;

    //ejecutamos la consulta
    $resultado= mysql_query($consulta,$c);

    //únmero de faltas del alumno
    $numero_filas=mysql_num_rows($resultado);

    //mostramos el nombre de alumno (en negrita)
    echo "<b>";
    for ($i=0;$i<3;$i++){
       echo $datos[$i]." ";
     }

    //mostramos el únmero de faltas
    echo "</b> ha faltado ".$numero_filas." d&iacute;as, la fechas son:<br><
        br>";

    # establecemos un bucle que recoge en un array
    # cada una de las ílneas del resultado de la consulta
    # como óslo constan de un campo áser el de índice 0
    while ($registro= mysql_fetch_row($resultado)){
       echo " ".date("d m Y",$registro[0]).";";
    }

    //retorno de ílnea
    echo "<p><br>";

    //libero los recursos de las consultas
    mysql_free_result($resultado);

    //cerramos la óconexin
    desconectar_bd($c);
?>

<!-- Formulario que permite cerrar la ventana -->
<center>
   <form>
      <font size=2 face="arial">
         <input type="button" value="Cerrar ventana" name="B1" onclick="
             window.close()">
      </font>
   </form>
</center>
```

Listado 20.6: listar-faltas.php

Cambia la cadena de consulta respecto a lo que habíamos usado hasta ahora

```
//cadena para la consulta
$consulta="SELECT ".$campos." FROM ".$tabla." WHERE codigo=".$codigo;
```

ya no nos interesan todos los registros, sino sólo aquellos cuyo código coincide con el de nuestro alumno. Por eso añadimos en ambas consultas la claúsula `WHERE codigo=".$codigo`. Además, tenemos que conseguir que la fecha salga en el formato al que estamos habituados (DD MM AA). Para eso hacemos un par de cambios:

- Con `UNIX_TIMESTAMP(fecha)` recuperamos la fecha en formato de tiempo Unix[44].

[44] Número entero de 32 bits que contiene el número de segundos transcurridos desde la media noche del 1 de enero

- Después, con `date("d m Y",fecha),` le damos el formato que deseamos (al poner `Y` en mayúsculas nos devolverá el año con 4 cifras y no con dos, que es como está almacenado)

El resto de sentencias del fichero se han analizado ya.

de 1970.

Capítulo 21

Moodle y PHP-Nuke

> Aunque PHP compite con ASP de Microsoft, Cold Fusion de Allaire, JSP de Sun, e incluso un primo de código fuente abierto llamado `mod_perl`, realmente se encuentra por encima de prácticamemente todos sus competidores por al menos un año. (*Servidor Apache 2*, Mohammed J. Kabir)

21.1. Entorno virtual de aprendizaje: Moodle

> La palabra Moodle era al principio un acrónimo de *Modular Object-Oriented Dynamic Learning Environment* (Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular), lo que resulta fundamentalmente útil para programadores y teóricos de la educación. También es un verbo que describe el proceso de deambular perezosamente a través de algo, y hacer las cosas cuando se te ocurre hacerlas, una placentera chapuza que a menudo te lleva a la visión y la creatividad. Las dos acepciones se aplican a la manera en que se desarrolló Moodle y a la manera en que un estudiante o profesor podría aproximarse al estudio o enseñanza de un curso en línea. Todo el que usa Moodle es un Moodler.
>
> Ven y ¡moodlea con nosotros!
>
> `http://moodle.org/doc/`

21.1.1. Introducción.

Las plataformas educativas o sistemas de gestión de aprendizaje son paquetes de software que permiten la educación a distancia. Esta fórmula educativa tiene cada vez mayor aceptación y se usa no sólo en educación a distancia sino también en la educación presencial. Ejemplos cercanos de esto son estos cursos y los centros TIC.

Dentro del software libre hay distintos proyectos donde poder elegir. Nosotros hemos optado por Moodle por su potencia y facilidad de uso, razones que determinaron así mismo su uso en estos cursos.

Para conocer qué software hay disponible para este cometido se puede consultar:

- `http://www.linuxjournal.com/article/7817`
- `http://www.elearningworkshops.com/modules.php?name=Web_Links&l_op=MostPopular`
- `http://www.uv.es/ticape/docs/sedelce/mem-sedelce.pdf`.

Además de Moodle, una plataforma que merece mención especial es *Ilias*, en la 2ª URL (documento en PDF) aparece justificada su valía y una guía de cómo instalarla. `http://www.gate.upm.es/`

Moddle es una "plataforma educativa" que se desarrolla bajo licencia GPL, se trata de "un paquete de software para la creación de cursos y sitios Web basados en Internet. Es un proyecto en desarrollo diseñado para dar soporte a un marco de educación social constructivista."

Es sencillo de mantener y actualizar y, salvo el proceso de instalación, no necesita prácticamente de "mantenimiento" por parte del administrador.

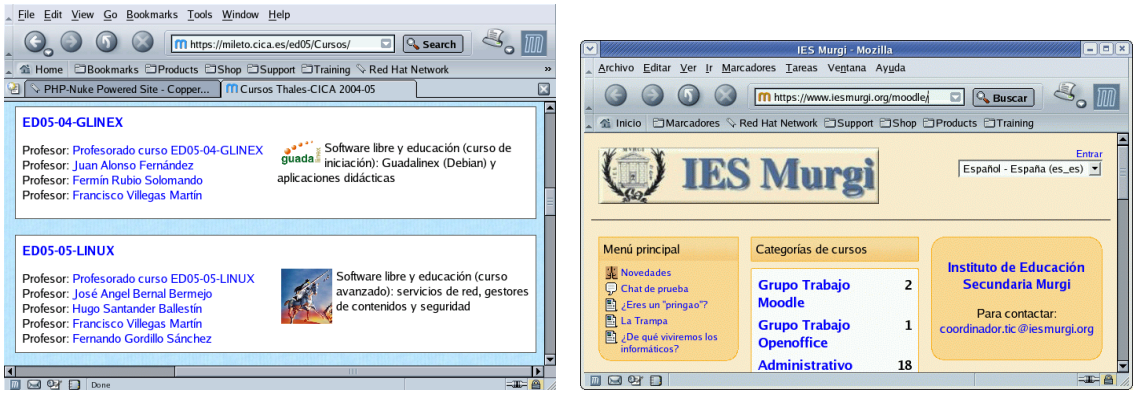


Figura 21.1: Moodle de Mileto e IES Murgi

21.1.2. Instalación

Con objeto de homogeneizar el proceso de instalación en ambas distribuciones antes de iniciar la instalación vamos a adecuar la configuración de Debian para que sea similar a la de Fedora. De esa forma todo lo que sigue será válido para ambas. Para eso, en Guadalinex crearemos un directorio de nombre html dentro del raíz de Apache2

```
#mkdir /var/www/html
```

Y modificaremos el fichero `/etc/apache2/sites-availables/default` para que apunte a él, se trata de cambiar las líneas, ajustando las primeras y comentando la última:

```
DocumentRoot /var/www/html
<Directory /var/www/html>
#RedirectMatch ^/$ /apache2-default/
```

Tras guardar los cambios

```
# apache2ctl restart
```

Existe un paquete para Debian de nombre moodle que nos automatiza el proceso de instalación. Pero en general no es mucho más sencillo que el proceso manual. El hecho de que además el método "general" es válido para ambas distribuciones nos ha llevado a optar por él.

La Web principal del programa es `http://moodle.org/`. Desde allí (sección **Downloads**) podemos bajar la última versión estable (por ahora), se trata de

`moodle-latest-14.tgz`

Para instalar Moodle en nuestro sistema necesitamos tener en funcionamiento: Apache, php[1] y MySQL, y partiremos de esto.

[1] Además es necesario que estén instalados los paquetes (se instalarán con ellos las librerías GD):

`php-gd` en Fedora

`php4-gd` en Debian

Tenemos que garantizarnos que la directiva de Apache `AcceptPathInfo` esté en `On`[2], así que revisemos el fichero de configuración de Apache[3] y añadámosla:

```
AcceptPathInfo on
```

después de cambiar la línea tendremos que reinicar el servidor.

Una vez en nuestra máquina, el proceso de instalación es muy simple y se encuentra bien guiado en las intrucciones de la web de moodle (`http://moodle.org/doc/?file=install.html`). Se resume en:

- Poner el paquete `moodle-latest-14.tgz` en el lugar adecuado, en general será `/var/www/html`, y desempaquetarlo

```
# cp moodle-latest-14.tgz /var/www/html; cd /var/www/html
# tar -xzvf moodle-latest-14.tgz
```

- Asignamos como dueño del directorio al usuario bajo el que se ejecuta el servidor Web[4]

```
# chown www-data moodle
```

- Crear el directorio `moodledata` y ajustarle los permisos de forma adecuada. Es preferible que no sea accesible directamente desde la web. Así que un lugar posible puede ser:

  Debian

```
#mkdir /var/moodledata
#chown www-data /var/moodledata
```

  Fedora

```
# mkdir /var/www/moodledata
# chown apache /var/www/moodledata
```

- Crear la base de datos[5] `moodle`

```
# mysqladmin -u root -p  create moodle
Enter password:
```

[2] Si no está así, no se pueden ver las fotos ni los ficheros que se suban.

[3]

Debian `/etc/apache2/apache2.conf`

Fedora `/etc/http/conf/httpd.conf`

[4]

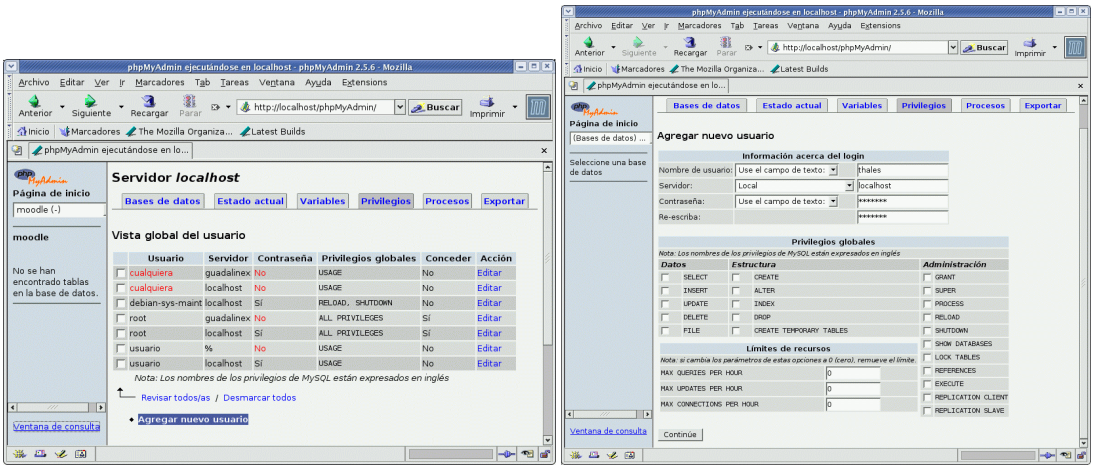
- Es necesario para que el script de instalación pueda hacer los cambios por nosotros. Si no lo hacemos, en el proceso de instalación se nos indicará la forma de corregir este pequeño problema.
- En Fedora será:

```
#chown apache moodle
```

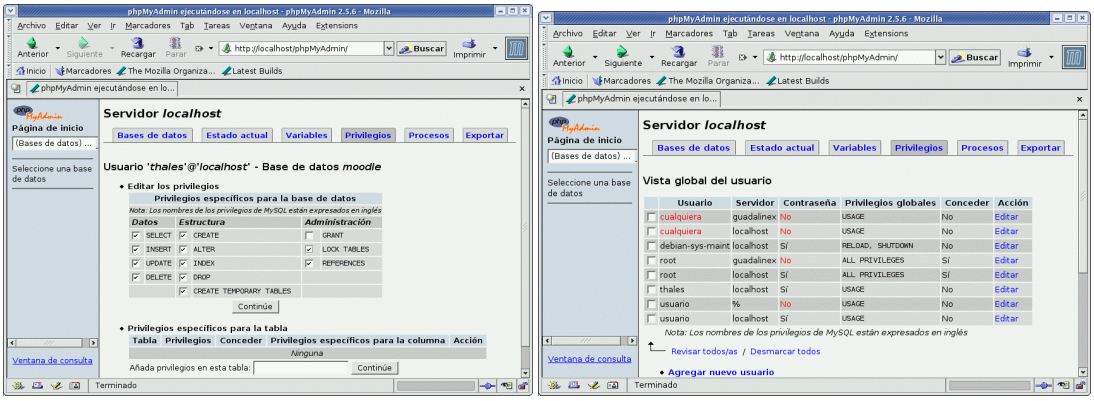
[5] Si se desea se puede usar phpMyAdmin o ejecutar:

```
# mysqladmin -u root -p
mysql> CREATE DATABASE moodle;
mysql> quit
```

De forma opcional, podemos optar porque un usuario (Thales por ejemplo) se conecte a esa base de datos. Para eso en la ventana principal de phpMyAdmin pulsemos sobre **Privilegios→Agregar nuevo usuario**

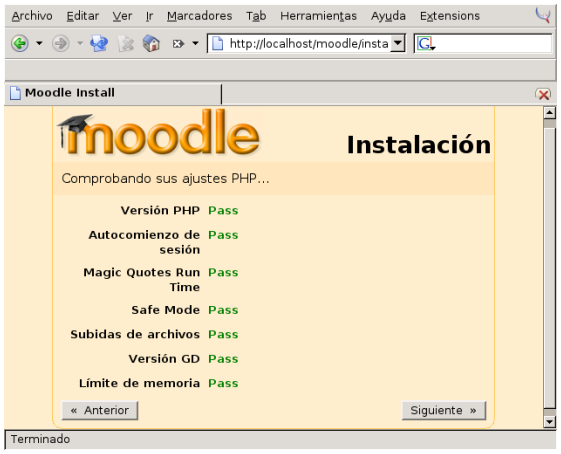


Introducimos el nombre de usuario, optamos porque las conexiones se realicen sólo desde **Local** y contraseña de acceso, y pulsamos sobre **Continúe**. Se nos abren más posibilidades y optamos por `moodle` en la lista desplegable **Añada privilegios en esta base de datos**. Después, marcamos todos los privilegios para ella:

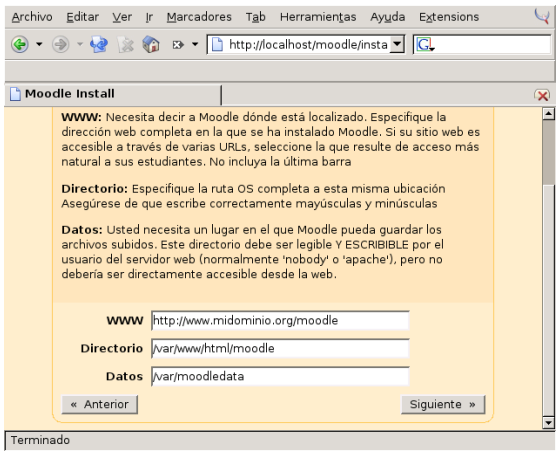


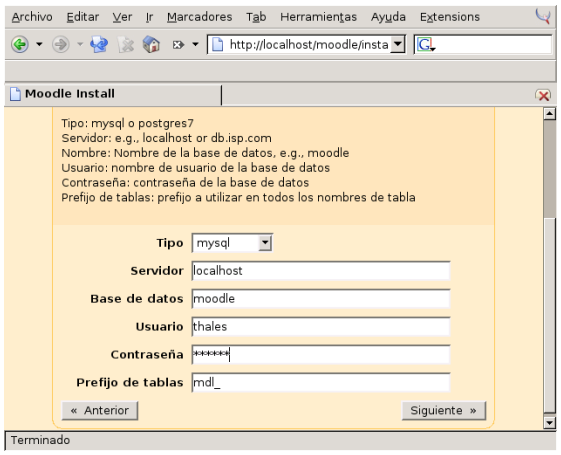
```
# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 55 to server version: 4.0.18-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>GRANT ALL PRIVILEGES ON moodle.* TO thales@localhost
      IDENTIFIED BY 'contraseña';
Query OK, 0 rows affected (0.00 sec)
```

- Comienza la "moodlemanía". Escribamos en un navegador web: `http://localhost/moodle`. Lo primero que debemos hacer será seleccionar el idioma adecuado, a renglón seguido se chequeará nuestro sistema para comprobar si reúne las condiciones necesarias para instalar Moodle



Ahora es el monento de instroducir los datos adecuados a nuestro sistema. En la captura hemos optado por `http://www.midominio.org/moodle` pero si no disponemos de un dominio aquí escribiremos `http://localhost/moodle`. Además debemos adecuar los directorios de datos (moodledata) a la distribución con que trabajemos.





Después seleccionaremos el programa servidor de bases de datos y configuraremos de forma adecuada el resto de campos para que el script de instalación tenga permisos para crear las tablas de la base de datos moodle.

Si apache tiene permisos para escribir en el directorio `/var/www/html/moodle`, el script de instalación creará el fichero `config.php` con los datos que hemos introducido hasta ahora, si no es así nos avisará de que tenemos que hacerlo "a mano"[6]. Ya casi, aceptemos los término de la licencia (GPL claro -:))

[6]Podemos crear ese fichero a partir del fichero `config-dist.php`

```
#cp config-dist.php config.php
```

Editamos el fichero y, además de ajustar los valores adecuados para poder conectar, hay que ajustar los path (por ejemplo) y mejor si restringimos un poco los permisos del directorio de datos:

```
$CFG->dbuser    = 'thales';
$CFG->dbpass    = 'password';
$CFG->wwwroot = 'http://localhost/moodle';
$CFG->dirroot = '/var/www/html/moodle';
$CFG->dataroot = '/var/www/moodledata';
$CFG->directorypermissions = 0750;
```

Si tenemos un dominio en vez de la línea anterior escribiremos

```
$CFG->wwwroot = 'http://www.midominio.org/moodle';
```

Después, cuando se han creado las tablas de la base de datos y directorios de datos, accederemos a la primera ventana de configuración propiamente dicha. Salvo que nos guste el inglés, lo mejor es seleccionar el castellano[7]. La ayuda de contexto es muy buena[8] , así que sólo comentaremos las variables susceptibles de ser cambiadas desde el principio:

`lang` `Español-España (es_es)`

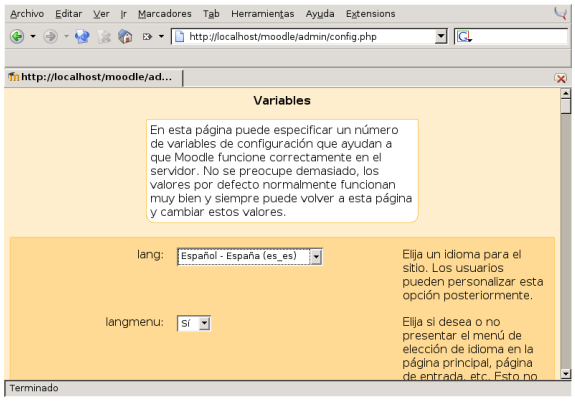
`locale` optaremos por escribir `es_ES`

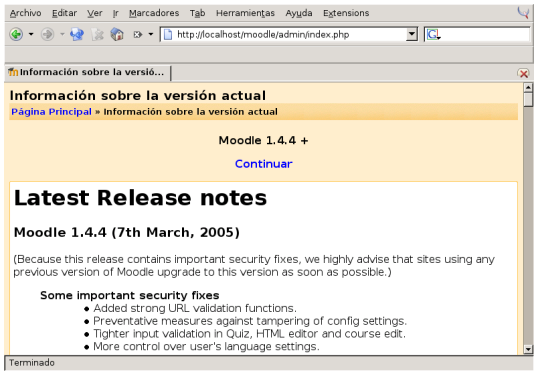
`zip` en general será `/usr/bin/zip`

`unzip` en general será `/usr/bin/unzip`

`country` deberíamos elegir el país por defecto para los nuevos usuarios.

`loginhttps` Deberíamos activar esta opción si hemos montado un servidor seguro, de esa forma el nombre de usuario y contraseña de entrada no viajarán en texto plano. Se iniciará una conexión segura (https) para la página de entrada y, una vez autenticados, se volverá a trabajar con una conexión http normal. **Si no tenemos un servidor seguro en marcha y activamos esta opción no podremos acceder a Moodle**.





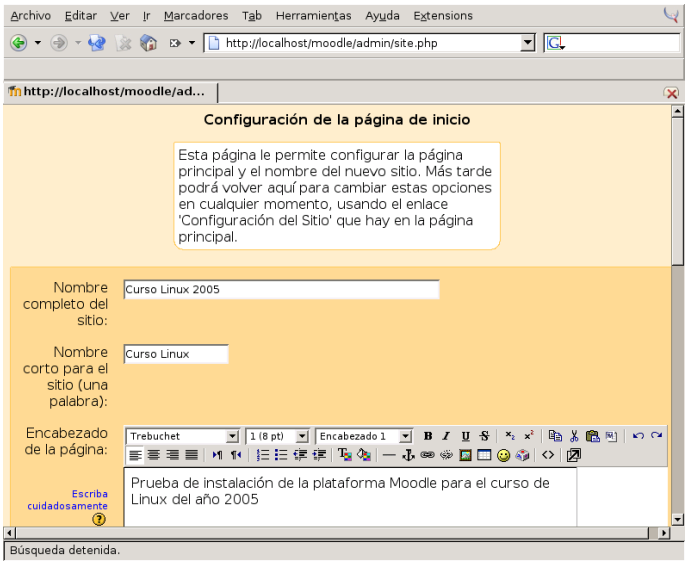
Merece la pena pararse en la que nos informa de los cambios surgidos en esta versión.

Sólo tenemos que ir aceptando en las distintas pantallas que nos van a ir saliendo, con ellas se van creando las tablas necesarias para los distintos módulos y bloques de la aplicación. Configuremos la página de inicio

[7] Todo lo que cambiemos desde este momento podrá ser modificado después. Así que no hay ningún problema si nos equivocamos en algo ahora.

Aunque la captura aparece en castellano, incialmente estará en inglés. Hasta que no se opte por el idioma y se guarden los cambios no la veréis así.

[8] Y en castellano, sólo hay que optar por la variable del idioma, guardar y retroceder luego en el navegador.



Con la lista de forma de encabezado podemos optar por la forma que tendrá la página inicial de la aplicación, podemos elegir entre `Mostrar items de noticias` (captura siguiente), `Mostrar un listado de cursos` o `Mostrar un listado de categorias` (véase la figura 21.1 en la página 462 para las dos últimas). Merece la pena pararse un poco en el magnífico editor de html que integra. Tiene de casi todo.

A continuación debemos configurar la cuenta para el administrador principal. Debemos asegurarnos de darle un nombre de usuario y contraseña seguras, completar adecuadamente los campos relativos a la Ciudad y País, y de una dirección de correo electrónico válida (y cómo no, la foto de rigor).

Posteriormente podremos crear más cuentas de administración.

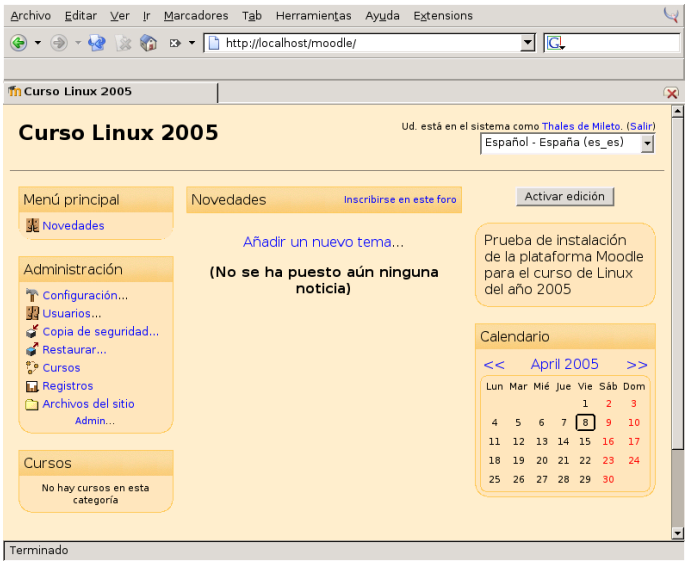
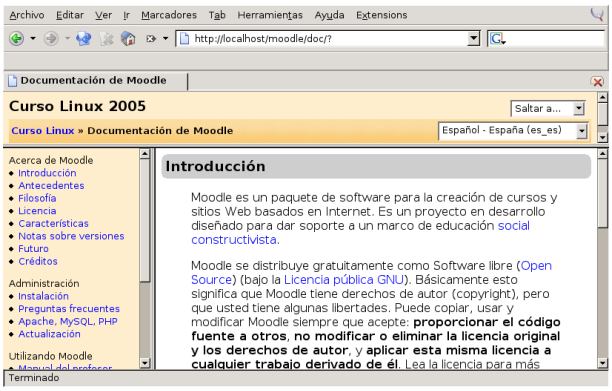


Figura 21.2: Inicio Moodle

Listo, ya tenemos nuestro Moodle en funcionamiento. Si pulsamos sobre **Admin**[9], además de

[9] A la izquierda, en la zona de abajo del bloque de Administración.

poder modificar todas la variables que definen el sitio, podremos acceder a la magnífica ayuda (en castellano) que acompaña al programa.



Antes de seguir es importante tener en cuenta que algunos módulos de Moodle necesitan revisiones continuas para llevar a cabo tareas. Por ejemplo, Moodle necesita revisar los foros para poder enviar copias de los mensajes a las personas que están suscritas. Lo podemos hacer de forma manual usando

```
http://www.midominio.org/moodle/admin/cron.php
```

o bien seguir la indicaciones de la documentación de Moodle:

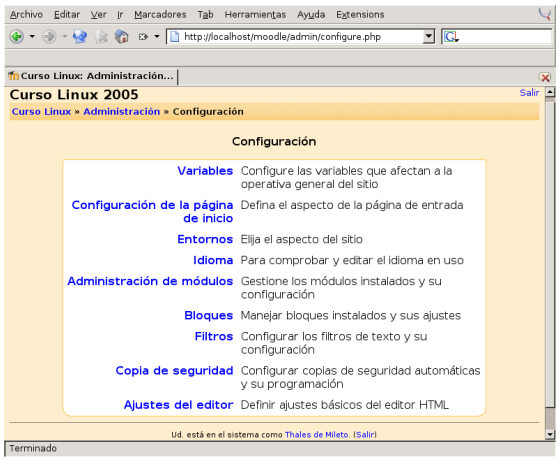
"En un sistema Unix: Use cron. Edite las opciones de cron desde la línea de comandos usando "`crontab -e`" y añada una línea como la siguiente:

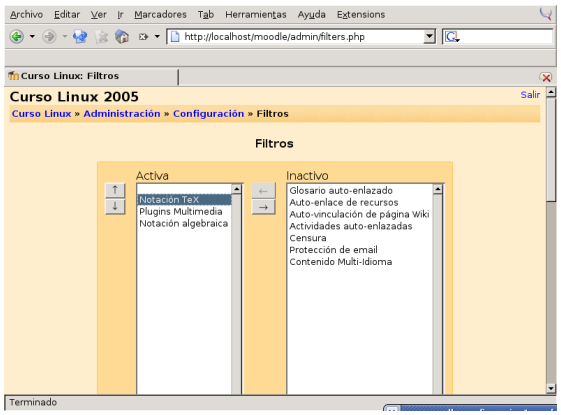
```
*/5 * * * * wget -q -
O /dev/null http://www.midominio.org/moodle/admin/cron.php
```

Normalmente, el comando "crontab" le enviará al editor 'vi'. Se entra en "modo de inserción" presionando "i", después teclee la línea de arriba, luego salga del modo de inserción presionando [ESC]. Se guardan los cambios y se sale tecleando ":wq", se puede salir también sin guardar usando ":q!" (sin las comillas)."

21.1.3. Primeros pasos en la administración.

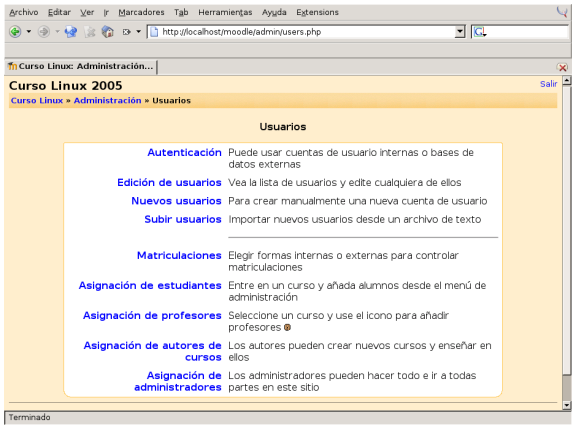
Ya tenemos nuestro entorno instalado. Para que algunas funcionalidades añadidas estén activas debemos hacerlo antes de continuar, así que entramos como administrador del sistema y en la página principal de la aplicación (gráfico 21.2 en la página anterior), pulsamos sobre **Configuración** y después sobre **Filtros**

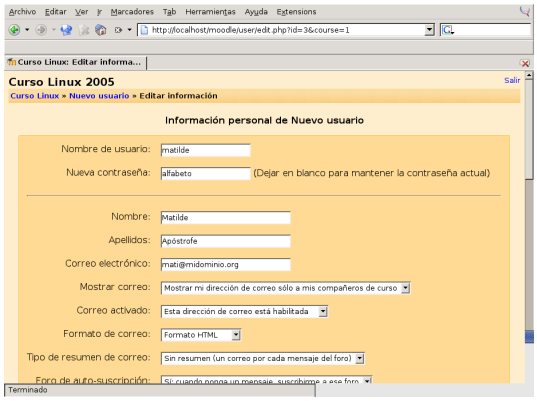




y activamos los que nos interesan (sobre todo Escritura TEX).

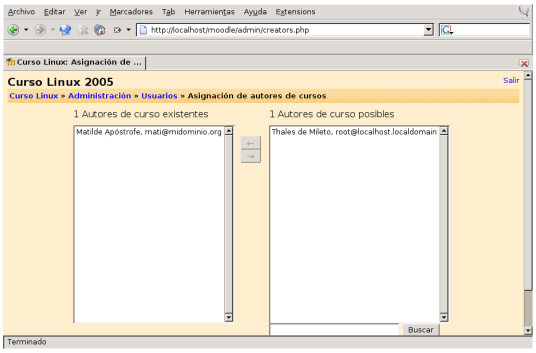
Pero nuestra intención es la de "descargarnos" un poco de trabajo, ya que la administración de la red es tarea ardua y complicada. Nada mejor que implicar al profesorado para que gestione sus cursos. Así que a la "profe" MATILDE APÓSTROFE, jefa del Dpto de Lengua le vamos a dar de alta[10] para que pueda crear y gestionar sus cursos. En la página de inicio pulsamos sobre **Usuarios**[11], obtendremos[12]





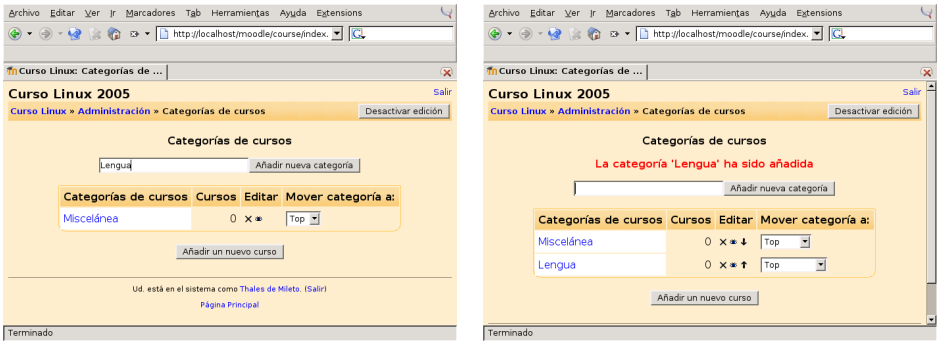
Pulsamos sobre **Nuevos Usuarios** e introducimos los datos, por último guardamos los cambios del formulario de introducción de datos y se nos mostrará la ventana de administración de usuarios.

Volvamos a la página de administración de usuarios y pulsemos sobre **Asignación de autores de cursos**[13], desde ella y pulsando sobre la flecha (←) que hay delante del nombre hagamos que nuestra compañera Matilde pueda crear[14] nuevos cursos y administrar los profesores de esos cursos



Hasta ahora Matilde puede crear cursos pero no asignarlos a una categoría, así que antes de terminar el trabajo como administrador vamos a crearle la categoría Lengua para que sus cursos se sitúen dentro de ella. Pulsamos[15] sobre **Administración** y después sobre **Cursos**

[10] También se puede dar de alta ella sola y nosotros después, sólo tenemos que permitirle esta opción.
[11] También se accede desde **Configuración→Administración→Usuarios**
[12] Como podemos observar el "manual" está incorporado.
[13] De esta forma podrá crear nuevos cursos y enseñar en ellos
[14] No pensemos que este proceso es "obligatorio", el administrador es el root de la plataforma y se puede optar por que sea él solo el que tenga el control de toda ella.
[15] También se puede acceder aquí de otras formas, por ejemplo desde la página inicial de la aplicación.



Una vez creada la categoría Lengua podemos descansar un poco. Pero:

Dos notas a tener en cuenta antes de que pasemos el trabajo a Matilde:

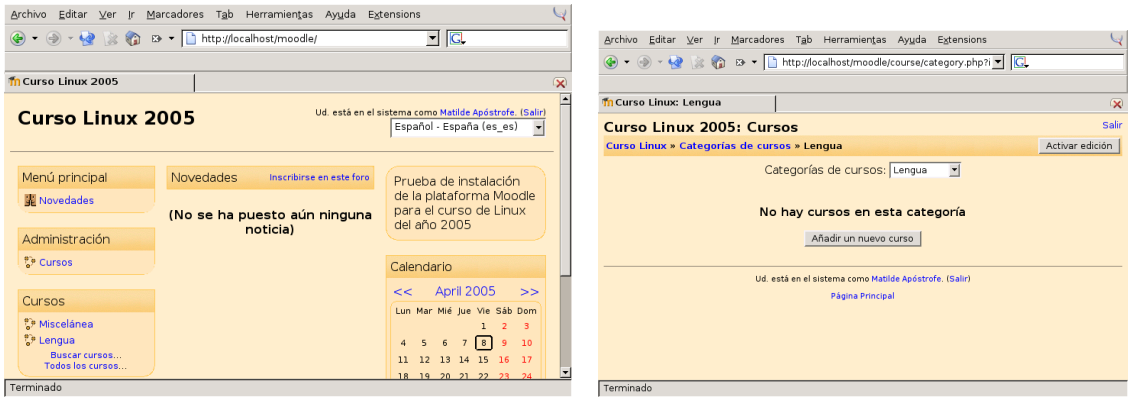
- Para que podamos modificar la plataforma es necesario ver el botón Desactivar edición, es decir, la edición ha de estar activa (Activar edición).
- Es importante que cuando terminemos el trabajo pulsemos sobre **Salir** para finalizar la sesión.

21.1.4. Nuestro primer curso

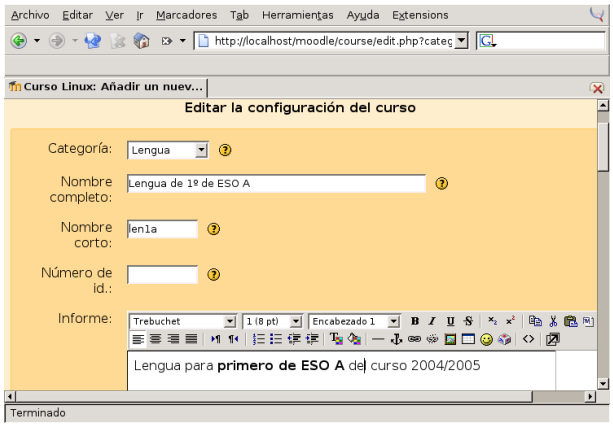
Además de la ayuda de contexto, en Moodle disponemos de una extensa documentación. En castellano podemos acceder a ella desde `http://moodle.org/course/view.php?id=11`. Merece la pena destacar el Manual de usuario y sobre todo el Manual del profesor. Para facilitar el acceso a ellos los hemos puesto también en la web del curso.

Como ya hemos dicho, el uso de Moodle es intuitivo (estáis trabajando con él en los cursos de formación). Pero para iniciarnos con él nada mejor que ver cómo configurar nuestro primer curso.

Le toca el turno a Matilde, va a crear un curso que integrará dentro de la categoría Lengua recién creada. Así que, se autentifica en la plataforma, pulsa sobre **Lengua** y crea su primer curso (Lengua de 1º de ESO A)

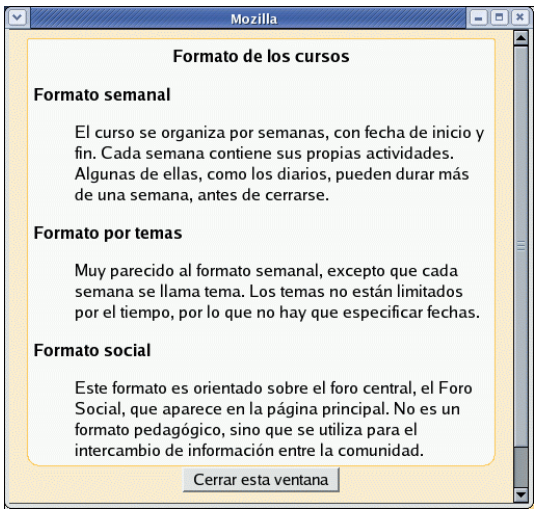


pulsa sobre **Añadir un nuevo Curso**, e introduce los datos adecuados en el formulario que se le presenta:

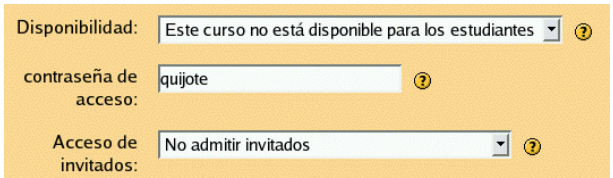


Además de lo reflejado en la captura, comentemos alguno de los campos "más" importantes:

Formato puede ser de tres tipos: semanal, temas y social. Como no sabe muy bien de qué va cada uno pulsa sobre y se abre la ventana de ayuda:



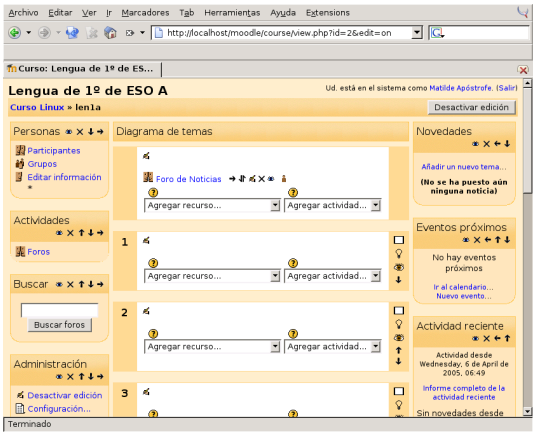
está claro, el que mejor se ajusta a lo que ella pretende es el formato temas. Opta por él y sigue con el campo **Disponibilidad**, no le interesa que cualquier alumno pueda acceder a su curso, así que selecciona



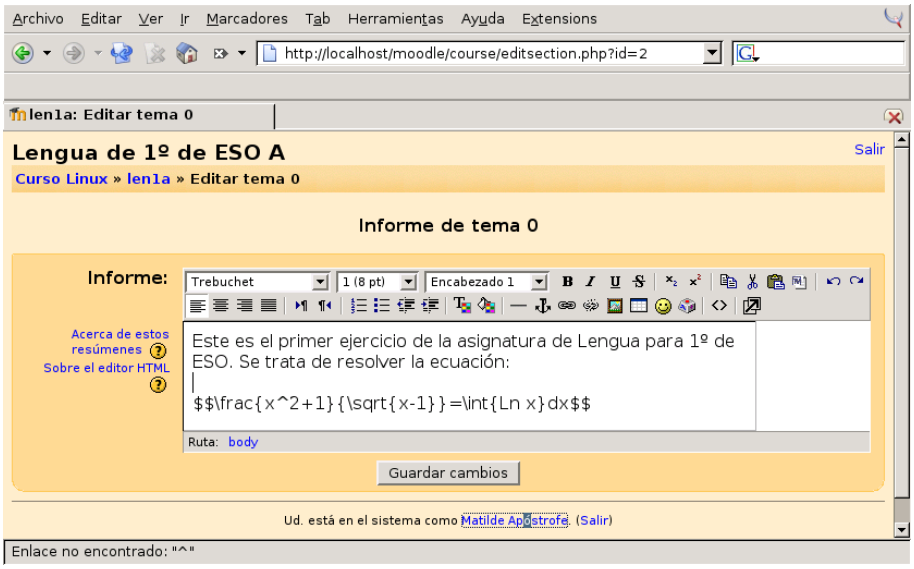
y escribe la contraseña de acceso.

De esa forma, para que un alumno pueda después matricularse en el curso, además de darse de alta en Moodle, tendrá que conocer la palabra de paso. Es un sistema que permite que los alumnos se automatriculen en los cursos pero manteniendo nosotros el control sobre quién lo hace.

Además, opta por no admitir invitados y guarda los cambios. Entra entonces por primera vez en la ventana de su curso[16] recién creado (¿os suena?) y activa la edición (Activar edición).



Como además, es aficionada a las matemáticas les va a gastar una broma a sus alumnos, y tras pulsar sobre el icono que hay encima de **Foro de Noticias**, comienza a escribir su texto de bienvenida



¿Qué dirán sus alumnos y alumnas cuando vean el resultado?

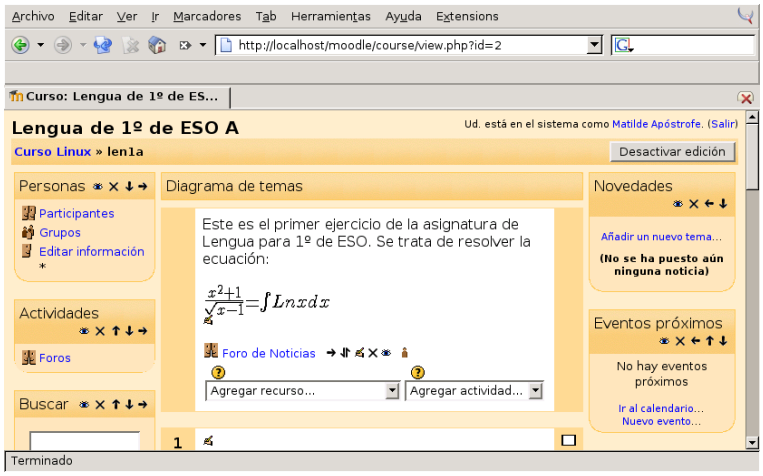
[16] Si aparecen errores del tipo

```
"Fatal error: Allowed memory size of 8388608 bytes exhausted (tried to
allocate 46080 bytes) in ...
Fatal error: Allowed memory size of 8388608 bytes exhausted (tried to
allocate 136 bytes) in Unknown on line 0"
```

lo podemos solucionar modificando la directiva del fichero de configuración de PHP

```
memory_limit = 8M
```

Un valor de 16 debe ser suficiente para Moodle.



Para seguir trabajando y conocer qué puede seguir haciendo sólo tiene que echar mano de la **Ayuda** accesible desde esta misma página[17] y sobre todo un documento ya comentado en la página 470: *El manual del profesor*.

21.1.5. Más configuración

En toda esta sección partiremos de la base de que somos el administrador de la plataforma.

Tamaño permitido de subida ficheros

Por defecto, la configuración de php no permite que subamos ficheros de más de 2MB. Si ese tamaño se nos queda pequeño tendremos que modificar nuestro sistema para que sí que lo permita. Supongamos que deseamos establecer un tamaño máximo de subida de ficheros de 16MB, si es así, los parámetros de configuración a cambiar son:

- Del fichero php.ini:

`post_max_size = 16M` con este valor establecemos el tamaño máximo de archivos enviados usando el método POST. Su valor condiciona el valor de la directiva siguiente así que debe ser siempre más grande que el asignado a `upload_max_filesize`.

`upload_max_filesize = 16M` del fichero de configración de php (`php.ini`) ajustándola al valor deseado.

- Del fichero de configuración de Apache:

`LimitRequestBody 16000000` **No es obligatorio añadirla** ya que por defecto está a cero, es decir, sin límite. Sólo hemos de añadirla si deseamos controlar cuál es el tamaño máximo de fichero a servir por una solicitud HTTP. Este parámetro se puede usar en la configuración del servidor, de los hosts virtuales, a nivel de directorio y en los archivos `.htaccess`. Notar que su valor se da en Bytes[18] y hemos redondeado a la baja. En el caso de que optemos por añadirla, un buen lugar para hacerlo es el fichero:

Debian `/etc/apache2/mods-available/php.conf`

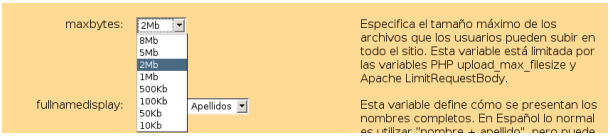
Fedora `/etc/httpd/conf.d/php.conf`

Cuando terminemos de hacer los cambios, reiniciemos el servidor Apache, por ejemplo con:

[17] Un documento que os puede resultar interesante es `http://www.iesmurgi.org/~ljoya/moodle/` o en la versión en formato pdf disponible en `http://www.iesmurgi.org/modules.php?name=Downloads&d_op=viewdownload&cid=3`.
[18] El valor máximo es de 2147483647 bytes, es decir, 2GB.

```
#apache2ctl restart
```

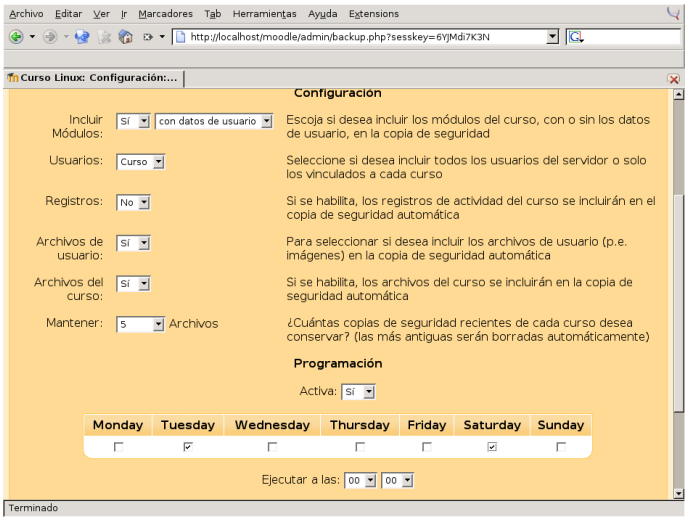
Además, tendremos que permitirlo desde Moodle: **Administración→Configuración→ Variables**



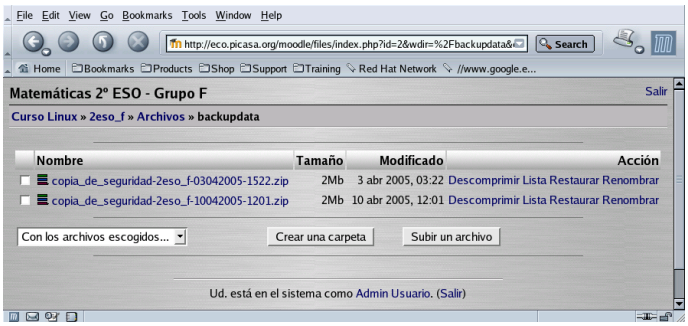
y después en la configuración de los cursos y módulos o actividades que podemos definir en ellos.

Copias de seguridad

Un aspecto fundamental que no debemos descuidar como administradores de cualquier sistema es el mantener una buena política de copias de seguridad. En este sentido Moodle es una maravilla ya que no tenemos que hacer "casi nada" para conseguirlo. Pulsamos sobre **Configuración→Copias de Seguridad**



y porgramamos la política de copias. Es buena idea (si nuestro disco lo permite) mantener varios ficheros y programar las copias para que se realicen en un par de días y a una hora en que sepamos que el servidor está con poca carga. En cada curso se creará una carpeta accesible desde **Administración→Archivos→Backupdata**



Añadir cursos

Desde cualquier curso podemos crear copias de seguridad que tienen una funcionalidad añadida: podemos restaurarlas en cualquier otro moodle. Para crear una copia de seguridad para un curso en concreto pulsaremos sobre **Administración→Copia de seguridad**

Su uso no presenta ninguna dificultad.

➥ **Restaurar cursos de un Moodle a otro.** En esta práctica vamos a guiar la forma de restaurar un par de cursos en castellano que tenemos disponibles en la Web de Moodle `http://moodle.org/course/view.php?id=11`. Se trata de los cursos:

Recurso Moodle Para Profesores `http://moodle.org/mod/resource/view.php?id=2210`

Moodle Para Alumnos `http://moodle.org/mod/resource/view.php?id=2209`

Características de Moodle `http://moodle.org/mod/resource/view.php?id=2574`

Guiaremos el proceso sólo para el primero dejando los otros dos como ejerecicio.

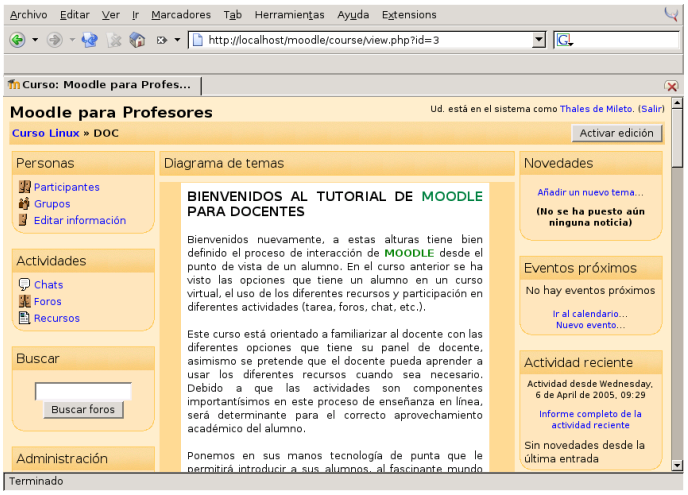
Bajaremos el fichero a nuestro equipo[19]. Después, desde la página principal de Moodle pulsaremos sobre **Restaurar**

para subir el fichero a la plataforma. Una vez en ella, sólo hemos de marcarlo y pulsar sobre **restaurar**.

[19]Si intentamos subir a Moodle ficheros de un tamaño mayor del que permite la configuración del sistema (por defecto 2MB) no nos lo permitirá. Aunque aumnetemos ese tamaño, la única solución para ficheros muy grandes es ponerlos en el sitio adecuado a mano. Por ejemplo con:

```
#cp fichero.zip /var/www/moodledata/1/
```

El proceso no presenta mayor dificultad y sólo hay que fijarse bien en las distintas opciones que se nos ofertan. Cuando acabemos tendremos creado un nuevo curso (si hemos elegido esa opción) similar a



■

Añadir módulos

Si bien las posibilidades de la instalación por defecto de Moodle son muy amplias, existen una serie de añadidos que nos puede interesar tener en nuestro sitio. Podemos ver cuáles hay disponibles en `http://moodle.org/download/modules/`. De los que hay disponibles, la mayoría están instalados pero hay otros que son interesantes y no lo están, por lo que tendremos que instalarlos nosotros. El proceso para todos ellos es parecido así que vamos a instalar sólo uno, se trata de:

`Questionnaire` que nos va a permitir poder realizar encuestas desde Moodle.

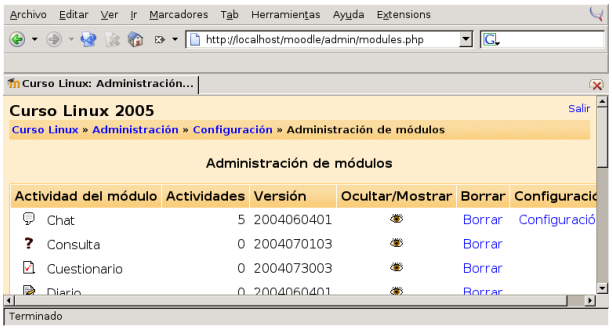
Para instalarlo, lo bajamos a nuestro ordenador y descomprimimos el fichero en el lugar adecuado

```
# cp questionnaire.zip /var/www/html/moodle/mod/
# cd /var/www/html/moodle/mod/
# unzip questionnaire.zip
```

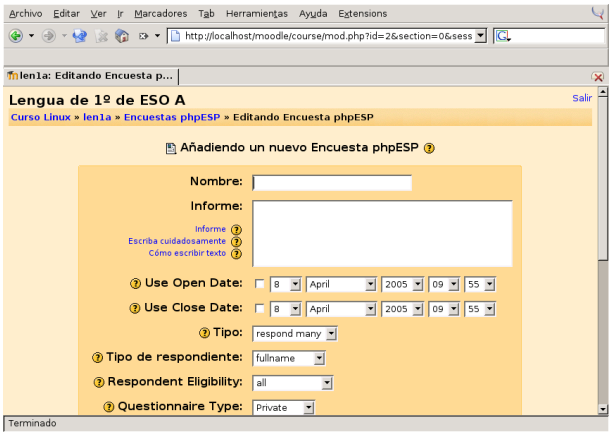
Ya está, así de fácil[20]. Cuando como administrador entremos en la configuración de los módulos[21] (en **Administración**) él sólo se encargará de todo: actualizar la base de datos, crear las tablas, etc.

[20] Puede que tengamos que copiar los archivos de `lang/help/questionnaire/*` en el subdirectorio `lang` que cuelga del raíz de Moodle.

[21] Desde aquí, a su vez, podremos borrar los módulos que no nos interesen.



Añadir en un curso un cuestionario es fácil. Con la edición activa marcaremos sobre **Agregar Actividad...→Encuesta phpESP**



Actualizaciones de Moodle

La versiones de Moodle se suceden casi a diario y a veces es necesario actualizar el sistema porque hay bugs que se han corregido para la versión con la que trabajamos o porque se han añadido funcionalidades interesantes que nuestra verisión no soporta. Afortunadamente, este tema también está bien resuelto. Para no alargar la entrega tenéis la forma de hacerlo (en castellano) explicada en `http://localhost/moodle/doc/?file=upgrade.html`.

Se puede resumir en:

- Crear una copia de seguridad de la base de datos

  ```
  #mysqldump -u root -p moodle > moodle-backup-10-04-2005.sql
  ```

- Movemos el moodle viejo a otro directorio por si acaso:

  ```
  #mv moodle moodle.backup
  ```

- Descomprimimos la última versión en el lugar adecuado

  ```
  #tar xvzf moodle-ultimo.tgz
  ```

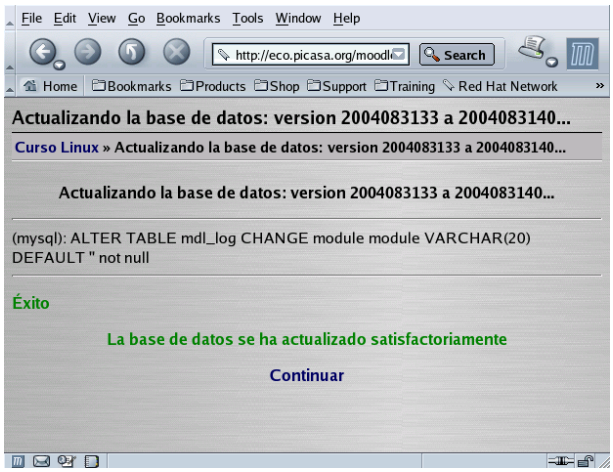
- Copiamos el fichero de configuración de la instalación "vieja" a donde debe de estar

  ```
  #cp moodle.backup/config.php moodle
  ```

  Si hemos hecho cambios en el Moodle que teníamos instalado tendremos que pasarlos al Moodle nuevo: por ejemplo módulos añadidos, temas instalados, etc.

- Por útlimo sólo hemos de ejecutar (él sólo se encarga del resto):

  `http://www.midominio.org/moodle/admin`



21.2. PHP-Nuke

¿Qué es PHP-Nuke?

PHP-Nuke es un sistema automatizado de noticias especialmente diseñado para ser usado en Intranets e Internet. El Administrador tiene el control total de su sitio Web, sus usuarios registrados, y tendrá a la mano un conjunto de herramientas poderosas para mantener una página web activa y 100 % interactiva usando bases de datos.

Su autor es Francisco Burzi, que es el que mantiene el código y realiza todas las modificaciones que lleva el paquete original. Podemos encontrar su trabajo en `http://www.phpnuke.org`. Los requisitos para usar PHP-Nuke, que veremos posteriormente, no forman parte del sistema y han de instalarse independientemente. (*Manual de referencia rápida para PHP-Nuke*, CARLOS PÉREZ PÉREZ `http://www.forodecanarias.org/doc/nuke/html/node4.html`

21.2.1. Introducción

Los portales web son sitios web pensados para manejar una gran cantidad de información y permitir el mantenimiento de páginas web actualizadas y dinámicas usando bases de datos.

En `http://www.aferve.com/index.php?module=bkbCompare&func=narrow_selection&id=1` tenéis un estudio comparativo de los distintos portales y sus características.

Hemos optado por PHP-Nuke porque es un clásico, su uso está bien documentado (lo que os permitirá seguir profundizando en su conocimiento) y es fácil de instalar. Además, una vez que se aprenda a manejar uno, el uso de cualquier otro portal es similar y no presentará ningún problema.

21.2.2. Instalación de PHP-Nuke

Con objeto de homogeneizar el proceso de instalación en ambas distribuciones partimos de la configuración del raíz del servidor Web de Apache, en Debian es similar a la de Fedora (véase 21.1.2 en la página 462) .

Vamos a ver la potencia de instalar MySQL (servidor de base de datos) y, junto con PHP y el software PHPNuke (`http://phpnuke.org/`), montar una web altamente configurable.

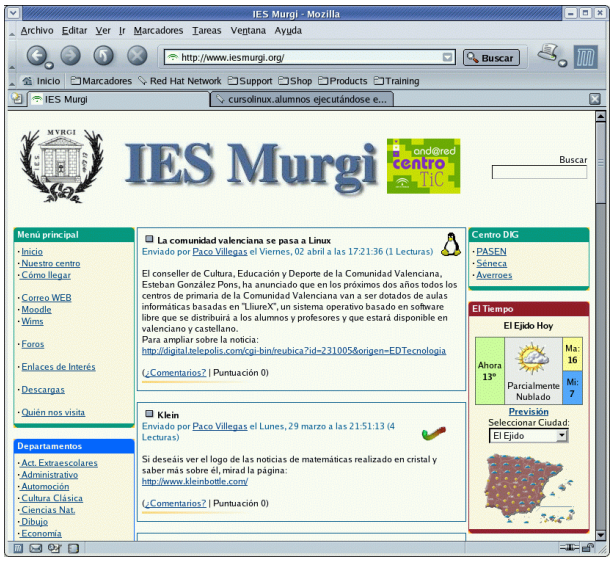


Figura 21.3: Web del IES Murgi

Para eso necesitamos:

1. Tener instalado Apache
2. Tener instalado PHP
3. Tener instalada la base de datos MySQL
4. Instalar el módulo que permite a PHP disponer de soporte de base de datos MySQL
5. Descomprimir el PHPNuke[22] en `/var/www` (esto es opcional, pero así se queda ya puesto en su sitio):

```
# cp PHP-Nuke-7.5.zip /var/www
# cd /var/www
# unzip PHP-Nuke-7.5.zip
```

6. Una vez descomprimido y tras situarnos en el directorio `/var/www/sql` ejecutemos[23]:

```
# mysqladmin -u root -p create nuke
```

para crear la base de datos `nuke`, y

```
# mysql -u root -p nuke < nuke.sql
```

para crear las tablas de esta base de datos según se establece en el fichero `nuke.sql`.

7. Ajustar la contraseña[24] de la base de datos en el fichero `/var/www/html/config.php`

```
$dbuname = "root";
$dbpass = "contraseña";
```

Podemos tener varios Nukes instalados en nuestra máquina. Para eso sólo debemos descomprimirlos en directorios diferentes y crear bases de datos diferentes para cada uno de ellos, por ejemplo para el segundo nuke

```
# mysqladmin -u root -p create nuke2
# mysql -u root -p nuke2 < nuke.sql
```

y ajustar la variable adecuada a la base de datos para ese phpNuke, en concreto en el fichero `config.php`

```
$dbname = "nuke2";
```

8. Comprobar que todo está bien, apuntando con nuestro navegador a[25] `http://localhost/`

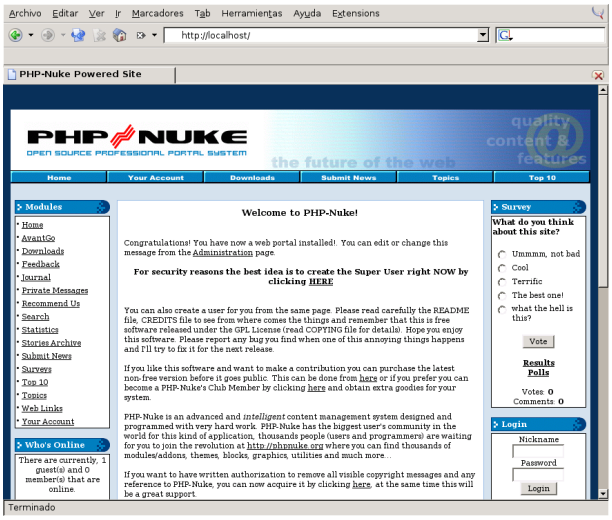
[22] Lo podemos bajar de `http://www.phpnuke.org/modules.php?op=modload&name=Downloads&d_op=viewdownload&cid=1`.

[23] Véase el fichero `/var/www/html/INSTALL` para ampliar sobre los detalles de la instalación.
Podemos usar phpMyAdmin para hacerlo, pero en general es más rápido desde el modo comando.

[24] Lo deseable es que sea otro usuario el que pueda conecar con la base de datos. para saber cómo se hace con phpMyAdmin debéis consultar en la página 464 cuando se explica para Moodle.

[25]

- En Debian es mejor ajustar el `DocumentRoot` modificando el fichero `/etc/apache2/sites-available/default` y ajustarlo a
  `DocumentRoot /var/www/html`
  Una vez cambiado:
  `# apache2ctl restart`
- Se consigue administrar el portal apuntando a: `http://localhost/admin.php`



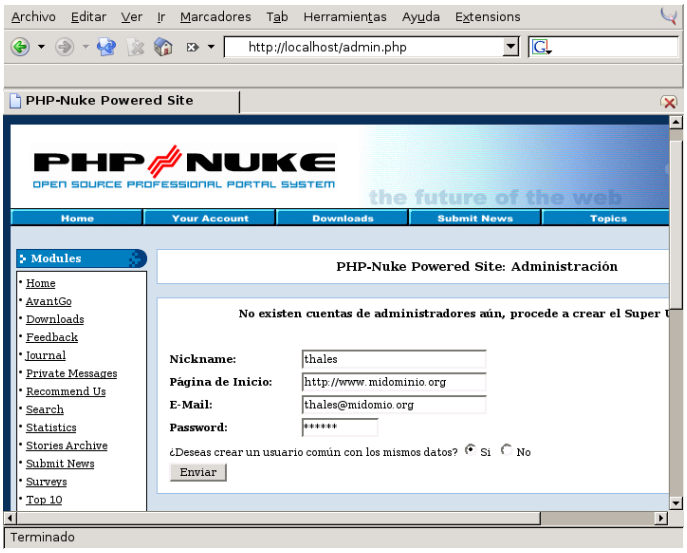
¡Bienvenidos a PHP-Nuke!

¡Enhorabuena!¡Ya tiene instalado un portal web! Puede editar o cambiar este mensaje desde la página "Administration".

Por razones de seguridad lo mejor es crear AHORA MISMO el Super-usuario pinchando AQUÍ.

En primer lugar pongamos el portal en castellano para eso optemos porque el interfaz se muestre así, seleccionando el idioma español del bloque de la derecha

Una vez pasado este trámite, vamos crear el super-usuario (para nosotros el socorrido THALES) pinchando allí[26]:



[26]Después, una vez autentificados como super-usuario, accederemos a la web de administración.

Al crear la cuenta *GOD* (se puede poner el nombre que se quiera) no podemos usar ni espacios ni caracteres "extraños". Si los usamos no podremos acceder al portal. En este caso, una solución rápida (ya que estamos empezando) puede ser:

```
# mysqladmin -u root -p drop nuke
Enter password:
Dropping the database is potentially a very bad thing to do.
Any data stored in the database will be destroyed.

Do you really want to drop the 'nuke' database [y/N]
```

y, tras confirmar, volver a realizar de nuevo el paso 6.

9. Enviemos el formulario y autentifiquémonos en el portal como administrador[27]. Accederemos a la web de administración:

Figura 21.4: phpNuke admin

Nos queda configurar el portal, ponerlo por defecto en castellano, crear usuarios del sistema, etc. Pero eso lo dejamos para el punto siguiente, salgamos de él pulsando sobre **Salir**  .

- Antes de meternos en la configuración, en `/var/www/html/docs` se nos ha instalado la documentación que acompaña al portal. Más de una duda seguro que se puede resolver desde aquí.

[27] Es conveniente guardar a buen recaudo los datos del administrador y la contraseña de acceso.

- Si lo que desamos es actualizar nuestro Nuke a una versión posterior podéis consular en: `http://www.phpnuke-hispano.com/modules.php?name=News&file=article&sid=77`

21.2.3. Configuración básica del portal.

Existen varios documentos bastante buenos para ampliar sobre este tema, por ejemplo:

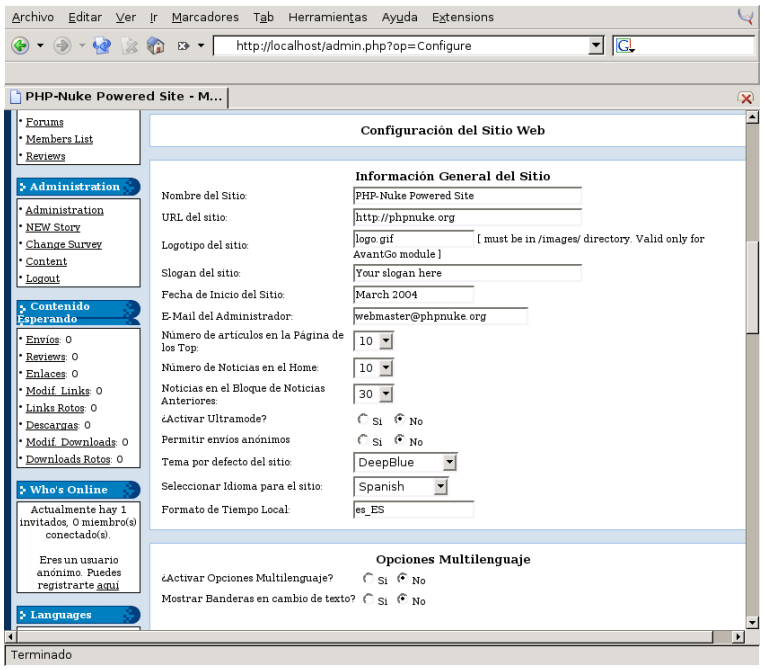
- `http://www.conocimientosweb.net/portal/modules/Manual/index.htm`
- `http://www.forodecanarias.org/doc/nuke/html/`

y en italiano:

- `http://www.claudioerba.com/`

Veamos algunos aspectos básicos de configuración. Desde el navegador web escribamos:
`http://localhost/admin.php`
y tras autenticarnos accederemos de nuevo a la web de administración (figura 21.4 en la página anterior).

Después, si pulsamos sobre **Preferencias**  nos llevará a una página desde la que podremos configurar los aspectos generales del portal[28]. Si nos desplazamos hacia abajo en la ventana del navegador, podremos modificar el valor de la lista desplagable **Seleccionar el Idioma para el sitio** *(Spanish)*, además cambiaremos el **Formato de Tiempo Local** (*Locale Time Format*) a `es_ES`



Está claro que antes de salir deberíamos adecuar algunos de los campos de esta sección y adaptarlos a nuestro sitio.

De esta ventana comentaremos únicamente que podemos optar por distintos temas por defecto para nuestro phpNuke. Sólo hay que pulsar sobre **Tema por defecto del sitio**, ir seleccionando los que nos van apareciendo en la lista desplegable y guardar los cambios.

[28] Una descripción más completa de las distintas opciones se puede consultar en: `http://www.forodecanarias.org/doc/nuke/html/node36.html`

Analicemos algunas de las posibilidades de la página de administración. Seguiremos el orden "lógico" inicial de configuración y no el que aparece en la Web.

Mensajes (Administración)

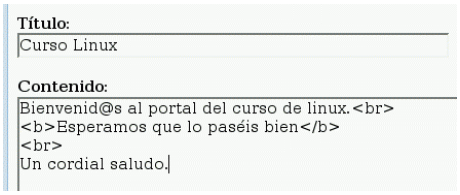
Desde este apartado podemos modificar el mensaje de bienvenida. Si accedemos a él podemos (tendremos que desplazarnos hacia abajo en la página Web)

1. Desactivar el mensaje de bienvenida por defecto. Para eso pulsamos sobre **Editar** el mensaje inicial

y después sobre **Desactivar mensaje** 

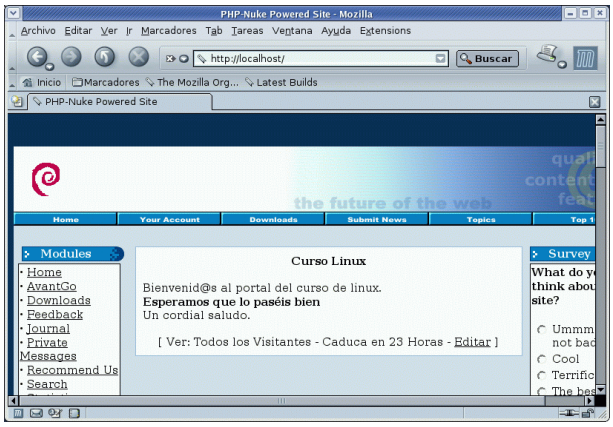
y [**Guardar Cambios**]

2. Añadir nuestro primer mensaje, para eso escribiremos




Si queremos remarcar el texto hay que usar las etiquetas del html.

El resto de opciones no presentan problema. Optamos por dejar las opciones por defecto y pulsamos sobre [**Agregar Mensaje**].




Topics (Módulos)

Es desde donde definiremos los temas en los que se clasificarán nuestras noticias. Por defecto sólo viene un tema predeterminado y, como es obvio, aquí tendremos que dedicar un tiempo antes de que nuestra web esté operativa. Vamos a añadir un tema sobre las cuestiones relacionadas con las matemáticas.

1. Lo primero es buscar un gráfico que se adecúe a nuestro propósito, por ejemplo, el fichero[29]

[29] Tendrán que ser gráficos apropiados: png, gif o jpg y de reducido tamaño.

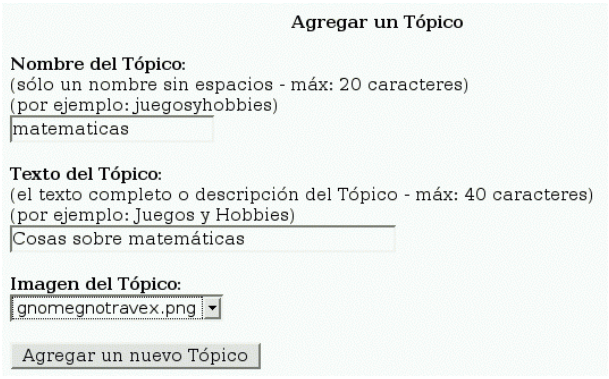
`/usr/share/pixmaps/gnome-gnotravex.png` ().

2. Situémoslo en su sitio:

```
# cp /usr/share/pixmaps/gnome-gnotravex.png
                    /var/www/html/images/topics/gnomegnotravex.png
```

 Notar que hemos sido cuidadosos y en el nombre de fichero no hemos permitido caracteres extraños (el guión), si no lo hacemos así, phpNuke no podrá trabajar con ese gráfico.

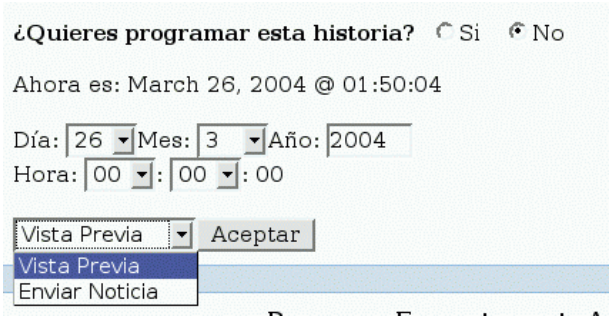
3. Añadamos el tema.



 **News (Módulos)**

Una vez definidos los temas, desde aquí, podemos añadir las noticias que se mostrarán en nuestro portal. Su uso no es complicado y sólo hay que tener en cuenta que:

- Hay que seleccionar obligatoriamente un tema principal desde la lista desplegable **Tópico**. Es opcional marcar las casillas de **Tópicos asociados.**
- Para remarcar el texto es necesario usar las etiquetas de html.
- El **Texto de la Noticia** es lo que se verá en la página principal. Al **Texto extendido** hay que acceder.



- Podemos programar la fecha en que se mostrará la noticia en la página principal (por ejemplo el 31/12/2004 a las 0 horas para felicitar del año nuevo) y añadir una encuesta a cada noticia que pongamos.
- Si pulsamos sobre **Aceptar** estando activa la opción de `Vista Previa` se nos mostrará cómo quedará la noticia, pero no se envía. Para enviarla hay que optar por `Enviar Noticia` y **Aceptar**.

- Desde la ventana principal de administración siempre podremos editar o borrar las noticias que nosotros hayamos enviado.

Analicemos ya con cierto orden el resto de opciones

Administración

Respaldo

Es para crear una copia de seguridad de la base de datos de nuestro PHP-Nuke.

Banner

Por si deseamos añadir carteles publicitarios a nuestro sitio y gestionar la "cartera" de clientes.

Bloques

Es uno de los elementos fundamentales de nuestro portal.

Administración de Bloques

Título	Posición	Peso	Tipo	Estado	Visible para	Funciones
Modules	Izquierda	1 ⇩	ARCHIVO	Activo	Todo el mundo	[Editar | Desactivar | Borrar | Ver]
Administration	Izquierda	2 ⇧⇩	SISTEMA	Activo	Sólo Administradores	[Editar | Desactivar | Borrar | Ver]
Who's Online	Izquierda	3 ⇧⇩	ARCHIVO	Activo	Todo el mundo	[Editar | Desactivar | Borrar | Ver]
Search	Izquierda	4 ⇧⇩	ARCHIVO	*Inactivo*	Todo el mundo	[Editar | Activar | Borrar | Ver]
Languages	Izquierda	5 ⇧⇩	ARCHIVO	Activo	Todo el mundo	[Editar | Desactivar | Borrar | Ver]
Random Headlines	Izquierda	6 ⇧⇩	ARCHIVO	*Inactivo*	Todo el mundo	[Editar | Activar | Borrar | Ver]

Desde aquí podemos administrar los bloques de nuestro portal. Las dos columnas "importantes" son:

Peso Nos permite subir o bajar un bloque con sólo pulsar sobre las flechas ⇑⇓

Funciones al pulsar sobre el enlace se ejecuta la acción asociada.

Editar para modificar el bloque

Activar/Desactivar se trata de un "interruptor" que nos permite que un bloque se muestre o no en la página principal.

Borrar pues eso.

Ver para acceder a una vista previa de él.

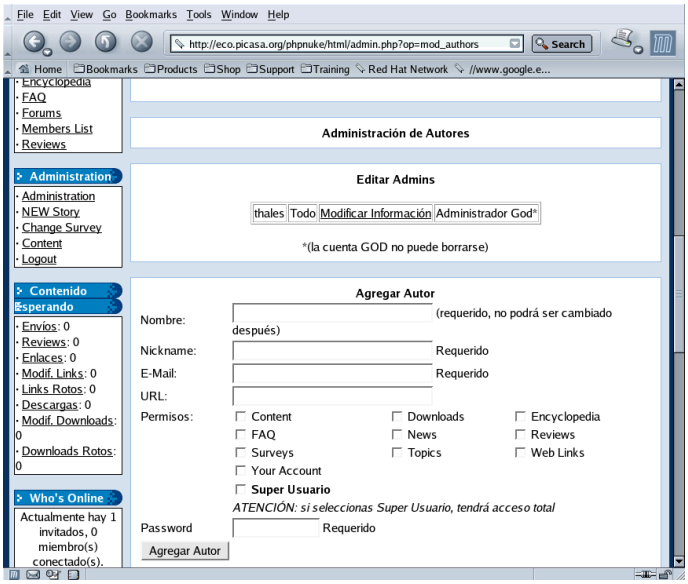
➥ **Añadamos un bloque:** en esta práctica vamos a añadir un bloque en el que se mostrarán las noticias de la web de *Barrapunto*. Para poder hacerlo necesitamos que el sitio de donde las vamos a "coger" tenga un fichero dentro de su servidor, del tipo RSS/RDF. La URL es:

```
http://backends.barrapunto.com/barrapunto.rss
```

Si guardamos los cambios (**Crear Bloque**) y nos situamos sobre la página principal tendremos un bloque donde aparecerán las noticias de la página que hemos elegido antes, su contenido será similar al que se muestra en la figura lateral. ■

Editar Admins

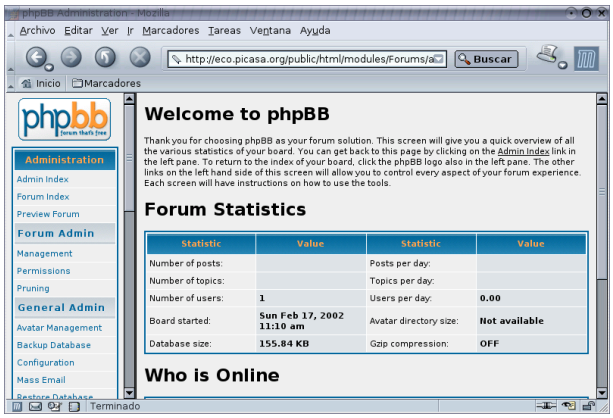
Si trabajamos en un centro de enseñanza y hemos sido o somos los WebMaster sabemos que es más que deseable poder repartir el trabajo. Desde esta página podemos dar de alta a usuarios que van a poder administrar con nosotros el portal. Está en nuestras manos definir qué nivel de "trabajo" le vamos a permitir al autor:



Para que el nuevo "administrador" ejecute sus funciones en nuestra máquina `www.midomio.org` tendrá que usar la URL: `http://www.midomio.org/admin.php`

Forums

phpNuke integra como módulo uno de los mejores paquetes para gestionar foros de la actualidad. Se trata de sistema de foros phpBB



La web del programa es, `http://www.phpbb.com/`, desde la sección *downloads* podemos acceder a los módulos de lenguaje. Se trata de bajarnos los que nos van a permitir poner nuestros foros en

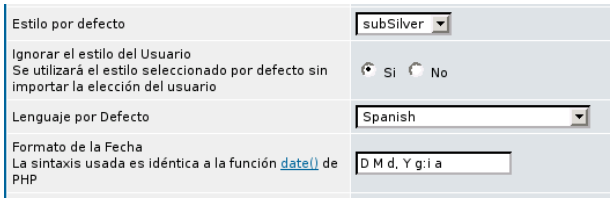
castellano, es decir, los paquetes[30]:

```
lang_spanish.tar.gz
subSilver_spanish.tar.gz
```

Descomprimimos el fichero `lang_spanish.tar.gz` en `/var/www/html/modules/Forums/language`

```
#cp lang_spanish.tar.gz /var/www/html/modules/Forums/language
#cd /var/www/html/modules/Forums/language
#tar -xzvf lang_spanish.tar.gz
```

y, tras entrar en el menú de administración del phpNuke, iniciamos la sección de foros. En el menú de la izquierda nos situamos sobre **General Admin Configuration** y en **Default Language** seleccionamos el idioma `español` (antes de descomprimir el fichero no aparece esa opción) y enviamos el formulario[31].



De esta forma conseguimos que el entorno esté en castellano.

Pero no está castellanizado "del todo", el tema por defecto se denomina *subSilver* y, además, deseamos que los gráficos estén en consonancia con el idioma que acabamos de instalar. Nada más simple, pongamos el segundo fichero donde debe estar y desempaquetémoslo:

```
# cp subSilver_spanish.tar.gz /var/www/html/modules/Forums/templates/
# cd /var/www/html/modules/Forums/templates/
# tar -xzvf subSilver_spanish.tar.gz
```

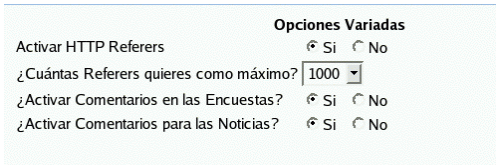
Está claro que ahora es más fácil modificar y adecuar la configuración de los foros: se deja como ejercicio :-).

 **Users Groups**

Permite crear grupos y asignar usuarios a esos grupos, además se puede conseguir que los grupos puedan ver sólo ciertos módulos y bloques.

 **HTTP Referers**

¿Quién enlaza nuestro sitio? El número (1000 por defecto) de referencias se controla desde la página de **Preferencias**



[30] Podemos optar por la versión en formato `.zip` o la `tar.gz`

[31] Si no nos funciona bien, probemos creando el fichero:

```
#touch lang-spanish.php
```

y se debe eliminar el problema.

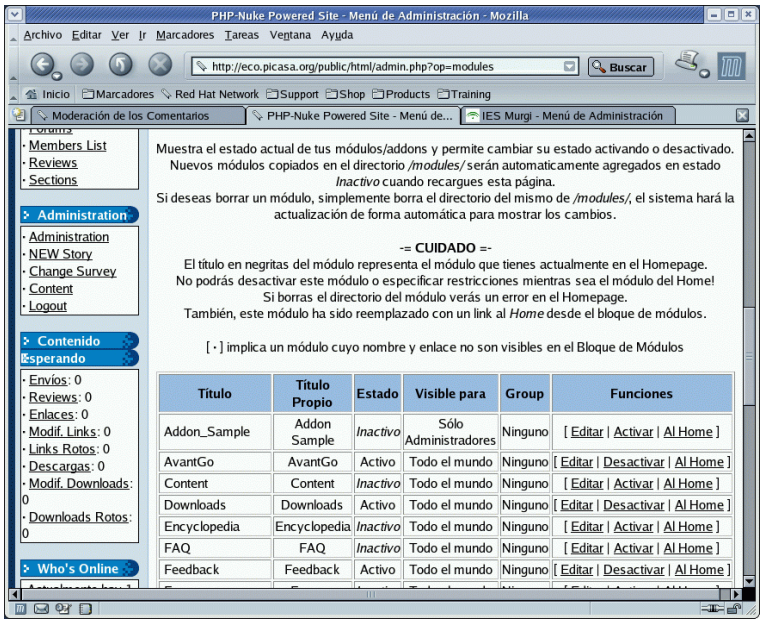
 **IP Ban**

Para bloquear las IP de los "chicos malos"

 **Módulos**

Se trata de uno de los elementos más importantes de la configuración. Los módulos son los que dotan (junto con los bloques) de contenido a nuestra Web.

En esta sección accederemos a una tabla en la que se nos informa de los distintos módulos, así como de sus características.



Título nombre del directorio que contiene al módulo.

Título propio nombre del módulo, este valor lo podemos castellanizar.

Estado si está activo o no

Visible para ¿qué usuarios pueden acceder a él?, ¿todos, sólo los registrados o sólo los administradores?

Funciones para editar, activar/desactivarlo o ponerlo como módulo por defecto en la página principal (*Al Home*)

No sólo podemos usar los que vienen por defecto. Añadir un módulo a phpNuke (en general) es bastante sencillo: sólo hay que bajar el módulo y descomprimirlo en el lugar adecuado[32] (`/var/www/html/modules`). Después tenemos que recargar la página desde la que controlamos los módulos, y se añadirá a la tabla de características de los módulos. Como por defecto estará desactivado, tendremos que activarlo y castellanizarlo. Veamos un ejemplo de cómo añadir un módulo.

➥**Coppermine:** este módulo[33] permite para crear álbumes de fotos[34] dentro de portal, y además está traducido al español. Podemos bajarlo de `http://www.nukebazar.com/`, la única "pega" es que antes hay que registrarse.[35]

[32] Puede que además tengamos que "retocar" la base de datos `nuke`.

[33] La Web del programa es: `http://coppermine.sourceforge.net`

[34] Existen en la Web múltiples utilidades de este tipo. Además de la comentada merece la pena probar:

gallery `http://gallery.menalto.com/`

My Photo Gallery `http://www.fuzzymonkey.org/cgi-bin/newfuzzy/software.cgi`

[35] Está a vuestra disposición en la Web del curso

Una vez en nuestra máquina, lo descomprimimos

```
$ mkdir coppermine
$ cp coppermine1,1d6.5.zip coppermine
$ cd coppermine
$ unzip coppermine1,1d6.5.zip
```

Al descomprimir se crean varios directorios[36]: `block` y `modules`. Lo primero será actualizar la base de datos nuke con:

```
#mysql -u root -p nuke <coppermine.sql
```

Por útlimo copiemos el contenido de los directorios anteriores en los lugares adecuados:

`block` en `/var/www/html/block`

`modules` en `/var/www/html/modules`

Ya está. Reiniciemos el bloque de módulos y activémoslo. Mejor si además le ponemos un nombre en castellano.

coppermine	Fotos	Activo	Todo el mundo	Ninguno	[Editar | Desactivar | Al Home]

Para administrarlo, pulsaremos sobre **Fotos** del bloque **Módulos (Modules)** de la página principal del portal

Al pusar sobre **Config** podremos optar por cambiar el idioma y configurarlo a nuestro gusto.

Para no perder de vista lo que es importante (por ahora) aparquemos la configuración de este módulo hasta la 21.2.4 en la página 492. En ella se explica cómo crear una galería y la forma de introducir nuestras fotos. Así que continuemos con la configuración del portal. ■

Boletín

Para enviar un correo (boletín) a los usuarios que están suscritos o a todos los usuarios del portal.

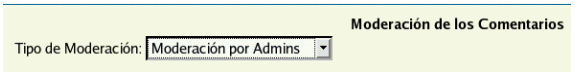
Optimize DB

Para optimizar la base de datos de nuestro portal.

Envios

Permite configurar desde **Preferencias** si deseamos que usuarios anónimos puedan poner artículos en nuestro portal o si, antes de que se publique, el administrador o usuarios han de darle el visto bueno.

[36] Además del fichero que explica cómo se instala: `Installation-english.txt`



Si establecemos algún tipo de moderación, aquí se almacenarán las noticias pendientes de publicar. Depende de nosotros decidir qué noticias publicamos y cuáles no.

Módulos

Contenido

Nos permite crear páginas de contenido diverso ("cajón de sastre") clasificadas por temas. Si usamos la etiqueta `<!-pagebreak->` podremos crear documentos (usando las etiquetas del HTML) de varias páginas.

Descargas (Downloads)

En primer lugar tendremos que crear las categorías de descargas, después añadiremos los programas y/o materiales que deseamos se puedan bajar desde nuestro portal. Una vez creada la categoría principal se nos "complicará" la página permitiéndonos añadir subcategorías y descargas clasificadas dentro de esas categorías. No presenta mayor dificultad.

Enciclopedia

Permite generar una enciclopedia basada en categorías. Se pone el nombre a la enciclopedia y después se agregan los términos. El trabajo es ir alimentando los términos.

FAQ

O su traducción PUF, es decir, que como estamos de trabajo fatal, seguro que diremos ¡puuf! cuando nos digan que hagamos la FAQ (*Frequently Asked Question*s). Es broma: se trata de que podemos crear una serie de preguntas de uso frecuente (PUF) con sus respuestas, que facilite a los usuarios el uso del portal.

Reviews

Abre una página dedicada al análisis de productos y servicios. Los usuarios pueden introducir comentarios amplios sobre un determinado producto, su nombre, e-mail, enlaces. Permite un análisis más completo e interactivo que las encuestas.

Secciones

Para crear secciones especiales, es el lugar adecuado para situar artículos que no entran dentro de la página principal.

Encuestas (Survey/Polls)

Administración de encuestas, su uso es sencillo e intuitivo.

Enlaces (Web Links)

Para gestionar los enlaces a páginas interesantes desde nuestro portal. Primero se crean las categorías (Ejemplo: enlaces de matemáticas) y después, dentro de cada categoría, se añaden los enlaces.

Usuarios

En PHP-Nuke podemos encontrar tres tipos de usuarios: superusuarios (administradores), todo el mundo (usuarios que visitan la página pero no se registran) y usuarios registrados (al registrarse se les envía un login y una contraseña que les permite interactuar con la página, por ej. enviando noticias, participando en foros, etc). Desde aquí podemos gestionar estos últimos.

Efemérides

Desde aquí tendremos la posibilidad de programar las efemérides que deseamos se muestren en la web principal para los días establecidos.

➥ **Cambiemos de Logo** (véase la figura 21.3 en la página 479)

Para poder poner un logo personalizado a nuestro portal lo hemos de hacer de forma manual. Existen dos formas de hacerlo:

1. Tendremos que garantizarnos que nuestro logo es un fichero en formato `gif` y de nombre `logo.gif`. Los temas de phpNuke se guardan en el directorio `html/themes`
   ```
   # ls /var/www/html/themes/
   3D-Fantasy ExtraLite Karate Odyssey SlashOcean
   Anagram index.html Milo Sand_Journey Sunset
   DeepBlue Kaput NukeNews Slash Traditional
   ```
   En general, en cada uno de ellos existe un directorio de nombre `images`. En él se encuentra el fichero `logo.gif`. Sólo hemos de sobreescribirlo.
2. La segunda forma de hacerlo es:
   a) Crear nuestro logo, por ejemplo, `logomicentro.png` y ponerlo en el subdirectorio `images` de ese tema (por ejemplo en `/var/www/html/themes/Kaput/images`).
   b) Entrar en el directorio del tema en el que deseamos poner nuestro logo y modificar el fichero `theme.php`[37], buscar una línea de la forma:
   ```
   <img border=\"0\" src=\"themes/DeepBlue/images/logo.gif\"
   alt=\"Welcome to $sitename!\" hspace=\"20\">
   ```
   y modificarla a nuestro gusto
   ```
   <img border=\"0\" src=\"themes/DeepBlue/images/logomicentro.png\"
   alt=\"Bienvenidos a $sitename!\" hspace=\"20\">
   ```
   ■

21.2.4. Coppermine

Sigamos con este módulo que habíamos dejado a medias y vamos a iniciarnos sobre su uso[38]. Vamos a crear un albúm de forma rápida en el que poner las fotos de la última Olimpiada Matemática Thales, esas fotos las hemos puesto en un directorio de nombre `olimpiada`. De nuevo os recordamos que no pretendemos tratar todas las posibilidades, sólo dar una primera idea de qué se puede hacer.

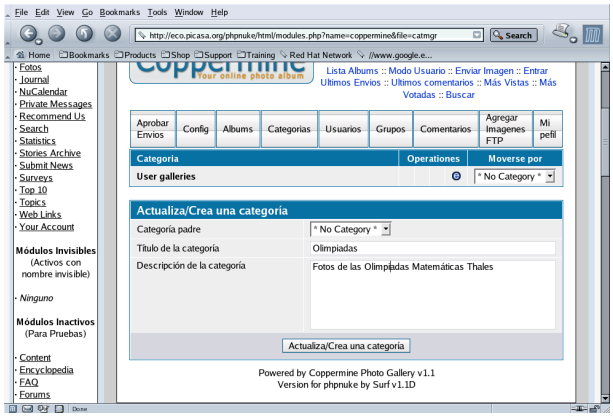
Antes de seguir, varias cuestiones a tener en cuenta si las fotos son de aproximadamente 512KB o más. Si es así, en la pestaña que permite configurar Coppermine (**Config**) accedemos a dos parámetros de configuración para los cuales es necesario aumentar[39] los valores:

- Máximo tamaño de los ficheros añadidos por los usuarios (KB), por defecto está a 1024, doblar su valor no es mala idea.
- Máxima anchura o altura de las imágenes/videos añadidos (pixels). Puede que no sea necesario modificar en exceso este parámetro, en cualquier caso aumentarlo un poco no debe generar ningún problema. Se trata de ajustarlo a las fotos de la máquina con que trabajamos.
- Añadir la ruta del programa `convert`, en general será de la forma `/usr/bin/`

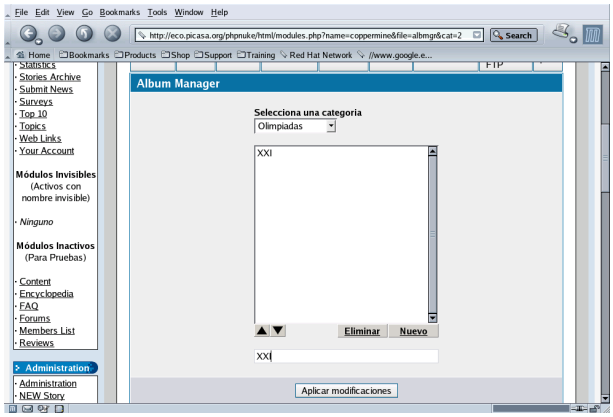
Además, si al intentar añadir fotos a un álbum, el proceso no finaliza, se debe a que tenemos que aumentar la directiva `memory_limit` del archivo de configuración de PHP`(php.ini)` a 32 MB al menos[40]. Si no tiene ese valor no podremos añadir a los álbumes fotos de calidad media.

La primera labor a realizar consiste en crear una categoría en la que pondremos las fotos de las distintas Olimpiadas celebradas. Para eso pulsamos sobre la pestaña **Categorías** y le ponemos el nombre y descripción adecuados.

[37] En los temas *3D-fantasy*, *NukeNews* y *Odyssey* se trata del fichero `header.html`
[38] Un manual en Inglés sobre él: `http://coppermine.sourceforge.net/manual.php`
[39] Si además, deseamos permitir que se puedan subir ficheros de más de 2MB véase 21.1.5 en la página 473
[40] Habrá que reiniciar el servidor web para que los cambios tengan efecto.



Creemos ahora el álbum de fotos para esta convocatoria de Olimpiada, se trata de la XXI edición, así que pulsamos sobre **Albums** y, al pulsar sobre **Nuevo** añadimos el álbum XXI dentro de la categoría Olimpiadas.



Una vez que tenemos creado el álbum, vamos a añadir todas las fotos de una tajada (es posible hacerlo de una en una, tanto como administrador como permitírselo a los usuarios). Pondremos nuestras fotos en el lugar adecuado[41] para no tener que subirlas una a una, por ejemplo con

```
# mv olimpiada/ /var/www/html/modules/coppermine/albums/userpics/
```

Y nos garantizamos que el directorio y las fotos tengan los permisos adecuados, bien haciendo que sean del usuario Apache[42]

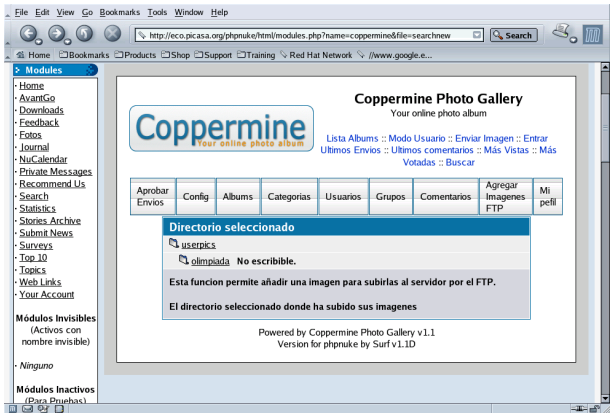
```
# cd /var/www/html/modules/coppermine/albums/userpics/
# chown www-data olimpiada/*
# chown www-data olimpiada
```

o bien relajándole al directorio los permisos al modo 777. Una vez que tenemos nuestras fotos listas pulsamos sobre **Agregar Imágenes**, nos situamos en el directorio `olimpiada` y marcamos las que deseamos añadir al álbum.

[41] Este directorio se puede cambiar desde la pestaña **Config**

[42] En realidad es suficiente con que tenga permisos sobre el directorio. El que tenga permisos sobre las fotos nos sirve sólo en el que caso de que deseemos hacer cambios sobre ellas desde el propio programa.

En Fedora, el usuario por defecto del servidor Web no es `www-data`, es `apache`.



El resultado se puede comprobar en la captura que sigue.

Ya podemos ir añadiendo las de la semana cultural de Instituto, las del viaje de estudios, etc. Sacarle más partido a este magnífico módulo es cosa vuestra.

Prácticas

Tipo I

E4-I-1

Realizar correctamente todas las preguntas del cuestionario que para esta entrega se ha habilitado en Moodle:

1. Señala la afirmación falsa

   a) PHP sólo esta disponible para sistemas LINUX/UNIX

   b) PHP se ejecuta totalmente en el servidor Web

   c) PHP permite conexiones con bases de datos

   d) Al ejecutar un script PHP alojado en un servidor Web, no puedo ver cómo esta hecho al decirle al navegador que me muestre el código fuente de la página.

2. El fichero de configuración de PHP es:

   a) Debian: `/etc/php4/apache2/php.ini` Fedora: `/etc/php.ini`

   b) Debian: `/etc/apache2/mods-available/php.conf` Fedora: `/etc/php.ini`

   c) Debian: `/etc/php4/apache2/php.ini` Fedora: `/etc/httpd/conf.d/php.conf`

   d) Debian: `/etc/apache2/mods-available/php.conf` Fedora: `/etc/httpd/conf.d/php.conf`

3. Para que funcionen bien los scripts php que siguen

```
$cat prueba.html
 <html>
    <body>
       <form action="accion.php" method="POST">
         Tu nombre: <input type=text name=nombre>
         <input type=submit>
       </form>
    </body>
 <html>
$cat accion.php
 Hola <?php echo $nombre?>
```

   hay que modificar en el fichero de configuración de php la directiva

   a) `register_globals` y ponerla en `on`

   b) `register_globals` y ponerla en `off`

   c) funciona bien sin más

d) `variables_order` para conseguir que php registre primero las variables "Post"

4. El conjunto mínimo de operadores que permiten realizar todas las manipulaciones posibles sobre los datos en el modelo relacional, está constituido por :

   a) SELECT, PROJECT, TIMES, UNION y MINUS
   b) JOIN, INTERSECT y DIVIDE
   c) SELECT, PROJECT, TIMES y UNION
   d) SELECT, PROJECT, TIMES, UNION, MINUS y JOIN

5. Una vez conectados con un servidor de bases de datos mysql, y suponiendo que el comando se puede usar sin más, deseamos conocer la estructura de la tabla "datos". Para ello, ejecutaremos:

   a) `DESCRIBE datos;`
   b) `SHOW datos;`
   c) `USE datos;`
   d) `SHOW TABLES datos;`

6. En MySQL, el tipo de datos `FLOAT [(M,D)]` es:

   a) Un entero grande
   b) Un decimal de precisión simple
   c) Un decimal de precisión doble
   d) Un decimal almacenado como cadena

7. Para recuperar el número de campos de un resultado, en una conexión de php con mysql, usaremos la sentencia

   a) `mysql_num_fields`
   b) `mysql_num_rows`
   c) `mysql_fetch_row`
   d) `mysql_fetch_fields`

8. Desde Moodle (con los filtros necesarios activos) para obtener la fórmula

$$\frac{x}{x+1}$$

   escribiremos

   a) `$$\frac{x}{x+1}$$`
   b) `$\frac{x}{x+1}$`
   c) `{x}/{x+1}`
   d) Ninguno de los anteriores, hay que hacerlo usando un gráfico

9. En Moodle, deseamos poder subir ficheros de hasta 32MB de tamaño. Para que nos lo permita ¿qué directivas hemos de modificar en el fichero de configuración de php?:

   a) `post_max_size` y `memory_limit`
   b) `post_max_size` y `upload_max_filesize`

c) `memory_limit` y `upload_max_filesize`

d) Obligatoriamente las tres: `post_max_size`, `memory_limit` y `upload_max_filesize`

10. Deseamos poder disponer de las noticias de Barrapunto intregadas en nuestro phpNuke. Entramos como administrador, y para añadirlas lo hacemos desde:

a) Topics

b) Bloques

c) Módulos

d) Web Links

e) News

E4-I-2 PHP y MySQL.

1. Hay que mandar un script PHP que dé como resultado una tabla en html cuyo contenido sea la tabla del 8. Parte de la salida puede ser similar a:

Tabla	Resultado
8x1	8
8x2	16
...	...

La solución a esta práctica será el script de nombre `e4-i-2a.php`

a) Se trata de que nos digáis qué sentencias SQL hay que usar para[43]:

- Crear una base datos de nombre `alumnos`.
- Crear una tabla de nombre `datos` en la base de datos alumnos con sólo cuatro campos:
  - nombre, apellidos, edad y nif. De longitud variable de 20 y 30 caracteres respectivamente los dos primeros, el tercero entero pequeño sin signo de amplitud 2, y el cuarto de amplitud fija de nueve caracteres.
  - Además, debe estar indexada en los campos nombre y apellidos, no permitir "campos vacíos[44]" y la edad debe tomar como valor por defecto el de 14 años.
- Obtener un listado ordenado por nombre y apellidos de los datos de la tabla.
- Borrar la tabla.
- Borrar la base datos.

La solución a esta práctica se incluirá en el fichero `e4-i-2b.txt`

Ambas cuestiones se integrarán en un fichero de nombre `e4-i-2.tgz`

Tipo II

E4-II-1

Esta práctica consta de dos cuestiones, de las que sólo es necesario realizar una. Es decir, se puede elegir hacer sólo uno de los apartados que siguen:

[43] Se puede usar phpMyAdmin y "copiar" las sentencias a partir del él.
[44] Informalmente hablando

1. Una captura gráfica de phpNuke en la que aparezca (claramente) como logo, un gráfico que contenga vuestro nombre de usuario de los cursos (`edxxxx`).

   a) Una captura gráfica de la pantalla de inicio de Moodle en la que aparezca vuestro nombre de usuario. Por ejemplo, modificando el campo

   ```
   Nombre completo del sitio: Curso Linux 2005 (edxxxx)
   ```

   en la configuración de la página de inicio.

   La solución será el gráfico `e4-ii-1.png`

E4-II-2

En esta práctica se pretende comprobar que se saben integrar php y MySQL. Se pide una página Web que, usando un formulario, nos solicite el nombre, apellidos, edad y nif de un alumno, e insertar los datos en una base de datos (la creada a tal efecto en la práctica *e4-i-1.b* de esta entrega). Se mandará un solo fichero comprimido que contenga el formulario y el código php necesario para que se realice la conexión y la inserción del registro en la base de datos. La solución se subirá en el archivo `e4-ii-2.tgz`

Bibliografía

[1] *Creación de sitios web con PHP4*, F^{o} javier Gil Rubio y otros, Osborne McGraw-Hill

[2] *Desarrollo Web con PHP y MySQL*, Luke Welling & Laura Thomson, Anaya Multimedia

[3] *MySQL*, Paul DuBois, Editorial Prentice-Hall

[4] *MySQL*, Ian Gilfillan, Anaya Multimedia

[5] *Servidor Apache.* Rich Bowen & Ken Coar. Prentice Hall

[6] *Servidor Apache 2.* Mohammed J. Kabir. Anaya Multimedia.

[7] `http://www.mysql-hispano.org/articles.php`

[8] `http://www.conocimientosweb.net/portal/modules/Manual/general5.htm`

[9] `http://html.conclase.net/w3c/html401-es/cover.html`

[10] `http://www.rinconastur.com/php/`

[11] `http://dev.mysql.com/doc/`

[12] Manual de PHP `http://www.php.net/manual/es/`

Parte V

Administración Avanzada

Capítulo 22

Copias de seguridad

> "La idea de hacer una copia de seguridad es hacer copia de tantos datos como sean posibles de tu sistema, pero con excepciones. No es lógico incluir determinados datos ya que perderás tiempo y espacio en el soporte para nada."
>
> *Securing and Optimizing Linux: The Ultimate solution*

22.1. Visión general

Para tener un servidor seguro y confiable es necesario realizar copias de seguridad de forma regular. Los fallos son indeterminados en el tiempo y pueden ocurrir en cualquier momento, pudiendo ser fallos de tipo lógico (borrado accidental de ficheros, errores al procesar shell scripts, ...) o físico (fallo de componentes del ordenador, sobretensiones en el suministro eléctrico, ...). Además, estos fallos pueden producirse de forma fortuita o ser la consecuencia directa de un ataque al servidor por parte de un agente externo. La forma más segura de hacer copias de seguridad es tener los datos almacenados en otro servidor distinto del que se hace la copia o, mejor aún, en un soporte de almacenamiento externo.

A continuación se mostrarán distintos métodos para realizar copias de seguridad usando las utilidades que están por defecto en la mayoría de sistemas linux: `tar`, `dump`, `cpio` y `dd`. Sin embargo, no se entrará en profundidad en su uso, limitándonos a dejar claros los conceptos básicos de las copias de seguridad aplicados a estas herramientas.

También están disponibles herramientas en modo texto como AMANDA, que se utilizarán para hacer más amigable la gestión de copias de seguridad y restauraciones.

La idea de hacer una copia de seguridad es realizar un volcado de la mayor parte del sistema para una posterior restauración de la situación original, todo esto en el menor tiempo posible. Es también conveniente que se realice esta copia en el menor espacio posible, evitando copiar archivos que no sean útiles para la recuperación del sistema.

22.2. Políticas de copias de seguridad

Tan importante como realizar la copia de seguridad es definir una política de copias de seguridad. Cuando se decide hacer copia de seguridad de los archivos del sistema debe adoptarse un esquema de copia de seguridad o *política de copia de seguridad*. No debe caerse en la tentación de "copiarlo todo, por si acaso" y es necesario hacer un estudio previo valorando la importancia de los datos. Existen muchas estrategias para realizar copias de seguridad y la elección de una determinada, depende de las políticas de copia de seguridad adoptadas. Se busca un equilibrio entre el uso de recursos y la disponibilidad de los datos.

Para implementar una política de seguridad, lo primero que se define son los datos de los cuales va a realizarse la copia. Podrán ser archivos sueltos, directorios o sistemas de archivos completos. El esquema propuesto parte de una copia completa del sistema o *backup* total, que será la primera

copia de seguridad que se haga. A partir de esta copia, el resto se realizará de forma incremental, guardando únicamente los archivos que hayan cambiado desde la copia de seguridad inicial[1].

Esta política, a pesar de su simpleza, aporta un ahorro de medios de almacenamiento. Sería más fácil almacenar todos los datos haciendo copias de seguridad completas diarias, pero:

- ¿podemos permitirnos ese gasto de medios de almacenamiento?
- ¿disponemos del sistema el suficiente tiempo para realizar estas copias de seguridad?
- y lo más importante ¿es necesario copiar los mismos datos a diario sin tener la certeza de que hayan cambiado?

Así, supongamos que el sistema al cual se realiza la copia de seguridad es un servidor web. No parece lógico hacer copias de todas las páginas todos los días cuando solo cambian unas pocas páginas que ocupan unos pocos kilobytes. Incluso puede que únicamente se cambien un día de la semana.

Sin embargo, tampoco se puede restringir excesivamente el uso de medios ya que las sucesivas escrituras en el mismo pueden deteriorarlo y producir errores al intentar recuperar los datos. Del mismo modo, no parece conveniente el utilizar siempre el mismo medio físico para almacenar todas las copias. ¿Qué pasaría si este medio se pierde o deteriora? Se perderían todos los datos almacenados y todas las copias de seguridad que contenía el medio.

Con todas estas premisas se llega a la conclusión que hay que buscar el equilibrio entre los datos que se van a guardar, los medios que se van a utilizar y el coste de los mismos. La siguiente política que se plantea intenta equilibrar estos aspectos sin que prime ninguno, pero no debe utilizarse como estándar de copias de seguridad, es simplemente una primera aproximación. Cada sistema tiene sus particularidades y cada administrador debe crear una política de seguridad a medida para sus sistemas.

Supondremos que se dispone de una unidad de cinta para realizar las copias de seguridad, la cual se corresponde con el dispositivo `/dev/st0`. Se utilizarán 6 cintas, etiquetadas CINTA1 a CINTA6, así se realizará la copia de seguridad cada día en soportes distintos. El proceso comienza el viernes haciendo una copia completa y etiquetando esta cinta como CINTA1. La siguiente copia de seguridad se realizará el lunes sobre la CINTA2 y se hará lo mismo hasta el jueves con el resto de cintas. Así, el viernes se ha conseguido una copia completa en CINTA1 y copias incrementales (una por día) en el resto de cintas. Utilizaremos la CINTA6 para hacer nuevamente una copia completa del sistema. La nueva semana empezaría reutilizando las cintas CINTA2 hasta CINTA5, ya que los datos que almacenan se encuentran en la copia completa de CINTA6. La siguiente copia completa se hará en CINTA1 reutilizándola de nuevo.

A pesar de la simplicidad del esquema anterior, el objetivo de la copia de seguridad se cumple de forma satisfactoria. Se puede realizar una primera modificación a la planificación si es necesario que los datos se almacenen por más de 1 semana, guardando las cintas con copias completas por un periodo superior en lugar de reutilizarlas.

Es importante tener claro durante qué periodo de tiempo van a ser válidos los datos ya que esto influirá en el número de soportes que deben guardarse, lo que repercute en el dinero que se invierte en hacer las copias de seguridad. Así, si se realizan las copias de seguridad de un servidor web, no es necesario almacenar copias de la web por más de 1-2 semanas.

Se ha introducido otro concepto importante en las copias de seguridad, el etiquetado del soporte físico donde se guardan las copias. De nada sirve el llevar a cabo una política de seguridad si luego no es posible identificar dónde está la copia de seguridad correspondiente a un determinado día. El soporte donde se realice una copia de seguridad debe estar correctamente etiquetado, conteniendo la fecha en que se ha realizado la copia, así como el nombre de la política de seguridad de la cual forma parte.

[1] Se copian sólo los datos que cambian con respecto a la copia completa. Estas copias parciales se denominan diferenciales cuando se copian los ficheros que han cambiado respecto la copia anterior (ya sea completa o parcial) o incrementales cuando se copian los datos que han cambiado respecto de la última copia completa.

22.3. Dispositivos de almacenamiento

Hasta ahora se ha hablado de cintas como soporte de almacenamiento de las copias de seguridad. Pueden considerarse como el dispositivo de almacenamiento de copias de seguridad por excelencia, existiendo multitud de fabricantes y formatos. Otro dispositivo que está utilizándose cada vez más para la realización de copias de seguridad son los discos, debido a que la relación coste/capacidad es cada vez más pequeña.

La diversidad de soportes para la realización de copias de seguridad puede representar un problema. Puede darse el caso que dispongamos de un dispositivo ultramoderno que sea muy rápido y con una gran capacidad, pero que no sea accesible desde cualquier sistema. Así, si el sistema que alberga el dispositivo deja de estar operativo no será posible recuperar los datos. El dispositivo de almacenamiento que se elija debe ser lo más estándar posible, de forma que en caso de desastre sea facilmente accesible desde cualquier otro ordenador.

En lo referente a las cintas, han sufrido una evolución considerable desde las clásicas cintas de 9 pistas (aquellas con un elemento circular donde se enrollaba la cinta) a las cintas DAT de 4mm. Es por ello que no se entrará en detalle al no estar a disposición de todo el mundo un sistema de almacenamiento basado en cinta. En los ejemplos se utilizará un dispositivo genérico sin entrar en detalles.

Otro dispositivo que no debe olvidarse es el CDROM, que a pesar de su capacidad limitada de 700Mb es importante tener en cuenta por el bajo coste que representa el soporte. Está claro que mejor opción aún es el DVD como soporte para copias. La evolución del mercado ha producido un abaratamiento de los costes tanto de las unidades grabadoras como del soporte DVD que hace que sea ya una opción a tener en cuenta más interesante que el anterior.

La utilización de un disco duro como dispositivo de almacenamiento de las copias de seguridad puede que sea el método más seguro y barato. La capacidad de los discos cada vez es más grande y su precio no es excesivo. Además, puede instalarse el disco duro en cualquier ordenador y acceder así al mismo sin problemas.

El uso del disco duro puede hacerse desde una perspectiva local o remota:

- Copias locales. Se instala el disco duro en el mismo ordenador del que se quiere realizar una copia de seguridad. De este modo las copias estarán accesibles en todo momento, aunque esto presenta varios inconvenientes. ¿Qué pasa si roban el ordenador? Perderíamos al mismo tiempo los datos originales así como todas las copias que se hayan realizado.

- Copias remotas. Un ordenador dedicado a almacenar las copias de seguridad del resto. Los requerimientos de hardware de este ordenador serán básicos, ya que se encargará únicamente de proporcionar espacio en disco al resto de ordenadores. En este caso las copias de seguridad se realizarán a través de la red.

En ambos casos es necesario identificar de forma clara y precisa la localización de las copias de seguridad. Una opción es tener un directorio con la identificación de cada máquina y dentro del mismo los ficheros que contienen las copias de seguridad, teniendo éstas nombres que hagan referencia tanto a la fecha en que se realizó la copia como a los datos que se copiaron y si la copia era completa o incremental.

22.4. Utilidades de archivado

22.4.1. Utilidad tar

La utilidad del sistema `tar` es un programa de archivado, diseñado para almacenar y extraer ficheros desde un archivo (conocido como `tarfile`). Es decir, una estructura de archivos y directorios se guarda en un solo fichero, con la posibilidad de recuperarla posteriormente. Dicho `tarfile` puede estar alojado en una unidad de cinta o en el propio disco duro del servidor como un fichero más.

Supongamos que queremos hacer una copia de seguridad de los ficheros con información sobre usuarios y claves (`/etc/passwd` y `/etc/shadow`).

```
[hugo@fedora backup]$ tar -cvf backup-password.tar /etc/passwd /etc/shadow
tar: Eliminando la / inicial de los nombres
etc/passwd
tar: /etc/shadow: No se puede open: Permiso denegado
tar: Salida con error demorada desde errores anteriores
```

El error aparecido al intentar hacer una copia de seguridad del archivo `/etc/shadow` es debido a que se ha ejecutado la utilidad `tar` con un usuario normal del sistema (hay que recordar que el fichero `/etc/shadow` sólo es visible por el usuario root).

Cuando se realice copia de seguridad de los archivos es recomendable mantener los datos referentes a dueños de los archivos y permisos de acceso. Por esto es conveniente realizar estas operaciones como `root` o verificar que el usuario que realiza la llamada a la utilidad `tar` tiene permiso de acceso a los ficheros de los que va a hacer copia de seguridad.

```
[root@fedora tmp]# tar -cvf backup-password.tar /etc/passwd /etc/shadow
tar: Eliminando la / inicial de los nombres
etc/passwd
etc/shadow
```

De esta forma, en el fichero `backup-password.tar` estarán almacenados los dos ficheros anteriores. Almacena los nombres con un camino (*path*) relativo, al quitarle la / inicial. Lo hace automáticamente para que luego se puedan restaurar los ficheros en un lugar diferente si así se desea, sin tener que machacar los originales.

Para restaurar dichos ficheros, se utilizará el comando `tar -xvf backup-password.tar`.

22.4.2. Utilidad dump/restore

A pesar de ser bastante antigua, la utilidad `dump` es muy utilizada en el mundo Unix y realiza el volcado de sistemas de ficheros completos. Por ejemplo, si queremos realizar una copia de seguridad con `dump` de los siguientes sistemas de ficheros, será necesario realizar una llamada a `dump` por cada uno de ellos.

```
/
/home
/usr
/var
```

También permite realizar las copias de los sistemas de ficheros entre máquinas remotas (`rdump/rrestore`). Como utilidad complementaria está `restore`, que será la encargada de realizar las recuperaciones de las copias realizadas con `dump`.

Una de las principales ventajas de la pareja de utilidades `dump/restore` es que son compatibles entre los distintos sabores de Linux y Unix. Además, durante la realización de la copia de un sistema de archivos a cinta, si detecta que se va a acabar la cinta, solicitará otra para continuar con el proceso.

Operaciones con dump/restore

La sintaxis general de la utilidad `dump` es:

```
dump opciones argumentos sistema_ficheros
```

Esta sintaxis, similar a la de otros comandos y utilidades, no se verifica en el resto de sistemas operativos Unix que no sean Linux. En el resto de Unix las opciones y argumentos irán agrupados y deberán coincidir en orden y número.

Por ejemplo, para hacer una copia de seguridad de la partición que contiene `/boot/`

```
[root@fedora root]# dump -f /tmp/dumpfile /boot/
DUMP: Date of this level 0 dump: Thu Dec 25 18:07:22 2003
DUMP: Dumping /dev/sda1 (/boot) to /tmp/dumpfile
DUMP: Added inode 8 to exclude list (journal inode)
DUMP: Added inode 7 to exclude list (resize inode)
DUMP: Label: /boot
DUMP: mapping (Pass I) [regular files]
DUMP: mapping (Pass II) [directories]
DUMP: estimated 5422 tape blocks.
DUMP: Volume 1 started with block 1 at: Thu Dec 25 18:07:23 2003
DUMP: dumping (Pass III) [directories]
DUMP: dumping (Pass IV) [regular files]
DUMP: Closing /tmp/dumpfile
DUMP: Volume 1 completed at: Thu Dec 25 18:07:24 2003
DUMP: Volume 1 5410 tape blocks (5.28MB)
DUMP: Volume 1 took 0:00:01
DUMP: Volume 1 transfer rate: 5410 kB/s
DUMP: 5410 tape blocks (5.28MB) on 1 volume(s)
DUMP: finished in 1 seconds, throughput 5410 kBytes/sec
DUMP: Date of this level 0 dump: Thu Dec 25 18:07:22 2003
DUMP: Date this dump completed: Thu Dec 25 18:07:24 2003
DUMP: Average transfer rate: 5410 kB/s
DUMP: DUMP IS DONE
```

Las opciones más utilizadas son las que se muestran en la siguiente tabla:

Opción	Acción realizada	Argumento
0-9	Nivel de la copia de seguridad	NO
u	Actualiza `/etc/dumpdates` al finalizar la copia	NO
f	Indica una cinta diferente de la usada por defecto	SI
b	Tamaño de bloque	SI
c	Indica que la cinta destino es un cartucho	NO
W	Ignora todas las opciones excepto el nivel de la copia	NO
a	Se escribirá hasta el final de la cinta en lugar de calcular el espacio	NO

La copia de seguridad puede realizarse tanto en un dispositivo de cinta (local o remoto) como en un fichero. Cuando la copia se realice a un dispositivo o fichero remoto será necesario indicarle también el nombre del servidor destino.

```
dump -f backuphost:/dev/rmt/0mn
```

El nombre del dispositivo de cinta remoto será el que tenga en el servidor remoto. Será necesario también que el servidor remoto permita el acceso por `rsh` sin necesidad de clave. Esto representa un agujero de seguridad bastante importante, por lo que se recomienda estudiar la conveniencia o no de usar este método para hacer copias remotas.

Para realizar la restauración de las copias realizadas con `dump` se utiliza la utilidad complementaria `restore`. La restauración puede realizarse de forma interactiva o no interactiva.

Restauración interactiva: Este tipo de restauraciones se utilizarán cuando sólo queremos restaurar ficheros sueltos. Será necesario utilizar la opción `-i` para realizar restauraciones de este tipo. Retomando el ejemplo anterior, donde realizamos una copia de seguridad del sistema de archivos `/boot`

```
[root@fedora root]# restore -if /tmp/dumpfile
restore > ls
.:
```

```
System.map              initrd-2.4.18-14.img    module-info-2.4.18-14
System.map-2.4.18-14    kernel.h                os2_d.b
boot.b                  lost+found/             vmlinux-2.4.18-14
chain.b                 message                 vmlinuz
config-2.4.18-14        message.ja              vmlinuz-2.4.18-14
grub/                   module-info
```

Dentro de este modo interactivo existen varios comandos:

```
restore > help
Available commands are:
ls [arg] - list directory
cd arg - change directory
pwd - print current directory
add [arg] - add 'arg' to list of files to be extracted
delete [arg] - delete 'arg' from list of files to be extracted
extract - extract requested files
setmodes - set modes of requested directories
quit - immediately exit program
what - list dump header information
verbose - toggle verbose flag (useful with ``ls'')
prompt - toggle the prompt display
help or '?' - print this list
If no 'arg' is supplied, the current directory is used
```

Restauración no interactiva: En este caso se restaurarán todos los ficheros

Como ya se ha comentado anteriormente, dentro de un mismo dispositivo se pueden tener varias copias de seguridad. Para acceder a una en concreto es necesario moverse de forma secuencial por la cinta. La utilidad que permite desplazarse por el dispositivo de cinta es `mt`. Esta utilidad tiene numerosas opciones y nos centraremos en las más utilizadas para el uso conjunto con `dump/restore`:

- Rebobinar la cinta

  ```
  mt -f tapedev rewind
  ```

- Moverse por la cinta

  ```
  mt -t tapedev fsf count
  ```

Así pues, en el caso que se disponga de una cinta con varias copias de seguridad realizadas con `dump`, si se quiere restaurar la que se encuentra en la segunda posición:

```
mt -f tapedev rewind
mt -t tapedev fsf 1
```

22.5. Sincronización de sistemas de ficheros

La utilidad `rsync` es una pequeña utilidad que nos permite realizar transferencias incrementales de ficheros, de esta forma sólo se copian las diferencias que se han producido entre los datos origen y destino. Estos datos pueden ser enviados comprimidos, y mediante `ssh` si queremos que viajen encriptados. Es condición indispensable que `rsync` se encuentre instalado en las dos máquinas que están involucradas en la transferencia de ficheros.

A modo de resumen, `rsync` basa su funcionamiento en los siguientes conceptos:

Diferencias. Únicamente se transfieren los ficheros que han cambiado en lugar de todos los ficheros y, de los que hayan cambiado, sólo las diferencias lo que hace que la transferencia sea mucho más rápida (conveniente en caso de tener enlaces de red lentos) a diferencia de `ftp` o `scp` que enviaría el archivo completo incluso si sólo cambia 1 byte.

Compresión. Las pequeñas partes de los ficheros que han cambiado se comprimen en el momento, con lo cual el tamaño de los datos a mandar es aún menor.

Encriptación. Es posible que los datos que han cambiado se envíen por la red a través de un canal seguro proporcionado con `ssh`, con lo cual evitamos que cualquier persona pueda capturarlos.

22.5.1. Trabajando con rsync

Básicamente, el funcionamiento es similar a las utilidades de copia remota `rcp` o `scp`. Ofrece grandes ventajas a la hora de realizar copias de seguridad o espejos de determinadas partes del sistema de ficheros de un servidor, especialmente en el caso de servidores web.

La utilidad `rsync` puede funcionar también en modo "`daemon`": a la escucha en un determinado puerto. Esta opción suele utilizarse para la distribución de ficheros en equipos de desarrollo de software. En nuestro caso nos centraremos en la ejecución de `rsync` en modo cliente ya que sacaremos más provecho.

La shell remota que utiliza `rsync` por defecto es `rsh`. En caso de querer cambiar a `ssh` (es lo más conveniente por los motivos de seguridad indicados) es necesario utilizar la opción `-e` en la llamada a `rsync` o establecer la variable de entorno `RSYNC_RSH` a `ssh`.

La mejor forma de comprender el funcionamiento de `rsync` es viendo el resultado de la ejecución de unos ejemplos. Utilizaremos un servidor web y se realizará una copia del directorio `/var/www/html` en `/home/hugo`:

```
[hugo@fedora www]$ rsync -av html /home/hugo/
building file list ... done
html/
html/usage/
html/usage/ctry_usage_200311.png
html/usage/daily_usage_200311.png
html/usage/hourly_usage_200311.png
html/usage/index.html
html/usage/msfree.png
html/usage/usage.png
html/usage/usage_200311.html
html/usage/webalizer.png
wrote 50357 bytes read 148 bytes 33670.00 bytes/sec
total size is 49757 speedup is 0.99
```

La opción `-v` (*verbose*) es conveniente utilizarla ya que informará de la evolución del proceso. El comando anterior sería equivalente a:

```
cp -a html home/hugo/
```

Sin embargo el uso de `rsync` es mucho más eficiente por lo comentado previamente. El ejemplo anterior suponía que tanto el directorio fuente como el directorio destino estaban localizados localmente. En caso que se encuentren en distintas máquinas el formato general es:

```
rsync -a -e ssh fuente/ usuario@maquinaremota:/ruta/a/destino/
```

En este caso se supondrá que el directorio destino se encuentra en la máquina `maquinaremota`.

Consideremos que se está desarrollando el contenido del servidor web en nuestro ordenador personal con Linux (`hugo.midominio.org`). Una vez conformes con las modificaciones hechas a las páginas web es necesario pasarlas al servidor web para que las vea todo el mundo (`www.micentro.org`).

```
rsync -av -e ssh /home/hugo/html/ www.micentro.org:/home/httpd/html/
```

- Aunque realmente no estaría dentro de los conceptos referidos a la utilidad `rsync`, merece la pena detenernos un momento en ver las diferencias existentes entre poner `directorio/` o `directorio` (se ha utilizado de las dos formas en distintos ejemplos) ya que el resultado puede diferir en algunos casos. Normalmente estamos acostumbrados a que los comandos no prestan especial atención a las barras de *path*. Por ejemplo, si `a` y `b` son dos directorios, los siguientes comandos serían equivalentes:

  ```
  cp -a a b
  cp -a a/ b/
  ```

  Sin embargo en el caso de `rsync` estas barras al final de una ruta sí son importantes, pero únicamente en el caso del directorio fuente. De esta forma, si el directorio `a` contiene a su vez un subdirectorio llamado `temp` se obtienen los siguientes resultados:

  ```
  rsync -a a b ⟼ b/a/temp
  rsync -a a/ b ⟼ b/temp
  ```

  Como puede verse, el resultado no es el mismo, así que debe tenerse en cuenta este aspecto para poner o no la barra de *path* al final del directorio origen.

Otra opción muy interesante a la hora de realizar copias con la utilidad `rsync` es `--delete`. Continuemos con el ejemplo anterior y supongamos que un fichero que estaba en el directorio `html` ha sido borrado porque ya no era necesario. La copia existente en el directorio `html` del servidor destino debería ser igual que la del directorio del que tomó los datos. Esto se consigue borrando el fichero del servidor de destino mediante:

```
rsync -av --delete -e ssh /home/hugo/html/ www.micentro.org:/home/httpd/html
    /
receiving file list ... done
deleting html/index.html
html/
wrote 16 bytes  read 395 bytes  91.33 bytes/sec
total size is 94287  speedup is 229.41
```

Como puede verse, hay una referencia al fichero `index.html` que se ha borrado al haber sido borrado previamente del origen. Como con cualquier operación que implique borrado de ficheros se recomienda utilizar con cuidado esta opción, asegurándonos que realmente es eso lo que se desea.

Otra posibilidad que ofrece `rsync` a la hora de realizar transferencias selectivas de ficheros es la opción `--exclude` y `--exlude-from`. Con la primera de las opciones, se excluirán de la copia los ficheros que verifiquen el patrón, mientras que con la segunda opción los ficheros a excluir se obtendrán de un fichero auxiliar.

```
rsync -av --exclude '*.bak' /home/hugo/html/ www.centro.org:/home/httpd/html
    /
rsync -av --exclude-from excluir.txt /home/hugo/html/ www.micentro.org:/home
    /httpd/html/
```

22.5.2. Copias de seguridad con rsync

Ahora que ya se ha trabajado un poco con `rsync` se aplicará a la realización de copias de seguridad. Como ya se vió, `rsync` proporciona un mecanismo cómodo para realizar copias de seguridad tanto en local[2] como en una máquina remota.

[2] En este caso, lo recomendable es que se realice en un sistema de archivos y disco independiente

Usando `rsync` junto a la utilidad `cron` se puede planificar una copia de seguridad para que se realice todos los días.

```
00 22 * * * rsync -a --delete -e ssh fuente/ usuario@maquinaremota:/path/to/destino/
```

De esta forma, todos los días a las 22:00 se realizaría una copia de seguridad del directorio en cuestión, teniendo en cuenta que únicamente viajarán por la red los cambios existentes.

Sin embargo, si se produce el borrado accidental de un fichero y se realiza la copia de seguridad, este fichero se borrará en el directorio destino, con lo que sería imposible su recuperación. Surge así la necesidad de avanzar un poco más, como ya se ha hecho anteriormente con otras utilidades, a la hora de definir el esquema de copias de seguridad.

No debe olvidarse que es recomendable hacer una copia completa al menos una vez a la semana y el resto de días realizar copias incrementales.

22.6. Copias de seguridad en CDROM

Al comienzo de este capítulo se indicó que, debido al bajo precio tanto del dispositivo de grabación como del soporte, se está extendiendo el uso del CDROM para la realización de las copias de seguridad en determinados casos.

En este caso no se utilizarán las utilidades vistas hasta ahora, será necesario hacer uso de una utilidad diseñada específicamente para realizar escritura sobre soporte CDROM.

Se asumirá que el sistema operativo ha reconocido de forma correcta el dispositivo. A partir de este punto es necesario un software capaz de utilizar la unidad para escribir datos. Por un lado una utilidad para crear las imágenes ISO (`mkisofs`) y por otro la utilidad para realizar el proceso de grabación.

El proceso a seguir en la realización de las copias de seguridad será crear una imagen con los datos que se desea copiar y posteriormente grabar esta imagen en el CDROM.

Supongamos que vamos a hacer la copia de seguridad del directorio `/home`:

```
[hugo@fedora hugo]$ ls -l /home
total 8
drwx------ 20 hugo hugo 4096 dic 27 09:33 hugo
drwx------ 2 pepito pepito 4096 nov 8 11:21 pepito
[hugo@fedora home]$ mkisofs -R -l -o /mnt/imagen.iso /home/
mkisofs: Permission denied. Unable to open directory /home/pepito
mkisofs: Permission denied. Unable to open disc image file
```

Nuevamente aparece un problema de permisos ya que el usuario `hugo` está intentando acceder a un directorio `/home/pepito` en el que no tiene permiso de acceso. Lo mismo pasa con el fichero que va a almacenar la imagen ISO. Para evitar este tipo de problemas se realizarán las copias utilizando el usuario `root`.

```
[root@fedora root]# mkisofs -R -l -o /mnt/imagen.iso /home
54.56% done, estimate finish Sat Dec 27 09:40:51 2003
Total translation table size: 0
Total rockridge attributes bytes: 63340
Total directory bytes: 313344
Path table size(bytes): 2314
Max brk space used 5c544
9168 extents written (17 Mb)
```

No es necesario entrar en detalle de todas las opciones que puede utilizar `mkisofs`, únicamente van a describirse las que puedan ser más útiles[3]:

[3]Respecto a la columna **Opción** hay que aclarar que aunque en la documentación aparecen las opciones `-m` y `-x` como distintas, son equivalentes y se puede utilizar una u otra indistintamente

Opción	Acción realizada
`-f`	Sigue los enlaces simbólicos cuando se genera el sistema de ficheros
`-l`	Permite nombres de ficheros de 31 caracteres
`-L`	Permite nombres de ficheros que empiecen por .
`-m patrón`	Excluye los nombres de ficheros que concuerden con el parámetro
`--exclude-list fichero`	Excluye los nombres de ficheros que concuerden con los patrones de fichero
`-o fichero`	Nombre del fichero donde se va a almacenar la imagen ISO
`-R`	Se registra información sobre permisos y dueños de los archivos
`-v`	Muestra información detallada sobre la creación de la imagen ISO
`-x ruta`	Excluye los ficheros que coinciden con la ruta especificada

Se puede afinar un poco más la copia de seguridad y excluir ficheros o directorios que no quieran incluirse en las copias de seguridad:

```
# mkisofs -R -l -x *~ -x /home/*/.openoffice -o /mnt/imagen.iso /home
59.10% done, estimate finish Sat Dec 27 09:55:32 2003
Total translation table size: 0
Total rockridge attributes bytes: 53335
Total directory bytes: 260096
Path table size(bytes): 1938
Max brk space used 4e000
8464 extents written (16 Mb)
```

Antes de grabar la imagen a CDROM es recomendable comprobar su contenido:

```
# mount /mnt/imagen.iso -r -t iso9660 -o loop /mnt/home
# ls -l /mnt/home/
total 14
drwx------ 20 hugo hugo 8192 dic 27 09:39 hugo
drwx------ 2 pepito pepito 2048 nov 8 11:21 pepito
dr-xr-xr-x 18 root root 4096 dic 27 09:40 rr_moved
```

Una vez montada la imagen con la opción `-o loop` es posible operar sobre ella como cualquier sistema de archivos. Se podrá añadir, borrar o realizar modificaciones sobre los ficheros y, una vez desmontado, estas modificaciones se mantendrán en el archivo de imagen ISO.

En este punto ya se dispone de una imagen creada y lista para almacenar en CDROM. Entra en juego ahora la utilidad de cdrom para realizar la grabación de los datos.

k3b

Si bien, el modo comando permite trabajar de forma eficiente y eficaz, las utilidades gráficas de creación de imágenes ISO y de grabación nos pueden facilitar enormemente la vida a la hora de hacer lo comentado. Existen varias de funcionalidad similar pero nos pararemos sólo en una de las mejores y más usadas, se trata de `k3b`[4]. K3b, lo mismo que xcdroast, es un *front-end* para los programas de grabación de siempre (`cdrecord, cdrdao, mkisofs y cdparanoia`) pero incorpora además las utilidades `dvd+rw-tools` y `growisofs` para hacer copias de DVD. Su interfaz gráfica es muy amigable e intuitiva y se tienen a mano todas las herramientas de grabación.

[4]

- La página oficial del programa es `http://www.k3b.org`. Para instalarla

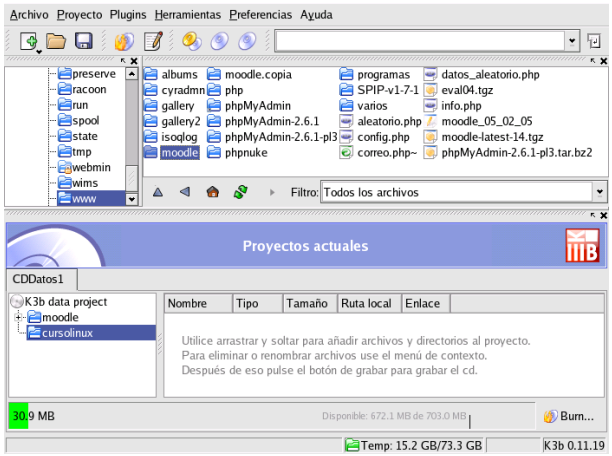
  ```
  #apt-get install k3b
  ```

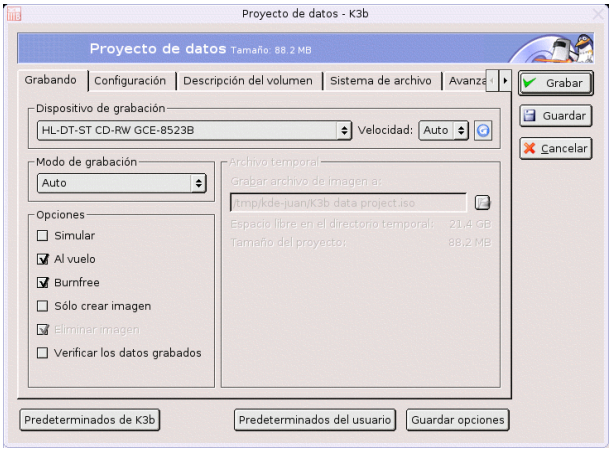
- Partiremos de que está instalada y configurada y sólo ejemplificaremos los aspectos que nos interesan respecto a copias de seguridad.

Figura 22.1: Inicio de k3b

Con ella podemos tanto crear la imagen `iso` de la zona de disco que deseemos como finalizar el proceso de grabación.

Ejecutemos k3b y seleccionemos **Archivo→Nuevo Proyecto → Nuevo proyecto de CD de Datos**[5]. Arrastremos las carpetas y ficheros que deseemos al panel del proyecto. Cuando hayamos elegido las que deseamos[6] pulsaremos sobre **Burn..** En la ventana ***Proyecto de datos*** accedemos a las pestañas de opciones de grabación: podemos copiar al vuelo, sólo crear imagen, iniciar multisesión, poner nombre al Cd, etc. Cuando esté todo a nuestro gusto, optaremos por [**Grabar**]. Si en ese momento no deseamos hacer la grabación podemos guardar el proyecto **Archivo→Guardar como...** darle nombre y grabarlo más tarde.





22.7. Solución integrada de copias de seguridad: AMANDA

Uno de los problemas con que se encuentran a menudo los administradores de grandes y pequeñas redes es la realización de copias de seguridad. Hacer que cada usuario haga copia de seguridad de sus propios archivos parece una solución obvia, pero es impracticable. Además, equipar a cada estación de trabajo y servidor de su propio hardware de copia de seguridad es muy caro.

[5] O DVD de datos.

[6] Si pulsamos con el botón derecho del ratón sobre un fichero, un menú emergente nos permitirá renombrar, borrar, crear carpeta, etc.

Podría enfocarse la solución a este problema desde un punto de vista centralizado, donde un único servidor, el servidor de *backup*, controla un proceso de copia por la red. Con este punto de vista, una vez configurado un cliente, no es necesaria la intervención del administrador y podría constituirse como solución válida.

AMANDA (*Advanced Maryland Automated Network Disk Archiver*) es una utilidad de dominio público desarrollada por la Universidad de Maryland que toma esta aproximación centralizada. AMANDA permite establecer un único servidor maestro para gestionar las copias de seguridad de múltiples nodos en una única unidad de cinta. Utiliza las utilidades `dump` y `tar`, pudiendo realizar las copias de seguridad de clientes Unix con distintas versiones de Sistema Operativo. Las versiones más recientes pueden también hacer copias de clientes Windows mediante SAMBA.

La idea surgió por la problemática existente con la introducción de los soportes de cinta de alta capacidad. En el caso de organizaciones con muchos servidores, no es rentable tener una unidad de cinta conectada a cada servidor para realizar las copias de seguridad. Con este sistema únicamente el servidor maestro tiene conectadas las unidades de cinta.

Aunque inicialmente se diseñó para trabajar con unidades de cinta, se realizarán unas modificaciones en la configuración para utilizar el disco duro como soporte final de las copias de seguridad. El precio que actualmente tienen los discos, así como la proliferación de los discos externos con conexión USB hace que se puedan considerar un serio aspirante a medio de almacenamiento de copia de seguridad.

El sistema de copias de seguridad que proporciona AMANDA permite hacer copias en un único soporte de varios ordenadores. Este sistema gestiona copias incrementales y completas, aunque normalmente sólo se hacen copias completas de pocos ordenadores en cada sesión de copias de seguridad.

22.7.1. Características de AMANDA

AMANDA está diseñado para manejar un gran número de clientes y datos, siendo aún así facil de instalar y configurar. Es fácilmente escalable, así que pequeñas configuraciones, de incluso un único nodo, son posibles. Su funcionamiento está sustentado por una serie de comandos que se encargan de diversas tareas relativas a la copia de seguridad. Las posibilidades de AMANDA son extraordinariamente amplias, limitándonos aquí a dar unos conceptos básicos que nos permitan dar los primeros pasos en su uso.

Trabaja sobre un protocolo propio encima de TCP y UDP. Cada cliente escribirá en la salida estándar, siendo AMANDA el encargado de recoger y transmitir dicha información al servidor maestro. Esto permitirá insertar compresión y encriptación así como la gestión de un catálogo con las imágenes creadas para su uso en la restauración. El cliente de red existente en el servidor de *backup* de AMANDA, contacta con los servidores AMANDA que se ejecutan en los clientes de los que se va a hacer copia de seguridad para iniciar la transferencia de datos. Para la restauración de los datos, AMANDA ejecuta varios servicios en el servidor de *backup* y un programa cliente (en el ordenador cliente del cual se quiere restaurar).

Múltiples clientes entran en el proceso de copia en paralelo al copiar por la red los datos a uno o varios discos, configurados en el servidor maestro para almacenar las imágenes de las copias. Posteriormente, el proceso de copia sobre el dispositivo de cinta toma como origen estos discos. AMANDA puede ejecutarse directamente sobre cinta sin utilizar este almacenamiento intermedio, pero se logrará un rendimiento peor.

Al utilizar software estándar para generar las imágenes, sólo las herramientas Unix como `mt`, `dd` y `gunzip` son necesarias para restaurar una copia desde cinta, si AMANDA no está disponible.

AMANDA utiliza una secuencia de cintas para las copias. Las cintas están numeradas con etiquetas que son leídas por los servidores. Así, AMANDA gestiona las cintas que son necesarias utilizar en cada sesión, evitando escribir en una cinta que no corresponda.

Este sistema realiza las copias basándose en el espacio en disco reservado por AMANDA en el servidor de cintas y que almacenará las copias hechas por la red. De esta manera, las copias de los servidores clientes se volcarán a disco mientras la unidad de cinta trabaja.

Una vez configurado, AMANDA necesita muy poca intervención por parte del administrador. Normalmente existirá una línea en el `cron` del sistema, que será el encargado de lanzar el proceso de copia. Los clientes que estén colgados o no estén disponibles en el momento de la copia son marcados e ignorados.

Cada mañana el administrador del sistema recibirá un informe con los detalles del proceso de copia de seguridad realizado durante la noche. Las primeras líneas son las más importantes, ya que indicarán si el proceso finalizó correctamente y se escribieron los archivos en la cinta correcta o si hubo algún fallo (pudo haberse realizado bien el proceso de volcado a disco pero fallar la copia a cinta). En el caso de fallo en el dispositivo de cinta, el sistema pasa a un estado degradado, donde las copias de seguridad se siguen realizando, pero únicamente a disco.

A continuación veremos una configuración básica de AMANDA que nos permita empezar a realizar copias de seguridad.

22.7.2. Instalación de AMANDA

La instalación de los distintos paquetes, así como de las dependencias que éstos presenten se realizará por el procedimiento habitual:

```
apt-get install amanda-common
apt-get install amanda-server
apt-get install amanda-client
```

La infraestructura que se describe a continuación se basa en un sistema Guadalinex que ejerce al mismo tiempo de servidor de cintas y de cliente de backup. No debe presentar dificultad ampliar este esquema mediante la configuración de nuevos clientes al sistema, según lo indicado en el apartado correspondiente.

22.7.3. Configuración de clientes

Los clientes de AMANDA ejecutan un servicio llamado `amandad`. Normalmente se ejecutará desde `inetd` o `xinetd`.

En Fedora, al trabajar con `xinetd` es necesario crear el fichero `/etc/xinetd.d/amanda`:

```
# default: off
# description: The client for the Amanda backup system.\
#              This must be on for systems being backed up\
#              by Amanda.
service amanda
{
        socket_type             = dgram
        protocol                = udp
        wait                    = yes
        user                    = backup
        group                   = backup
        server                  = /usr/lib/amanda/amandad
        disable                 = no
}
```

Por supuesto, dependiendo de la instalación de AMANDA, puede ser necesario cambiar la ruta donde se encuentra el programa servidor.

Es recomendable modificar también el fichero `/etc/services` para que refleje el nuevo servicio, en caso de no estar definido:

```
amanda          10080/tcp       # amanda backup services
amanda          10080/udp       # amanda backup services
```

También es necesario comprobar que haya una cuenta de sistema para AMANDA. Normalmente, el nombre de usuario de esta cuenta será `amanda`, aunque en el caso de Guadalinex la cuenta que se crea en la instalación y que se utilizará es `backup`. Si se va a utilizar otra cuenta distinta de `backup` pueden producirse problemas referentes a permisos con los ficheros y directorios que utiliza AMANDA. El directorio `$HOME` de este usuario deberá contener un fichero de autenticación llamado `.amandahost,` que en realidad es un enlace simbólico al fichero `/etc/amandahost`. Contendrá el nombre del servidor de *backup* y el usuario con el que este servidor accederá al servicio que tenemos ejecutándose en el cliente de *backup*. Proporciona un control de acceso sin el cual AMANDA no podrá realizar la copia de seguridad. Teniendo en cuenta que el servidor tiene como nombre de host `guadalinex` y que el usuario que se utiliza es `backup`, el contenido de este fichero será:

```
guadalinex backup
```

Una vez configurado el cliente con las indicaciones que acabamos de dar, es preciso reiniciar `inetd` o `xinetd` para que los cambios tomen efecto. Este proceso es necesario repetirlo en cada uno de los clientes que se desee salvaguardar con AMANDA.

22.7.4. Configuración del servidor de cintas

Lo primero que hay que decidir, una vez instalado AMANDA, es qué máquina será el servidor de cintas. Necesitará tener acceso directo al dispositivo de cintas y un espacio en disco lo suficientemente grande para almacenar las imágenes. En el caso que nos ocupa no se accederá a ningún dispositivo de cintas ya que éstas se simularán sobre el espacio en disco.

La máquina que ejerza de servidor de *backup* no es necesario que ejecute ningún servicio para realizar las operaciones de copia de seguridad. Únicamente necesita ejecutar servicios para la gestión de restauraciones iniciadas desde los clientes. Al igual que en el caso de los clientes, estos servicios se gestionarán desde `inetd` o `xinetd`. En el caso de `xinetd` son necesarios los ficheros `/etc/xinetd.d/amandaidx`:

```
# default: off
#
# description: Part of the Amanda server package
service amandaidx
{
        socket_type             = stream
        protocol                = tcp
        wait                    = no
        user                    = backup
        group                   = backup
        server                  = /usr/lib/amanda/amindexd
        disable                 = no
}
```

Y el fichero `/etc/xinetd.d/amidxtape`:

```
# default: off
#
# description: Part of the amanda server package
#
service amidxtape
{
        socket_type             = stream
        protocol                = tcp
        wait                    = no
        user                    = backup
        group                   = backup
        server                  = /usr/lib/amanda/amidxtaped
```

```
        disable                 = no
}
```

Las correspondientes entradas en `/etc/services` serán:

```
amandaidx       10082/tcp               # amanda backup services
amidxtape       10083/tcp               # amanda backup services
```

La configuración para gestionar las copias de seguridad se define en `amanda.conf`. Este fichero se encuentra localizado en un subdirectorio que indica el nombre de la copia de seguridad dentro de `/etc/amanda`. Por ejemplo, si se define una política de copias de seguridad que se llama `Daily`, el fichero de configuración debería estar en `/etc/amanda/Daily/amanda.conf`. Este subdirectorio debe tener permisos de escritura para el usuario `backup`.

Algunos de los parámetros de configuración de este fichero son:

`dumpuser` Especifica el nombre de usuario con el que se ejecutarán las operaciones de copia de seguridad.

`dumpcycle` Define el tiempo que AMANDA toma para realizar un *backup* completo del sistema.

`runspercycle` Es el número de veces que `amdump` se ejecuta en cada ciclo.

`tapecycle` Es el número de cintas que van a ser usuadas en un único ciclo de carga. Este número suele ser mayor que `runspercycle` por si hay cintas dañadas.

`runtapes` Es el número de cintas que se usan en cada ejecución de `amdump`. Normalmente es 1 si no hay cargador de cintas.

`tapedev` Es el dispositivo de cinta sin rebobinado, por ejemplo `/dev/nst0` o `/dev/nht0`.

`tapetype` Es el tipo de dispositivo de cinta.

`labelstr` Establece la etiqueta para las cintas.

`holdingdisk` Establece información sobre el espacio en disco reservado para las copias de seguridad.

La parte más complicada de la configuración es establecer el ciclo de copia. Los parámetros de esta categoría interaccionan para definir el número de cintas requeridas para cada ciclo y el número de cintas disponibles en total. Los parámetros `dumpcycle`, `runspercycle` y `tapecycle` se definen de forma que se hace una copia de cada sistema de ficheros al menos una vez por `dumpcycle` y el número de cintas en `tapecycle` debe ser más grande que `runspercycle * runtapes`. Al estar establecido `runtapes` a 1, si no tenemos un cargador de cintas, el valor de `tapecycle` dependerá únicamente de `runspercycle`.

Otros ficheros que podemos encontrar dentro de este directorio `/etc/amanda/Daily` son[7]:

`disklist` Es un fichero que puede editarse, contiene la combinación de servidor/disco para las copias de seguridad con el formato `hostname diskdev dumptype [spindle [interface]]`

`tapelist` No es editable, contiene el nombre, estado y fecha de último uso para todas las cintas de la configuración.

`tapelist.amlabel` No editable, contiene el estado original de las cintas cuando fueron etiquetadas.

[7] Algunos de estos ficheros no estarán disponibles hasta que no se haya realizado la configuración completa o se haya realizado algún backup. En caso de no encontrarlos en el sistema completar el proceso de configuración y volver a comprobarlos.

`tapelist.yesterday` No editable, contiene el nombre, estado y fecha de último uso (desde la última ejecución de AMANDA) para todas las cintas definidas en la configuración.

Los ficheros `tapelist.*` son gestionados por las utilidades `amdump` y `amlabel` y se crearán conforme se utilice AMANDA. No deben editarse de forma manual.

El fichero de configuración que se muestra a continuación facilitará bastante la comprensión de los conceptos anteriores. También hay que tener en cuenta que AMANDA está diseñado para realizar copias de seguridad sobre dispositivos de cinta. Se efectuarán unas modificaciones en la definición de los dispositivos de cinta que proporciona AMANDA por defecto para utilizar el disco duro local. Así no será necesario comprar ningún dispositivo de cinta y el uso del sistema se basará en escritura sobre cintas virtuales que están definidas sobre espacio en disco.

El fichero `/etc/amanda/Daily/amanda.conf` quedaría como sigue:

```
#
org "Daily"              # Nombre descriptivo para los mensajes
mailto "amanda"          # lista de mails que reciben los logs
dumpuser "backup"        # usuario propietario de los backups
inparallel 1             # procesos en paralelo
netusage  10             # ancho de banda ámximo
dumpcycle 14 days        # numero de dias de un ciclo completo
tapecycle 14             # numero total de cintas
runtapes  1              #
tpchanger "chg-multi"    # script controlador de cintas
changerfile "/etc/amanda/Daily/changer.conf" # óconfiguracin de las cintas
tapetype HARD-DISK               # tipo de almacenamiento
labelstr "^HISS[0-9][0-9]*$"     # expresion regular de la etiqueta de las
    cintas
infofile "/var/lib/amanda/Daily/curinfo"    # fichero de datos
logfile  "/var/log/amanda/Daily/log"        # fichero de log
indexdir "/var/lib/amanda/Daily/index"  # fichero de indice
# ódefinicin del almacenamiento
define tapetype HARD-DISK {
        comment "Esto es un disco duro, no una cinta"
        length 4000 mbytes  # 4 GB de espacio
}
# ódefinicin de volcado de datos completo
define dumptype hard-disk-dump {
        comment "Backup en disco en lugar de cinta - usando dump"
        holdingdisk no
        index yes
        options compress-fast, index, exclude-list "/etc/amanda/exclude.gtar
            "
        priority high
}
# ódefinicin de volcado de datos con 'tar'
define dumptype hard-disk-tar {
        program "GNUTAR"
        hard-disk-dump
        comment "Backup en disco en lugar de cinta - usando tar"
}
```

En la configuración anterior hemos hecho referencia al fichero `/etc/amanda/Daily/changer.conf`, encargado de la configuración de las cintas. En la configuración que nos atañe, definirá la localización del espacio en disco correspondiente a cada cinta virtual.

```
multieject 0
gravity 0
needeject 0
```

```
ejectdelay 0
statefile /var/lib/amanda/Daily/changer-status
firstslot 1
lastslot 14
slot 1 file:/backups/tape01
slot 2 file:/backups/tape02
slot 3 file:/backups/tape03
slot 4 file:/backups/tape04
slot 5 file:/backups/tape05
slot 6 file:/backups/tape06
slot 7 file:/backups/tape07
slot 8 file:/backups/tape08
slot 9 file:/backups/tape09
slot 10 file:/backups/tape10
slot 11 file:/backups/tape11
slot 12 file:/backups/tape12
slot 13 file:/backups/tape13
slot 14 file:/backups/tape14
```

Se define cada cinta virtual como un directorio dentro de `/backups/`, siendo el número de cintas de las que disponemos 14. A efectos prácticos AMANDA no notará la diferencia.

Ya estaría configurado AMANDA, pero no hay que olvidarse de crear los directorios en los que se mapean las cintas virtuales. Lo primero será crear los directorios que simularán las cintas.

```
root@guadalinex:~# mkdir /backups
root@guadalinex:~# mkdir -p /backups/tape01/data
root@guadalinex:~# mkdir -p /backups/tape02/data
[...]
root@guadalinex:~# mkdir -p /backups/tape14/data
root@guadalinex:~# chown -R backup:backup /backups
```

Cuando se crea un conjunto o *pool* de cintas es necesario etiquetarlas para que AMANDA pueda referenciarlas de forma unívoca. Las cintas se etiquetan con el comando `amlabel` y la etiqueta deberá seguir la expresión regular definida en `amanda.conf`. Todas las cintas definidas en `tapecycle` deben ser etiquetadas con objeto de estar disponibles en la fase de testeo. El comando `amlabel` graba cada cinta etiquetada en `tapelist` de forma que los otros programas de AMANDA pueden identificarlas. Es necesario crear el fichero vacío `/etc/amanda/Daily/tapelist` mediante el comando `touch` y posteriormente ejecutar `amlabel` para etiquetar cada una de las cintas:

```
-bash-2.05b$ /usr/sbin/amlabel Daily HISS01 slot 1
backup@guadalinex:/etc/amanda/Daily$ /usr/sbin/amlabel Daily HISS01 slot 1
labeling tape in slot 1 (file:/backups/tape01):
rewinding, reading label, not an amanda tape
rewinding, writing label HISS01, checking label, done.
-bash-2.05b$ /usr/sbin/amlabel Daily HISS02 slot 2
[...]
-bash-2.05b$ /usr/sbin/amlabel Daily HISS14 slot 14
```

Por último, se definen los clientes con los ficheros de los que se va a realizar la copia de seguridad en `/etc/amanda/Daily/disklist`. En este ejemplo únicamente definimos un cliente:

```
# El nombre de la ámquina debe ser el que aparezca en el DNS o /etc/hosts
guadalinex.midominio.org     /home/hugo/CICA/ed05     hard-disk-tar
```

22.7.5. Salvaguarda de datos con AMANDA

Ya parece que está preparado el sistema para realizar copias de seguridad, pero es conveniente realizar un chequeo previo con `amcheck`. No hay que preocuparse si la salida no es exactamente igual a la que se muestra a continuación. Lo realmente importante es que no aparezca ningún mensaje de error[8].

```
backup@guadalinex:/etc/amanda/Daily$ /usr/sbin/amcheck Daily
Amanda Tape Server Host Check
---------------------------------
ERROR: log dir /var/log/amanda/Daily: not writable
amcheck-server: slot 14: date X        label HISS14 (new tape)
NOTE: skipping tape-writable test
Tape HISS14 label ok
NOTE: info dir /var/lib/amanda/Daily/curinfo: does not exist
NOTE: it will be created on the next run
NOTE: index dir /var/lib/amanda/Daily/index/guadalinex.midominio.org: does
    not exist
Server check took 1.164 seconds
Amanda Backup Client Hosts Check
--------------------------------
ERROR: guadalinex.midominio.org: [Can't open exclude file '/etc/amanda/
    exclude.gtar': No such file or directory]
Client check: 1 host checked in 0.382 seconds, 1 problem found
(brought to you by Amanda 2.4.4p3)
```

El error aparecido sobre permisos del directorio que almacena los logs es facilmente solucionable. Se crea el directorio en cuestión y se le asigna como dueño al usuario `backup` y al grupo `backup`. El segundo error hace referencia al fichero `/etc/amanda/exclude.gtar`, que no ha sido creado aún. Basta con ejecutar `touch /etc/amanda/exclude.gtar` para crearlo.

Una vez realizadas estas modificaciones se ejecuta de nuevo `amcheck` para comprobar que no aparecen más errores.

```
backup@guadalinex:/etc/amanda/Daily$ /usr/sbin/amcheck Daily
Amanda Tape Server Host Check
---------------------------------
amcheck-server: slot 14: date X        label HISS14 (new tape)
NOTE: skipping tape-writable test
Tape HISS14 label ok
NOTE: info dir /var/lib/amanda/Daily/curinfo: does not exist
NOTE: it will be created on the next run
NOTE: index dir /var/lib/amanda/Daily/index/guadalinex.midominio.org: does
    not exist
Server check took 0.858 seconds
Amanda Backup Client Hosts Check
--------------------------------
Client check: 1 host checked in 10.272 seconds, 0 problems found
(brought to you by Amanda 2.4.4p3)
```

Al no existir errores puede ejecutarse la primera copia de seguridad. Para eso se utiliza el comando `amdump`. Es necesario indicarle el esquema de copia de seguridad que se desea ejecutar.

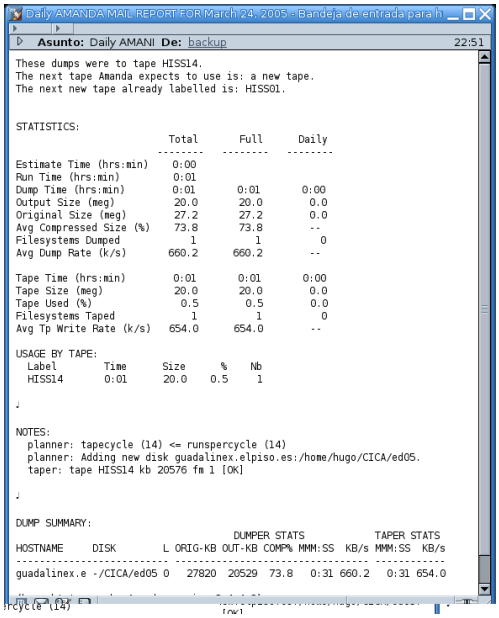
```
backup@guadalinex:/etc/amanda/Daily$ /usr/sbin/amdump Daily
```

Hasta ahora la ejecución de todos los comandos de AMANDA se ha realizado con el usuario `backup`. Esto es importante y no debe olvidarse a la hora de ejecutar estos comandos desde cualquier script auxiliar que se utilice.

[8] Mirando las páginas del manual se comprueba que `amcheck` puede mandar este informe por correo en lugar de mostrarlo por pantalla.

En la configuración que se definió en `amanda.conf` se indicó el destinatario de los informes que genera AMANDA al finalizar una copia de seguridad.

Figura 22.2: Informe de copia de seguridad sin errores de AMANDA



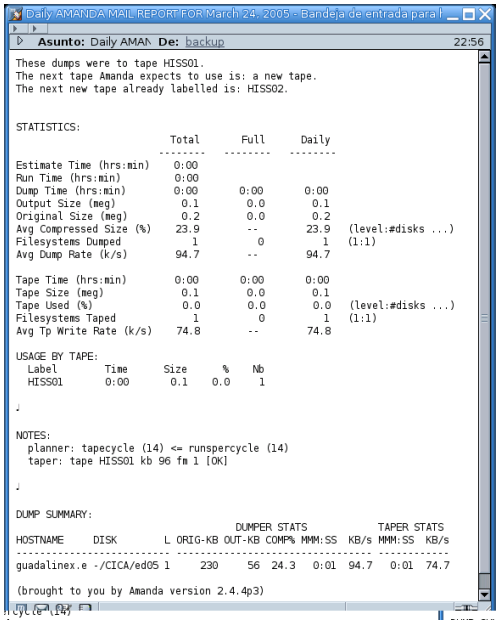
Daily AMANDA MAIL REPORT FOR March 24, 2005 - Bandeja de entrada para h

Asunto: Daily AMANI **De:** backup 22:51

```
These dumps were to tape HISS14.
The next tape Amanda expects to use is: a new tape.
The next new tape already labelled is: HISS01.


STATISTICS:
                          Total       Full      Daily
                        --------   --------   --------
Estimate Time (hrs:min)    0:00
Run Time (hrs:min)         0:01
Dump Time (hrs:min)        0:01       0:01       0:00
Output Size (meg)          20.0       20.0        0.0
Original Size (meg)        27.2       27.2        0.0
Avg Compressed Size (%)    73.8       73.8         --
Filesystems Dumped            1          1          0
Avg Dump Rate (k/s)       660.2      660.2         --

Tape Time (hrs:min)        0:01       0:01       0:00
Tape Size (meg)            20.0       20.0        0.0
Tape Used (%)               0.5        0.5        0.0
Filesystems Taped             1          1          0
Avg Tp Write Rate (k/s)   654.0      654.0         --

USAGE BY TAPE:
  Label       Time      Size      %    Nb
  HISS14      0:01      20.0    0.5     1


NOTES:
  planner: tapecycle (14) <= runspercycle (14)
  planner: Adding new disk guadalinex.elpiso.es:/home/hugo/CICA/ed05.
  taper: tape HISS14 kb 20576 fm 1 [OK]


DUMP SUMMARY:
                                       DUMPER STATS            TAPER STATS
HOSTNAME     DISK        L ORIG-KB OUT-KB COMP% MMM:SS  KB/s MMM:SS  KB/s
-------------------------- ---------------------------------- ------------
guadalinex.e -/CICA/ed05 0   27820  20529  73.8   0:31 660.2   0:31 654.0
```

El informe de esta primera copia de seguridad es bastante completo y permite conocer todos los aspectos del proceso. Si a continuación se lanza de nuevo la copia de seguridad el informe obtenido es distinto.

Figura 22.3: Informe de copia de seguridad de AMANDA



Daily AMANDA MAIL REPORT FOR March 24, 2005 - Bandeja de entrada para l

Asunto: Daily AMAN **De:** backup 22:56

```
These dumps were to tape HISS01.
The next tape Amanda expects to use is: a new tape.
The next new tape already labelled is: HISS02.


STATISTICS:
                          Total       Full      Daily
                        --------   --------   --------
Estimate Time (hrs:min)    0:00
Run Time (hrs:min)         0:00
Dump Time (hrs:min)        0:00       0:00       0:00
Output Size (meg)           0.1        0.0        0.1
Original Size (meg)         0.2        0.0        0.2
Avg Compressed Size (%)    23.9         --       23.9   (level:#disks ...)
Filesystems Dumped            1          0          1   (1:1)
Avg Dump Rate (k/s)        94.7         --       94.7

Tape Time (hrs:min)        0:00       0:00       0:00
Tape Size (meg)             0.1        0.0        0.1
Tape Used (%)               0.0        0.0        0.0   (level:#disks ...)
Filesystems Taped             1          0          1   (1:1)
Avg Tp Write Rate (k/s)    74.8         --       74.8

USAGE BY TAPE:
  Label       Time      Size      %    Nb
  HISS01      0:00       0.1    0.0     1


NOTES:
  planner: tapecycle (14) <= runspercycle (14)
  taper: tape HISS01 kb 96 fm 1 [OK]


DUMP SUMMARY:
                                       DUMPER STATS            TAPER STATS
HOSTNAME     DISK        L ORIG-KB OUT-KB COMP% MMM:SS  KB/s MMM:SS  KB/s
-------------------------- ---------------------------------- ------------
guadalinex.e -/CICA/ed05 1     230     56  24.3   0:01  94.7   0:01  74.7

(brought to you by Amanda version 2.4.4p3)
```

Puede comprobarse que en este caso se ha cambiado de cinta y que el volumen de datos copiados ha sido menor. Efectivamente, el esquema definido está funcionando tal como se desea y se produce el cambio de cinta en esta segunda copia, en la que unicamente se copian los datos que han sido modificados.

Ya puede añadirse la línea anterior al cron del sistema para que se ejecute la copia de seguridad a la hora que establezcamos, preferentemente por la noche, para no interferir con otros procesos.

22.7.6. Recuperación de datos con AMANDA

Según los informes anteriores, se ha realizado la primera copia de seguridad con éxito. Sin embargo, todos los sistemas de copia de seguridad tienen que ser comprobados de forma periódica en lo referente a la restauración.

AMANDA proporciona el comando `amrecover` para la restauración de los datos de una salvaguarda. Al ejecutar `amrecover` se obtiene una sesión interactiva en la que especificando una fecha, un servidor y un disco podemos caminar por los archivos disponibles para restaurar. A diferencia de los anteriores, este comando debe ser ejecutado con el usuario `root` para poder acceder a cualquier archivo del sistema. Será necesario entonces modificar el fichero `.amandahosts` para permitir que el usuario `root` pueda acceder a los datos de AMANDA.

```
backup@guadalinex:~$ more /etc/amandahosts
guadalinex backup
localhost root
```

Con esta configuración del fichero `/etc/amandahosts` se puede iniciar la ejecución del comando `amrecover`:

```
root@guadalinex:~# /usr/sbin/amrecover Daily
AMRECOVER Version 2.4.4p3. Contacting server on localhost ...
220 guadalinex AMANDA index server (2.4.4p3) ready.
200 Access OK
Setting restore date to today (2005-03-25)
200 Working date set to 2005-03-25.
200 Config set to Daily.
501 Host guadalinex is not in your disklist.
Trying host guadalinex ...
501 Host guadalinex is not in your disklist.
Trying host guadalinex.elpiso.es ...
200 Dump host set to guadalinex.midominio.org.
Trying disk / ...
Trying disk rootfs ...
Can't determine disk and mount point from $CWD '/root'
amrecover> setdisk /home/hugo/CICA
501 Disk guadalinex.midominio.org:/home/hugo/CICA is not in your disklist.
amrecover> setdisk /home/hugo/CICA/ed05
200 Disk set to /home/hugo/CICA/ed05.
amrecover> history
200- Dump history for config "Daily" host "guadalinex.midominio.org" disk "/
    home/hugo/CICA/ed05"
201- 2005-03-24 0 HISS14 1
201- 2005-03-24 1 HISS01 1
201- 2005-03-24 1 HISS02 1
200 Dump history for config "Daily" host "guadalinex.midominio.org" disk "/
    home/hugo/CICA/ed05"
amrecover> cd /home/hugo/CICA/ed05/tema5
/home/hugo/CICA/ed05/tema5
amrecover> ls
2005-03-24 .
```

```
2005-03-24 Daily/
2005-03-24 entrega5.lyx
2005-03-24 entrega5.lyx~
2005-03-24 images/
2005-03-24 paquetes_amanda.txt
amrecover> add entrega5.lyx
Added /tema5/entrega5.lyx
amrecover> add paquetes_amanda.txt
Added /tema5/paquetes_amanda.txt
amrecover> list
TAPE HISS14 LEVEL 0 DATE 2005-03-24
        /tema5/paquetes_amanda.txt
        /tema5/entrega5.lyx
amrecover> lpwd
/root
amrecover> lcd /tmp
amrecover> settape file:/backups/tape14
Using tape "file:/backups/tape14" from server localhost.
amrecover> extract
Extracting files using tape drive file:/backups/tape14 on host localhost.
The following tapes are needed: HISS14
Restoring files into directory /tmp
Continue [?/Y/n]? Y
Extracting files using tape drive file:/backups/tape14 on host localhost.
Load tape HISS14 now
Continue [?/Y/n/s/t]? Y
./tema5/entrega5.lyx
tar: ./tema5/entrega5.lyx: implausibly old time stamp 1970-01-01 01:00:00
./tema5/paquetes_amanda.txt
tar: ./tema5/paquetes_amanda.txt: implausibly old time stamp 1970-01-01
    01:00:00
amrecover>
```

Ya se habrían recuperado los archivos `entrega5.lyx` y `paquetes_amanda.txt` en `/tmp/tema5`. No parece que sea difícil ¿verdad? De todas formas se van a repasar más detenidamente los pasos seguidos en la recuperación.

Una vez en la consola interactiva de `amrecover` el primer comando que se utiliza es:

```
amrecover> setdisk /home/hugo/CICA
501 Disk guadalinex.elpiso.es:/home/hugo/CICA is not in your disklist.
amrecover> setdisk /home/hugo/CICA/ed05
```

El comando `setdisk` especifica el disco que vamos a considerar para navegar por los archivos salvados. El error `501` que se produce es debido a que en la copia de seguridad se ha guardado `/home/hugo/CICA/ed05` por lo que no encuentra la ruta `/home/hugo/CICA`. A continuación se muestra el histórico de copias de seguridad para ese directorio:

```
amrecover> history
```

Ya se puede navegar por las imágenes almacenadas con objeto de recuperar los archivos. Los archivos a recuperar se guardan en una lista hasta que demos la orden de recuperar.

```
amrecover> cd /home/hugo/MisDocumentos
amrecover> add entrega1.lyx
amrecover> add mailscanner.lyx
```

Para ver el contenido de la lista que se está creando con los archivos a recuperar:

```
amrecover> list
```

Antes de recuperar los archivos deseados es necesario definir en qué lugar se van a recuperar. Primero se comprueba en qué directorio se recuperan por defecto, para posteriormente cambiarlo a `/tmp`. Los ficheros se recuperarán con los mismos permisos y dueño que tenían originalmente, ésta es la razón por la que hay que ejecutar `amrecover` desde el usuario `root`.

```
amrecover> lpwd
amrecover> lcd /tmp
```

A continuación se define dónde está montada la cinta que contiene las imágenes que se van a utilizar.

```
amrecover> settape file:/backups/tape14
```

Y por último se da la orden de recuperar los archivos.

```
amrecover> extract
```

El error que aparece referente a la fecha es debido a un bug en la utilidad `tar`. Los ficheros recuperados pueden verificarse en la localización que se indicó `/tmp/tema5`.

La utilidad `amrecover` ofrece más opciones de las que se han utilizado en el ejemplo anterior.

```
amrecover> help
valid commands are:
add path1 ...      - add to extraction list (shell wildcards)
addx path1 ...     - add to extraction list (regular expressions)
cd directory       - change cwd on virtual file system (shell wildcards)
cdx directory      - change cwd on virtual file system (regular expressions)
clear              - clear extraction list
delete path1 ...   - delete from extraction list (shell wildcards)
deletex path1 ...  - delete from extraction list (regular expressions)
extract            - extract selected files from tapes
exit
help
history            - show dump history of disk
list [filename]    - show extraction list, optionally writing to file
lcd directory      - change cwd on local file system
ls                 - list directory on virtual file system
lpwd               - show cwd on local file system
mode               - show the method used to extract SMB shares
pwd                - show cwd on virtual file system
quit
listdisk [diskdevice] - list disks
setdate {YYYY-MM-DD|--MM-DD|---DD} - set date of look
setdisk diskname [mountpoint] - select disk on dump host
sethost host       - select dump host
settape [host:][device|default] - select tape server and/or device
setmode smb|tar    - select the method used to extract SMB shares
```

Como puede verse, la potencia de AMANDA a la hora de realizar copias de seguridad viene acompañada de la misma potencia en lo referente a recuperaciones. A pesar de todo no se han descrito todas las posibilidades de AMANDA, limitándose este apartado a las nociones básicas que permiten poner en marcha un sistema de backup/recuperación lo suficientemente estable como para dar tranquilidad al administrador de sistemas.

Capítulo 23

Logs del sistema

"Se debe confiar, pero también verificar"

23.1. Archivos de bitácora

Tan importante como establecer unos mecanismos de seguridad adecuados, es vigilar el sistema. El proceso de vigilar cómo se comporta el sistema se denomina *auditar*.

Los sistemas UNIX en general y Linux en particular, mantienen una serie de archivos de bitácora o logs de sistema que ayudan al administrador del mismo en las funciones de auditoría. Los archivos de log son bloques importantes para construir sistemas seguros, ya que muestran el pasado del sistema así como ayudan a la localización de errores intermitentes o ataques maliciosos.

Sin embargo, los archivos de log tienen un punto negativo muy importante: se encuentran situados en el propio sistema. De esta forma si se pierde el acceso al sistema por error grave, se perderá el acceso a estos archivos, con lo que pierden toda su utilidad.

Para solucionar esto, se pueden llevar los archivos de log a otro sistema, no siendo necesario que tenga la misma potencia que el sistema principal. Este sistema secundario tiene una única función, la de almacenar los archivos de bitácora, por lo que los requerimientos de software y hardware serán mínimos. Es importante también restringir al máximo el acceso a este sistema secundario para evitar que se pierdan los archivos de bitácora, ya sea de forma accidental o provocada.

Otra opción, muy recomendable, es tener en cuenta los archivos de bitácora en las políticas de copia de seguridad. De esta forma podremos recuperar siempre los archivos que hayan sido eliminados y acceder de esta forma a la historia del sistema.

23.2. Archivos de log existentes en el sistema

Por defecto, los archivos de bitácora se encuentran en el directorio `/var/log` del sistema de archivos de Linux. Veamos qué tiene ese directorio en el sistema que estamos utilizando[1]:

```
-rw-r-----   1 root    root       540 2005-03-25 09:14 acpid
drwxrwx---   4 backup  backup    4096 2005-03-24 21:49 amanda
drwxr-xr-x   2 root    root      4096 2005-02-15 18:34 apache
-rw-r-----   1 root    adm      25410 2005-03-25 17:09 auth.log
-rw-rw-r--   1 root    utmp         0 2005-03-20 06:28 btmp
drwxr-xr-x   2 root    root      4096 2005-03-25 16:30 cups
-rw-r-----   1 root    adm     607991 2005-03-25 17:09 debug
-rw-r--r--   1 root    root     10034 2005-03-25 09:14 dmesg
-rw-r--r--   1 root    root       118 2005-01-02 23:29 fontconfig.log
```

[1] Hay que tener en cuenta que este listado puede tener más o menos archivos, dependiendo de las aplicaciones que estén instaladas y que utilicen este directorio para almacenar sus logs.

```
drwxr-xr-x  2 root   root     4096 2005-03-25 09:15 gdm
-rw-r-----  1 root   adm     76839 2005-03-25 09:15 kern.log
-rw-rw-r--  1 root   utmp   584876 2005-03-25 09:17 lastlog
-rw-r-----  1 root   adm       308 2005-03-25 00:40 lpr.log
-rw-r-----  1 root   adm         0 2005-03-20 06:47 mail.err
-rw-r-----  1 root   adm     41539 2005-03-25 17:00 mail.info
-rw-r-----  1 root   adm     42397 2005-03-25 17:00 mail.log
-rw-r-----  1 root   adm         0 2005-03-20 06:47 mail.warn
-rw-r-----  1 root   adm     97629 2005-03-25 17:00 messages
drwxr-xr-x  4 nagios nagios   4096 2005-03-24 00:00 nagios
drwxr-xr-x  2 root   root     4096 2004-05-03 16:27 news
drwxr-xr-x  2 root   root     4096 2005-03-25 09:18 ntpstats
drwxr-x---  2 root   adm      4096 2005-03-20 06:28 samba
-rw-r--r--  1 root   root        0 2005-03-20 06:28 scrollkeeper.log
drwxr-x---  2 proxy  proxy    4096 2005-02-15 14:18 squid
-rw-r-----  1 root   adm    808262 2005-03-25 17:09 syslog
-rw-r-----  1 root   adm     22824 2005-03-25 17:00 user.log
-rw-r--r--  1 root   root        0 2004-09-22 16:26 uucp.log
-rw-rw-r--  1 root   utmp    57600 2005-03-25 09:31 wtmp
-rw-rw-r--  1 root   utmp   108288 2005-03-20 06:07 wtmp.1
-rw-r--r--  1 root   root    61691 2005-03-25 16:21 XFree86.0.log
```

Los subdirectorios que se encuentran dentro de `/var/log` van a almacenar archivos de bitácora específicos de otras aplicaciones. Tal es el caso de los directorios `/var/log/apache` y `/var/log/samba` que almacenarán, respectivamente, los archivos de bitácora del servidor web apache y de la utilidad de compartición de ficheros samba.

A continuación se describe brevemente el objetivo de algunos de los archivos de log que se encuentran en un sistema Linux.

`/var/log/cron.log` Mensajes que aparecen relativos al funcionamiento del `cron`.

`/var/log/daemon.log` Mensajes que aparecen relativos al funcionamiento de los demonios del sistema.

`/var/log/dmesg` Mensajes que aparecen durante el arranque del sistema.

`/var/log/mail.*` Mensajes relativos al demonio de correo, en distintos ficheros según su severidad.

`/var/log/messages` Mensajes genéricos del sistema incluyendo los generados en el arranque.

`/var/log/lastlog` Información de últimos accesos de los distintos usuarios al sistema.

`/var/log/utmp` Información acerca de quiénes están usando el sistema actualmente. Puede haber más usuarios de los que muestre este fichero, ya que no todos los programas utilizan utmp como registro de sesiones.

`/var/log/wtmp` Información acerca de los inicios y finales de sesión

`/var/log/XFree86.0.log` Mensajes relativos a las X.

Como puede verse, se almacena información de cualquier evento que pueda producirse en el sistema, lo cual permite tener una visión hacia atrás en el tiempo en el caso que se produzca un error. Es posible conocer qué pasó antes de producirse el error, lo que ayuda a averiguar las causas del mismo.

La mayoría de los ficheros que se acaban de describir están en formato texto y son visibles desde cualquier editor. Sin embargo, hay uno de los ficheros de log del sistema que requiere del uso de un comando externo para su visualización. El fichero es `/var/log/lastlog` y la utilidad que se necesita para extraer la información contenida en él es `lastlog`. Esta utilidad formatea e

imprime el contenido del fichero `/var/log/lastlog` de forma más legible para el usuario que lo ejecuta.

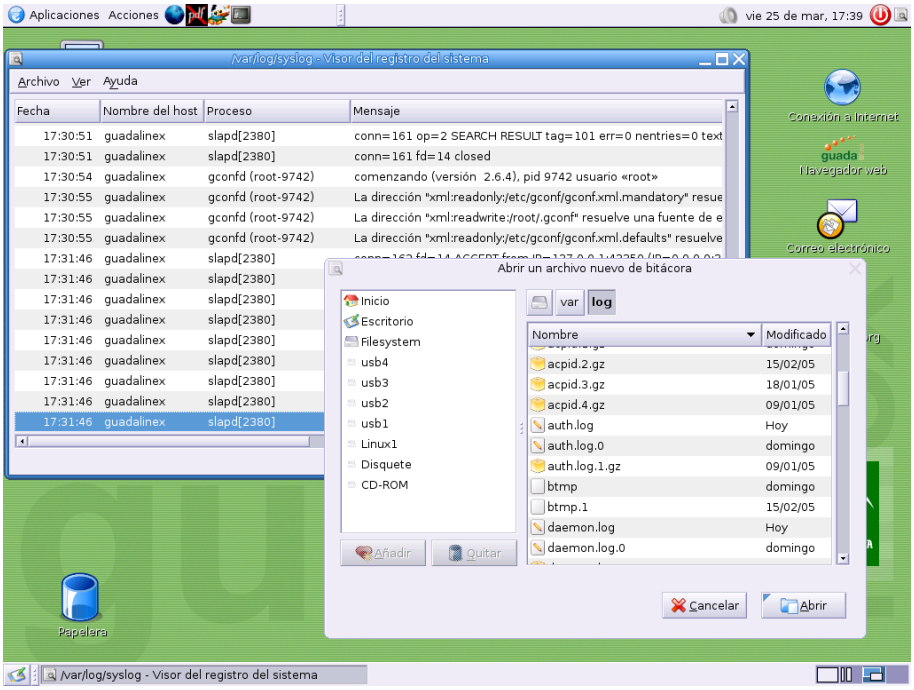
```
root@guadalinex:~# lastlog
Nombre           Puerto    De Ú              ltimo
root             :20 á                      sb mar 19 13:24:44 +0100 2005
daemon                                      **Nunca ha entrado**
bin                                         **Nunca ha entrado**
sys                                         **Nunca ha entrado**
sync                                        **Nunca ha entrado**
games                                       **Nunca ha entrado**
man                                         **Nunca ha entrado**
lp                                          **Nunca ha entrado**
mail                                        **Nunca ha entrado**
...
...
hugo             :0                         vie mar 25 09:17:36 +0100 2005
telnetd                                     **Nunca ha entrado**
legolas          pts/4     192.168.0.13     mar feb 15 21:19:53 +0100 2005
smmta                                       **Nunca ha entrado**
smmsp                                       **Nunca ha entrado**
postfix                                     **Nunca ha entrado**
nagios                                      **Nunca ha entrado**
amanda                                      **Nunca ha entrado**
jose.fernandez   pts/6     guadalinex       dom ene 30 20:09:53 +0100 2005
hugo.santander                              **Nunca ha entrado**
```

Otro comando que también se utiliza para ver los últimos accesos al sistema es `last`. Esta utilidad busca en el fichero `/var/log/wtmp` y muestra una lista de los usuarios conectados al sistema desde que el fichero fue creado.

```
root@guadalinex:~# last
hugo     pts/2        :0.0             Fri Mar 25 09:31   still logged in
hugo     pts/1        :0.0             Fri Mar 25 09:30   still logged in
hugo     pts/0        :0.0             Fri Mar 25 09:30   still logged in
hugo     :0                            Fri Mar 25 09:17   still logged in
reboot   system boot  2.6.5            Fri Mar 25 09:14         (08:15)
hugo     pts/2        :0.0             Thu Mar 24 13:34 - 00:39  (11:05)
hugo     pts/1        :0.0             Thu Mar 24 13:19 - 00:39  (11:20)
hugo     pts/0        :0.0             Thu Mar 24 13:19 - down   (11:20)
hugo     :0                            Thu Mar 24 13:14 - down   (11:25)
reboot   system boot  2.6.5            Thu Mar 24 11:10         (13:30)
hugo     pts/2        :0.0             Wed Mar 23 22:53 - down   (01:10)
hugo     pts/1        :0.0             Wed Mar 23 22:37 - down   (01:25)
hugo     pts/0        :0.0             Wed Mar 23 22:37 - down   (01:26)
hugo     :0                            Wed Mar 23 22:33 - down   (01:29)
reboot   system boot  2.6.5            Wed Mar 23 22:31         (01:32)
hugo     pts/1        :0.0             Sun Mar 20 14:29 - down   (00:19)
hugo     pts/0        :0.0             Sun Mar 20 14:17 - down   (00:30)
hugo     :0                            Sun Mar 20 14:15 - down   (00:33)
reboot   system boot  2.6.5            Sun Mar 20 14:11         (00:36)
wtmp begins Sun Mar 20 07:03:08 2005
```

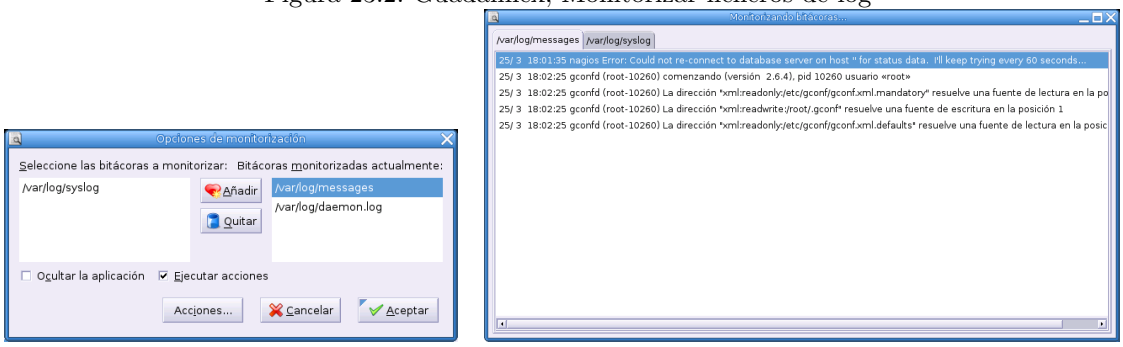
En Guadalinex existe una utilidad gráfica, fácilmente configurable, que visualiza los ficheros de log. Se encuentra en el menú **Aplicaciones→Configuración→Sistema→Bitácora del Sistema**. Se corresponde con la utilidad `gnome-system-log` y es necesario ejecutarla como usuario `root`.

Figura 23.1: Guadalinex, Bitácora del Sistema



Es posible usar esta utilidad para supervisar los archivos de log del sistema previamente seleccionados, usando para ello la opción **Monitor**. Esta opción permite visualizar de forma simultánea varios ficheros de log.

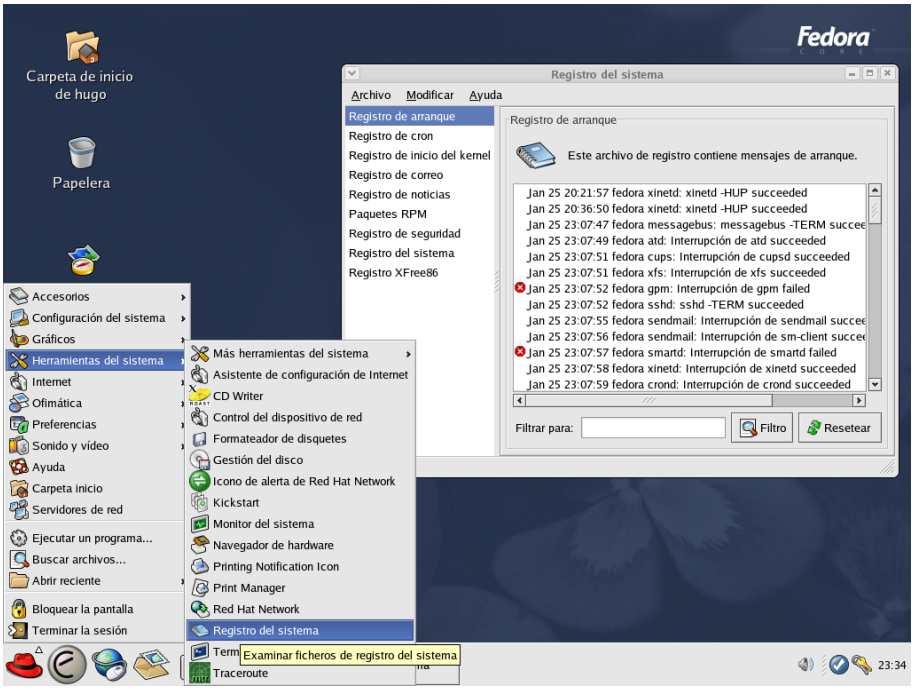
Figura 23.2: Guadalinex, Monitorizar ficheros de log



En el caso de Fedora, para facilitar la visión de algunos de estos ficheros, el sistema proporciona una utilidad gráfica denominada "Registro del Sistema"[2]

[2]Se corresponde con la utilidad Log Viewer versión 0.9.3

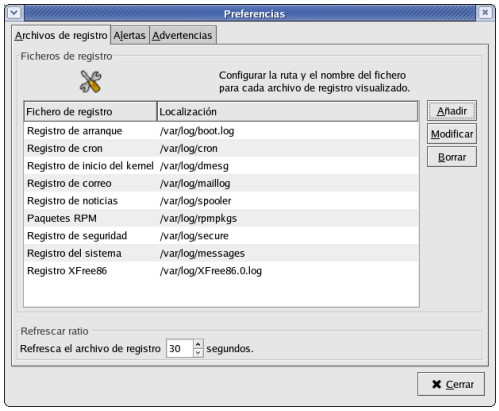
Figura 23.3: Fedora, Registro del sistema



Anteriormente, se vió el listado de algunos de los archivos de log que pueden encontrarse en el sistema. Hay que notar que la mayoría de ellos tenían establecido como dueño y grupo a `root`. Esto significa que únicamente `root` puede ver el contenido de estos ficheros. Lo mismo ocurre con los que pertenecen a otros usuarios, pudiendo visualizar su contenido los propios dueños o `root`. Esta aplicación para ver los archivos de logs se rige también por estos permisos, por lo que será necesario ejecutarla como `root`.

Si se pulsa sobre la opción **Modificar→Preferencias** pueden verse los archivos de log que tiene definidos por defecto:

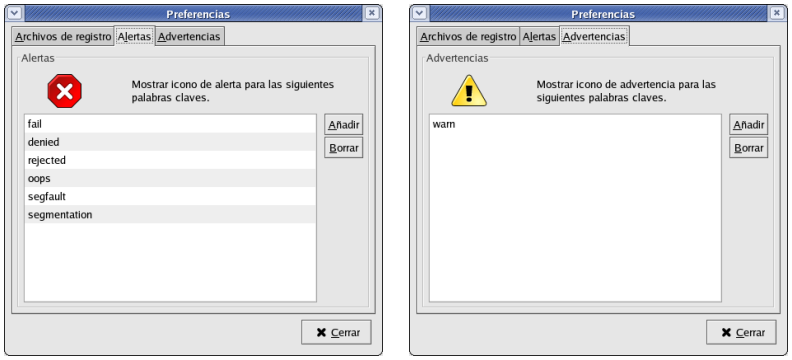
Figura 23.4: Fedora, Archivos de log por defecto



A esta lista se pueden añadir más archivos de log, así como definir en qué circunstancias se va a visualizar un indicador gráfico al lado de las líneas de log, que indicará alguna circunstancia

especial o de error.

Figura 23.5: Fedora, Alarmas y Advertencias

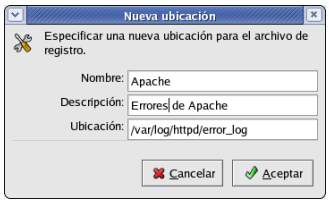


(a) Alarmas (b) Advertencias

En el caso de las alertas, las palabras que se han elegido para indicarnos esta circunstancia son: *fail*, *denied*, *rejected*, *oops*, *default*, *segmentation*. En el caso de las advertencias se tiene *warm*. Esto es configurable a según las necesidades para cada uno de los archivos de log.

La funcionalidad de esta utilidad no acaba aquí, es posible añadir nuevos archivos de log para visualizar.

Figura 23.6: Fedora, Configuración de nuevos archivos de log



En este caso, se ha elegido el archivo de log del servidor web apache que almacena las incidencias del servidor web `/var/log/httpd/error_log`.

23.3. Bitácora del sistema: syslog

La utilidad `syslogd` proporciona soporte al sistema de log de sistema así como del *kernel*. Soporta el almacenamiento de logs tanto de forma local como remota. El soporte para los logs del *kernel* lo proporciona la utilidad `klogd`.

Esta utilidad corre como un servicio que se ejecuta en el inicio del sistema. Es usado por distintas aplicaciones y otros servicios para guardar información sobre los distintos eventos que pueden ocurrir en el sistema. Por ejemplo, cuando el demonio de `cron` está intentando ejecutar un trabajo, manda una petición de "*logging*" a `syslogd`, que a su vez está configurado para enviar la entrada de información al archivo de log correspondiente `/var/log/cron`.

```
Mar 25 20:48:42 guadalinex crontab[4539]: (root) LIST (root)
Mar 25 20:48:46 guadalinex crontab[4540]: (root) BEGIN EDIT (root)
```

```
Mar 25 20:48:52 guadalinex crontab[4540]: (root) REPLACE (root)
Mar 25 20:48:52 guadalinex crontab[4540]: (root) END EDIT (root)
```

Cada de uno de los mensajes que se guardan en los archivos de log contienen al menos un campo con la hora y el nombre del *host*. Dependiendo de las características del programa de log y cómo sea de configurable esta información será más o menos completa, acorde a nuestras necesidades.

`Syslogd` puede mantener múltiples ficheros de log para distintas aplicaciones y servicios. La configuración del demonio `syslogd` se encuentra en `/etc/syslogd.conf`. Es en este fichero donde se le indica a `syslogd` la localización de los archivos de log dependiendo del grado de severidad del mensaje que ha provocado la aplicación de la cual se pretende registrar información.

```
#  /etc/syslog.conf     Configuration file for syslogd.
#
#                       For more information see syslog.conf(5)
#                       manpage.
#
# First some standard logfiles.  Log by facility.
#
auth,authpriv.*                 /var/log/auth.log
*.*;auth,authpriv.none          -/var/log/syslog
cron.*                          /var/log/cron.log
daemon.*                        -/var/log/daemon.log
kern.*                          -/var/log/kern.log
lpr.*                           -/var/log/lpr.log
mail.*                          -/var/log/mail.log
user.*                          -/var/log/user.log
uucp.*                          /var/log/uucp.log
#
# Logging for the mail system.  Split it up so that
# it is easy to write scripts to parse these files.
#
mail.info                       -/var/log/mail.info
mail.warn                       -/var/log/mail.warn
mail.err                        /var/log/mail.err
# Logging for INN news system
#
news.crit                       /var/log/news/news.crit
news.err                        /var/log/news/news.err
news.notice                     -/var/log/news/news.notice
#
# Some 'catch-all' logfiles.
#
*.=debug;\
        auth,authpriv.none;\
        news.none;mail.none     -/var/log/debug
*.=info;*.=notice;*.=warn;\
        auth,authpriv.none;\
        cron,daemon.none;\
        mail,news.none          -/var/log/messages
#
# Emergencies are sent to everybody logged in.
#
*.emerg                         *
#
# I like to have messages displayed on the console, but only on a virtual
# console I usually leave idle.
#
#daemon,mail.*;\
```

```
#        news.=crit;news.=err;news.=notice;\
#        *.=debug;*.=info;\
#        *.=notice;*.=warn        /dev/tty8
# The named pipe /dev/xconsole is for the 'xconsole' utility.  To use it,
# you must invoke 'xconsole' with the '-file' option:
#
#    $ xconsole -file /dev/xconsole [...]
#
# NOTE: adjust the list below, or you'll go crazy if you have a reasonably
#       busy site..
#
daemon.*;mail.*;\
        news.crit;news.err;news.notice;\
        *.=debug;*.=info;\
        *.=notice;*.=warn        |/dev/xconsole
```

Las entradas de este fichero de configuración se encuentran divididas en dos columnas: qué registrar y dónde registrarlo.

23.3.1. ¿Qué podemos registrar en los ficheros de log?

La primera columna tendrá el formato:

```
servicio.severidad
```

La porción `servicio` de la entrada del fichero `syslogd.conf` especificará el tipo de proceso o servicio al que se aplica la norma.

Cuadro 23.1: Tipos de procesos disponibles en `/etc/syslogd.conf`

Tipo	Descripción
`*`	Todos los tipos posibles de procesos
`auth`	Mensajes relacionados con la autorización
`authpriv`	Mensajes relacionados con la autorización privada
`cron`	Mensajes desde los demonios cron y at
`daemon`	Mensajes de los demonios del sistema no especificados en esta tabla
`kern`	Mensajes del núcleo
`local0-local7`	Mensajes enviados a un terminal específico
`lpr`	Mensajes del demonio de impresión
`mail`	Mensajes del servidor de correo
`news`	Mensajes del servidor de noticias
`syslog`	Mensajes del demonio `syslogd`
`user`	Mensajes generales

Con `severidad` se indica qué información quiere guardase, teniendo en cuenta las limitaciones existentes de espacio en disco.

Cuadro 23.2: Niveles de severidad

Nivel	Descripción
`none`	Nada de este tipo de elemento
`emerg`	Mensajes que pueden estar relacionados con un mal funcionamiento o fallo
`alert`	Mensajes de alerta del sistema
`crit`	Mensajes relacionados con cuestiones críticas
`err`	Mensajes de error
`warning`	Mensajes que contienen advertencias
`notice`	Mensajes que contienen noticias que da el programa
`info`	Mensajes de información general
`debug`	Mensajes de depuración
`*`	Todos los niveles para este tipo de elemento

Pueden utilizarse comodines (`*`) tanto en los servicios como en las prioridades para indicar que concuerda con cualquiera. Existe también la partícula `none` como severidad que indica que no se guarde información sobre el evento en particular. Ésta última tiene sentido si se utiliza de forma conjunta con comodines

```
*.info
mail.none
```

Con la configuración anterior se indica que se guarde la información de severidad `info` para todas las aplicaciones excepto para el servicio `mail`.

Otra opción es especificar un subconjunto de servicios separados y una severidad:

```
mail,uucp,news.info
```

También pueden agruparse múltiples reglas juntas para que realicen la misma acción, utilizando el siguiente formato:

```
mail.info;cron.warning
```

Hay que tener en cuenta un aspecto importante en lo referente a la severidad. Cuando se configura una acción para una severidad, se configura para esa severidad y para las que son más altas. Existe sin embargo una funcionalidad que permite decirle al demonio `syslogd` que registre únicamente el nivel de severidad que se puso en la lista[3]. Esta funcionalidad se activa colocando el signo `=` delante de la severidad.

```
mail.=warning
```

En este caso se está indicando que únicamente registre los mensajes con severidad `warning`.

También puede indicarse que no incluya una severidad concreta, colocando el signo `!` delante de ésta.

```
mail.warning; mail.!err
```

Ahora se registrarán los mensajes de severidad `warning` y superiores, excepto los de severidad `err`.

23.3.2. Acciones en respuesta a eventos.

Una vez se ha establecido la información sobre los servicios que se van a registrar, es necesario indicar a syslogd dónde tiene que colocar esa información. En la segunda columna de `/etc/syslog.conf` se indica el destino de la información.

Normalmente lo que aparece aquí es la ruta al archivo en el que se guardarán los mensajes con una determinada severidad que vaya generando el servicio.

En el fichero de configuración mostrado al principio aparece la siguiente entrada:

[3]Es una característica adicional incluida en las distribuciones basadas o con origen en RedHat

```
cron.*                                  /var/log/cron
```

Esta línea indica al sistema operativo que almacene cualquier mensaje proveniente de la aplicación `cron` en el fichero `/var/log/cron`. El `*` se refiere a la severidad del mensaje, optando por almacenar todos los mensajes, independientemente de su severidad, en el fichero `/var/log/cron`. Puede especificar también distintas localizaciones dependiendo de la severidad:

```
mail.info                       -/var/log/mail.info
mail.warn                       -/var/log/mail.warn
mail.err                        /var/log/mail.err
```

En este caso, los mensajes de error serán almacenados en `/var/log/mail.err`, los mensajes de advertencia en `/var/log/mail.warn` y los mensajes con carácter informativo en `/var/log/-mail.info`.

Tras registrar la información que genera un servicio, se sincroniza el archivo donde se están guardando los mensajes. Esto es debido a que la información no siempre se guarda de forma inmediata en el sistema de archivos, sino que ésta se encuentra inicialmente en memoria. Si el trasvase de información es elevado, la constante sincronización puede afectar al rendimiento del sistema. Para anular esta sincronización se añade un signo `-` delante de la ruta.

Dentro de las acciones a realizar con la información que estamos registrando está el enviarla, mediante una tubería, a otro programa o script para un posterior procesamiento. Simplemente tendremos que añadir el símbolo `|` antes de la ruta al programa o script.

```
|/usr/local/bin/procesarlogs.sh
```

Como se indicó al comienzo de esta sección, existe la posibilidad de centralizar los logs del sistema en una máquina remota. El requisito que debe cumplir el servidor remoto es que tenga el demonio `syslogd` ejecutándose.

```
@maquinaremota
```

Por último, la información de log no tiene por qué enviarse a un fichero, puede enviarse directamente a la pantalla o la consola de administración (`/dev/tty3` y `/dev/console` respectivamente). Otra opción es enviar la información de log que se genera por correo a los usuarios que definamos, únicamente tenemos que poner la lista de usuarios separados por comas.

23.4. Gestión de los logs

23.4.1. Registro de nuestros scripts

Acabamos de ver cómo el sistema almacena los eventos que se van produciendo en distintas localizaciones. Así, cualquier fallo o error en el sistema queda reflejado para su posterior estudio. Sería interesante disponer de esta misma funcionalidad en cualquiera de los scripts que creemos para ayudar en la administración del sistema. Conoceríamos en todo momento si la ejecución de los mismos ha sido correcta o no. Para realizar esta función está `logger` (`/usr/bin/logger`), es una interfaz de línea de comando con syslog.

```
logger [-isd] [-f fichero] [-p severidad] [-t tag] [-
u socket] [mensaje ...]
```

Con `logger` podemos escribir los mensajes de nuestros scripts a la localización estándar de los logs, siendo gestionado por el demonio `syslogd`.

Cuadro 23.3: Opciones de la utilidad `logger`

Opción	Descripción
`-i`	Registra el pid del proceso en cada línea
`-s`	Registra el mensaje en la salida de error estándar
`-d`	Utiliza un datagrama en lugar de una conexión stream con el socket
`-f fichero`	Registra el mensaje en el fichero especificado
`-p severidad`	Registra el mensaje con la severidad especificada
`-t tag`	Marca cada línea con la etiqueta específica
`-u socket`	Escribe en un socket
`--`	Finaliza la lista de opciones para poder empezar el mensaje con -
`mensaje`	Escribe el mensaje en el log

La prioridad indicada con la opción `-p` puede ser especificada numéricamente o con la pareja `servicio.severidad`. Si no se especifica el valor por defecto es `user.notice`. Por ejemplo para registrar los mensajes del servicio `local3` con la severidad `info`:

```
logger -p local3.info
```

En caso de que no especifiquemos mensaje y tampoco proporcionemos la opción `-f`, se registrará la entrada estándar.

La salida de `logger` será `0` si hay éxito ó `>0` en caso de error.

Consideraciones previas que hay que tener en cuenta antes de empezar a utilizar `logger`:

- Es necesario que el fichero de log exista antes de enviar un mensaje al mismo a través de `logger`.
- Es necesario crear una entrada en `/etc/syslog.conf` que refleje la existencia del fichero de log que `logger` va a mantener.

23.4.2. Rotación de los logs

Los distintos ficheros de log con la información de registro de aplicaciones y sistema van a almacenar una gran cantidad de datos. Es necesario implementar un mecanismo que permita borrar los datos de registro antiguos así como facilitar la búsqueda de información en los mismos. En el caso de no tener esto en cuenta se podrá comprobar cómo el sistema se llena cada vez más con ficheros de log, los cuales serán cada vez más grandes.

Los sistemas actuales cuentan con la utilidad `logrotate`. Esta utilidad se encarga de realizar la rotación de los ficheros de registro, renombrando el archivo y creando uno nuevo que pasa a ser el fichero de log activo.

Esta utilidad se configura de forma general a través del fichero `/etc/logrotate.conf`. Este fichero suele tener por defecto una configuración similar a la siguiente:

```
# Mirar "man logrotate" para áms detalles
# Rota los ficheros de log de forma semanal
weekly
# Guarda 4 copias de los ficheros de logs
rotate 4
# Crea un fichero nuevo (vacio) de log édespus de rotar los antiguos
create
# Los ficheros de log que se guardan ásern comprimidos
compress
# Los paquetes de las aplicaciones dejan la óinformacin sobre la órotacin de
    logs en este directorio
include /etc/logrotate.d
# éTambin se define íaqu el comportamiento de logs íespecficos
```

```
/var/log/wtmp {
    missingok
    monthly
    create 0664 root utmp
    rotate 1
}
/var/log/btmp {
    missingok
    monthly
    create 0664 root utmp
    rotate 1
}
# La órotacin del resto de logs íespecficos del sistema se define a partir de
    íaqu
```

Contendrá la configuración global para todo el sistema de rotación de logs. Para configurar la rotación de forma particular también tenemos cada uno de los ficheros de `/etc/logrotate.d`.

23.5. Análisis de logs con logwatch

El análisis de los logs de sistema por parte del administrador es un trabajo rutinario y pesado. Son muchos los ficheros que hay que mirar y algunos de ellos pueden generar miles de líneas de información cada día, siendo necesario analizarlos a diario en busca de posibles fallos en el sistema o intentos de acceso no permitidos al mismo.

Una utilidad que puede ayudar en este trabajo es **Logwatch**[4]. Tiene como función principal analizar los archivos de log del sistema, así como crear informes sobre el estado del mismo. Analizará las entradas en los ficheros de log del sistema por el periodo de tiempo definido y realizará un informe sobre determinadas áreas con el nivel de detalle especificado.

```
logwatch [opciones]
```

Las opciones con que puede llamarse a esta utilidad son:

Opción	Acción realizada
`--detail`	Indica el nivel de detalle del informe (`high`, `med` y `low`)
`--logfile`	Fuerza a logwatch a utilizar únicamente el grupo de ficheros de registro indicado. De esta forma sólo se procesará información de los servicios que escriban en estos ficheros de registro
`--service`	Fuerza a logwatch a procesar únicamente los servicios indicados en este parámetro. Procesará todos los ficheros de registro que utiliza este servicio.
`--print`	Muestra el resultado del análisis por la salida estándar.
`--mailto`	Dirección a la que va a enviar el informe de análisis de ficheros de registro.
`--archives`	Esta opción indica a logwatch qué ficheros de registro archivados (normalmente con extensión gz) puede analizar, junto con los ficheros de registro actuales.
`--range`	Rango en el que se va a realizar el análisis (Today, Yesterday y All)
`--debug`	Nivel de debug que se va a aplicar durante el análisis. Únicamente utilizado para detectar errores durante el análisis.
`--save`	Nombre de fichero donde se va a guardar el resultado del análisis de los ficheros de registro.
`--usage` y `--help`	Muestra información de uso de logwatch.

[4]Esta aplicación normalmente está instalada en las distribuciones Fedora. En el caso de Guadalinex es preciso instalarla mediante `apt-get`:

```
apt-get install logwatch
```

Mediante estos parámetros puedes modificarse la configuración durante la ejecución del análisis. Sin embargo, existe también un fichero de configuración `/etc/logwatch/logwatch.conf`[5] que puede establecer el valor de estos parámetros por defecto. Así, cuando se ejecute `logwatch` sin ningún parámetro, tomará los valores establecidos en este fichero. Los parámetros que aparecen en este fichero son:

- Directorio de Log por defecto

  ```
  LogDir = /var/log
  ```

- Directorio temporal por defecto

  ```
  TmpDir = /tmp
  ```

- Persona a la que se le envían los correos con los informes

  ```
  MailTo = root
  ```

- Indica si se envía por correo (NO) o si se muestra por *stdout* (YES)

  ```
  Print = No
  ```

- Para compatibilidad con `mktemp`

  ```
  UseMkTemp = Yes
  ```

- Si está definido, guardará los informes en la ruta indicada en lugar de enviarlo por correo o mostrarlo.

  ```
  Save = /tmp/logwatch
  ```

- Define si se busca en archivos con extension `gz` además de en los ficheros de log actuales.

  ```
  Archives = Yes
  ```

- El rango de tiempo por defecto para el informe (`Today, Yesterday, All`)

  ```
  Range = yesterday
  ```

- El detalle del informe por defecto (`Low = 0, Med = 5, High = 10`)

  ```
  Detail = Low
  ```

- El servicio por defecto para el que se hace el informe. Espera el nombre de un filtro en `/etc/log.d/scripts/services/*` o `All` para todo

  ```
  Service = All
  ```

- Si sólo quisiéramos un informe acerca de ftp

  ```
  Service = ftpd-messages
  Service = ftpd-xferlog
  ```

- Si queremos que únicamente se analice un fichero de log

  ```
  LogFile = messages
  ```

[5] Dependiendo de la distribución de linux el directorio con la configuración de Logwatch puede variar, p.e. `/var/log.d`.

- Localización del programa que envía los correos

  `mailer = /bin/mail`

- Si se establece como Yes se muestran sólo los mensajes referidos al host donde se ejecuta

  `HostLimit = Yes`

La salida generada por `logwatch` para el nivel de detalle por defecto puede ser como la siguiente:

```
 ##################### LogWatch 5.1 (02/03/04) ####################
 Processing Initiated: Fri Mar 25 22:10:02 2005
 Date Range Processed: today
 Detail Level of Output: 5
 Logfiles for Host: guadalinex
 ################################################################
 ------------------------ samba Begin -------------------------
 **Unmatched Entries**
 nmbd/nmbd_nameregister.c:register_name(482) register_name: NetBIOS name
     G2004_1108431407 is too long. Truncating to G2004_110843140 : 6 Time(s)
 ------------------------ samba End ---------------------------
 ---------------------- Disk Space ----------------------
 S.ficheros ñTamao Usado Disp Uso% Montado en
 /dev/hda1 4,0G 2,7G 1,1G 72% /
 tmpfs 93M 0 93M 0% /dev/shm
 ---------------------- Fortune -----------------------
 OpenOffice es potente, Abiword es áms árpido
 ######################## LogWatch End #########################
```

En este caso el informe es reducido, pero dependiendo de la actividad del sistema será más amplio.

Continuando con el resto de ficheros de la utilidad Logwatch, se encuentra la siguiente estructura de directorios que soporta su funcionamiento:

`/etc/logwatch/logwatch.conf` Ya se habló anteriormente de este fichero. Realmente es un enlace simbólico a `/etc/log.d/conf/logwatch.conf`

`/etc/logwatch/conf/services/*` Configuración para los servicios que se van a analizar, así como los ficheros de registro que utilizan

`/etc/logwatch/conf/logfiles/*` Configuración para los ficheros de registro de los servicios a analizar

`/etc/logwatch/scripts/shared/*` Filtros comunes a servicios y/o ficheros de registro

`/etc/logwatch/scripts/logfiles/*` Filtros utilizados para determinados ficheros de registro

`/etc/logwatch/scripts/services/*` Filtros utilizados para los distintos servicios

La herramienta viene configurada para una serie de servicios bastante amplia, pero aún así, es posible ampliar el rango de servicios a analizar. Únicamente debe escribirse el filtro que extraiga la información que se busca de los ficheros de registro correspondientes a una aplicación.

Capítulo 24

Utilidades de administración

> "¿Para quién es Webmin? Webmin es una excelente herramienta tanto para administradores noveles como experimentados."
>
> *System Administation with Webmin*

24.1. Administración remota de sistemas

Webmin es una utilidad de administración de sistemas UNIX vía web, desarrollada por Jamie Cameron y basada en una serie de scripts escritos en Perl. Básicamente, puede decirse que Webmin es un conjunto de CGIs escritos en Perl. Esto le confiere una gran flexibilidad y portabilidad[1], permitiendo esta característica la ampliación con nuevos módulos y funcionalidades.

Webmin utiliza su propio servidor web, escuchando en el puerto que se le indique en la instalación, siendo éste totalmente independiente (si lo tiene) del que se tenga configurado mediante Apache.

24.2. ¿Por qué utilizar Webmin?

Mediante el uso de Webmin, se dispone de una interfaz gráfica fácil de utilizar y que proporciona soporte para un gran número de servicios y labores de mantenimiento del sistema.

Al basarse en una interfaz web, se puede acceder a la totalidad de sus funcionalidades desde prácticamente cualquier sitio de la red, independientemente del sistema operativo. El único requisito es disponer de una conexión a la red donde se encuentra instalado Webmin y de un navegador web.

Por su simplicidad de uso, está indicado tanto para administradores noveles como para los que tienen una experiencia de años. Para los primeros, proporciona una forma visual de acercarse a la administración de sistemas, al mostrar las opciones de los distintos servicios de forma gráfica. En el caso de los administradores con amplia experiencia hay que pensar en la cantidad de opciones definidas para los distintos servicios activos en el sistema, así como los scripts diseñados por ellos mismos para ayudarles en sus labores de administración y monitorización. Mediante el uso de Webmin, pueden crearse llamadas gráficas a estas funcionalidades, evitando así la necesidad de recordar todas y cada una de las opciones o scripts que se ejecutan en un momento dado.

En los siguientes apartados se describirán los pasos necesarios para tener Webmin instalado en un sistema y explicaremos algunas de las opciones más comunes.

[1] Se puede utilizar en más de 35 sistemas UNIX y Linux

24.3. Instalación de Webmin

Una vez que se tiene una ligera idea de lo que Webmin es, se verá de inmediato cómo puede instalarse para empezar a tener otro aspecto, más visual, de la administración de sistemas.

El mejor sitio desde donde bajarse la última versión disponible de Webmin es su web

`http://www.webmin.com/`

que permite bajar el paquete RPM o un tgz que contiene los ficheros de instalación. En el caso de instalar Webmin sobre un sistema Guadalinex puede utilizarse el procedimiento seguido a lo largo del curso, mediante `apt-get`.

```
apt-get install webmin
apt-get install webmin-core
```

La versión que se encuentra disponible en los repositorios de Guadalinex es la 1.130 de 26 de enero de 2004[2]. A pesar de no ser la última versión disponible, esto no presenta ningún incoveniente debido a que Webmin permite la actualización desde la interfaz web. Así, una vez finalizado el proceso de instalación se realizará la actualización para estar a la última versión.

Existe un bug en los paquetes que se instalan en Guadalinex y Debian en general. Está relacionado con la gestión de los temas de webmin y no presenta graves problemas. El bug consiste en que no se puede cambiar de tema, por lo que hay que limitarse a utilizar el tema por defecto para Debian. Este fallo no afecta al resto de funcionalidades de Webmin, que son las que realmente nos ocupan.

En el caso de optar por realizar la instalación a partir del tgz descargado el proceso es el siguiente:

```
root@guadalinex:~# cd /usr/local/
root@guadalinex:/usr/local# tar -zxvf /home/hugo/webmin-1.190.tar.gz
```

Una vez extraídos todos los ficheros del paquete tgz es el momento de comenzar el proceso de instalación.

```
root@guadalinex:/usr/local# cd webmin-1.190
root@guadalinex:/usr/local/webmin-1.190#  ./setup.sh
```

Este script servirá de guía a través del proceso de instalación.

```
*************************************************************************
*           Welcome to the Webmin setup script, version 1.190           *
*************************************************************************
Webmin is a web-based interface that allows Unix-like operating
systems and common Unix services to be easily administered.
Installing Webmin in /usr/local/webmin-1.190 ...
*************************************************************************
Webmin uses separate directories for configuration files and log files.
Unless you want to run multiple versions of Webmin at the same time
you can just accept the defaults.
Config file directory [/etc/webmin]:
Log file directory [/var/webmin]: /var/log/webmin
*************************************************************************
Webmin is written entirely in Perl. Please enter the full path to the
Perl 5 interpreter on your system.
Full path to perl (default /usr/bin/perl):
```

[2]En el momento de escribir estos apuntes la versión más actualizada de Webmin es la 1.190 de 24 de marzo de 2005.

```
Testing Perl ...
Perl seems to be installed ok
******************************************************************
Operating system name:    Debian Linux
Operating system version: 3.1
******************************************************************
Webmin uses its own password protected web server to provide access
to the administration programs. The setup script needs to know :
 - What port to run the web server on. There must not be another
   web server already using this port.
 - The login name required to access the web server.
 - The password required to access the web server.
 - If the webserver should use SSL (if your system supports it).
 - Whether to start webmin at boot time.
Web server port (default 10000):
Login name (default admin):
Login password:
Password again:
Use SSL (y/n): y
Start Webmin at boot time (y/n): y
******************************************************************
Creating web server config files..
..done
Creating access control file..
..done
Inserting path to perl into scripts..
..done
Creating start and stop scripts..
..done
Copying config files..
..done
Configuring Webmin to start at boot time..
Created init script /etc/init.d/webmin
..done
Creating uninstall script /etc/webmin/uninstall.sh ..
..done
Changing ownership and permissions ..
..done
Running postinstall scripts ..
..done
Attempting to start Webmin mini web server..
Starting Webmin server in /usr/local/webmin-1.190
..done
******************************************************************
Webmin has been installed and started successfully. Use your web
browser to go to
 https://guadalinex:10000/
and login with the name and password you entered previously.
Because Webmin uses SSL for encryption only, the certificate
it uses is not signed by one of the recognized CAs such as
Verisign. When you first connect to the Webmin server, your
browser will ask you if you want to accept the certificate
presented, as it does not recognize the CA. Say yes.
```

Ya estaría instalado Webmin en el sistema, para comprobarlo puede ejecutarse:

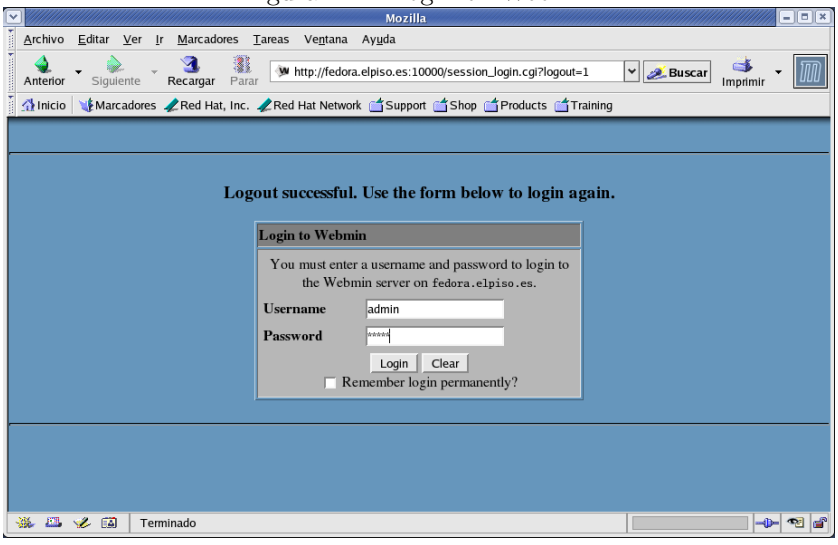
```
root@guadalinex:~# netstat -lp | grep 10000
tcp        0      0 *:10000                 *:*                     LISTEN      5631/perl
udp        0      0 *:10000                 *:*                                 5631/perl
```

Tal como se ha configurado, el puerto 10000 está a la escucha y preparado para recibir peticiones. Al estar Webmin basado en scripts realizados en Perl, será éste el proceso asociado al puerto 10000.

24.4. Primera toma de contacto

Tenemos Webmin recién instalado en nuestro sistema y estamos ansiosos por ver cómo funciona y cómo nos va a permitir administrarlo de una forma más "amigable". Si se escribe en nuestro navegador web `http://nombreservidor:10000/` aparecerá la pantalla de login en Webmin.

Figura 24.1: Login en Webmin

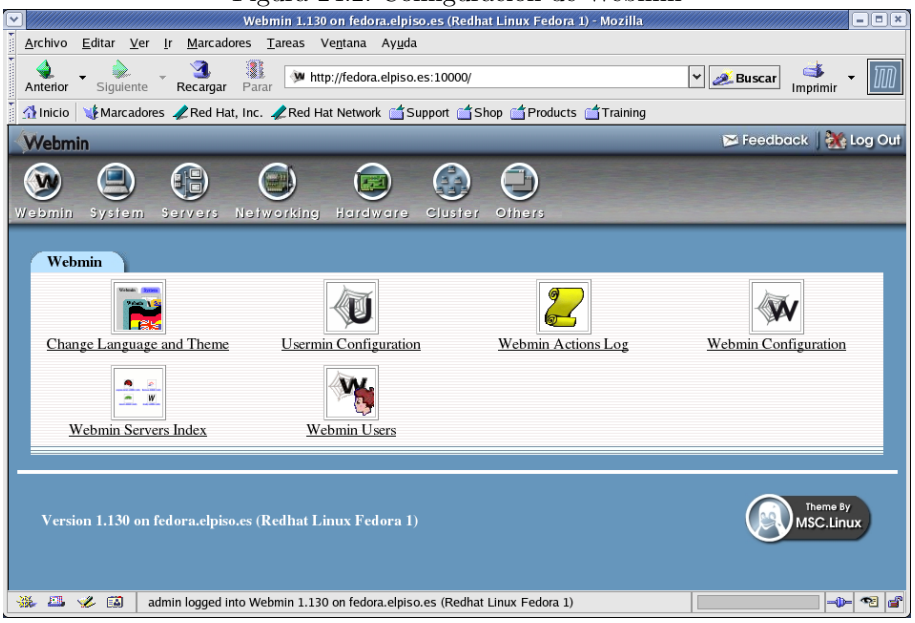


Hasta el momento, únicamente está definido el usuario `admin`[3]. Posteriormente se verá cómo añadir más usuarios, de momento es suficiente con el creado durante la instalación.

Una vez logado con éxito, la pantalla que nos sirve Webmin es la siguiente:

[3]En el caso de la instalación en Guadalinex el usuario es `root`.

Figura 24.2: Configuración de Webmin



Bueno, parece que la primera pantalla no tiene mala pinta ¿verdad? Puede intuirse que la herramienta va a ser bastante completa, sin olvidar que es ampliable con más módulos y scripts existentes en el sistema creados por el administrador.

En esta primera pantalla encontramos las siguientes secciones:

Change Language and Theme. Esta opción permite cambiar el idioma de Webmin.

Configuración Usermin. Permite la configuración del módulo opcional Usermin. Este módulo es una versión simplificada de Webmin diseñada más para usuarios del sistema que para administradores.

Configuración de Webmin. Es el módulo principal de configuración de Webmin. Permitirá actualizar Webmin, gestionar los módulos, gestionar la seguridad y los archivos de log.

Diario de Acción de Webmin. Cuando se habilita el sistema de log, este módulo permitirá búsquedas avanzadas en los logs.

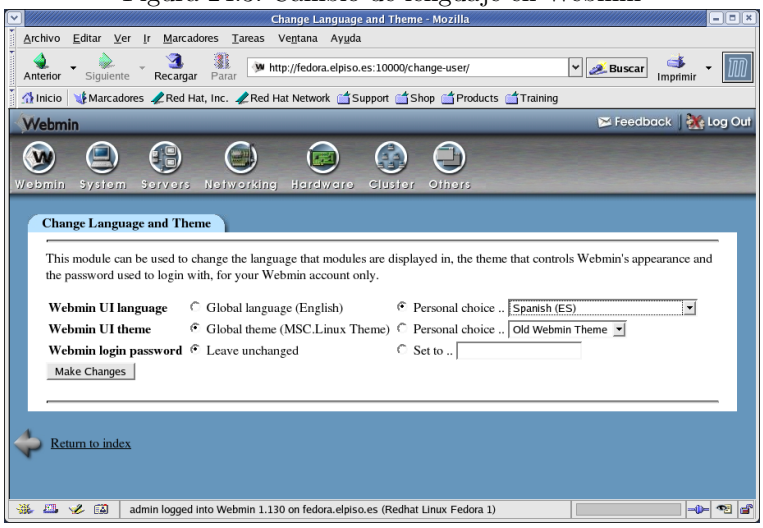
Usuarios de Webmin. Con la configuración de usuarios puede controlarse a qué módulos puede acceder cada usuario e incluso definir grupos para el acceso a módulos.

Índice de Servidores de Webmin. Permite añadir múltiples servidores Webmin existentes en la red para hacerlos accesibles desde una única interfaz.

No esperemos más y vamos a empezar a utilizar Webmin. Lo primero es cambiar el idioma y adaptar Webmin al castellano.

Dentro de esta opción puede cambiarse la palabra de entrada en Webmin del usuario actual (en nuestro caso sigue siendo admin). Aunque se haya cambiado el idioma, puede que algunos nombres de opciones sigan aún en inglés. Webmin es un proyecto que crece día a día y hay algunos aspectos que aún tienen que mejorar.

Figura 24.3: Cambio de lenguaje en Webmin



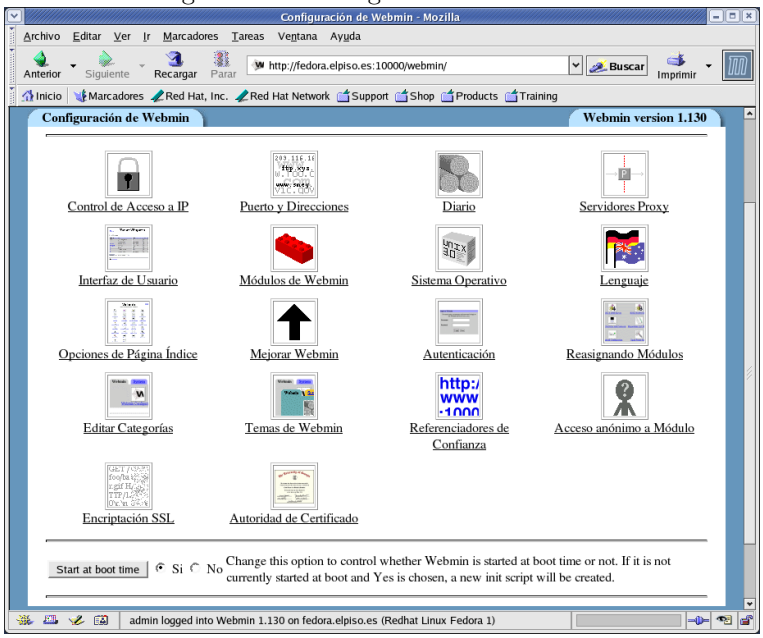
Otro módulo que aparece en esta pantalla inicial es Diarios de Accion de Webmin. Desde aquí se accede a los logs que vayan generándose por el uso de la herramienta, permitiendo una posterior monitorización del sistema. Esta funcionalidad es también configurable como se verá en la siguiente sección.

Posteriormente se volverá al resto de opciones referentes a usuarios y al índice de servidores Webmin. Ahora nos centraremos en seguir conociendo la herramienta.

24.5. Administración de Webmin

24.5.1. Configuración de Webmin

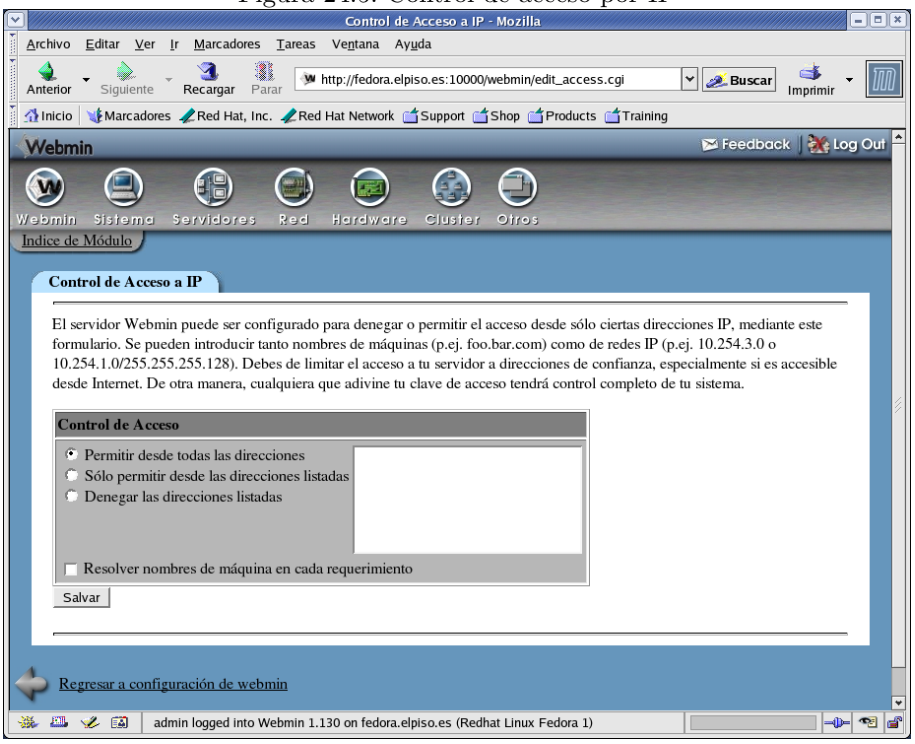
Figura 24.4: Configuración de Webmin



Este módulo permite configurar los aspectos más importantes de Webmin, así como instalar o actualizar nuevos módulos o incluso actualizar la propia herramienta.

Uno de los aspectos que no hay que olvidar es la seguridad en el acceso a Webmin. Es posible configurar esta herramienta para que se acceda únicamente desde las direcciones IP que se le indique. El módulo que controla este aspecto es Control de Acceso a IP.

Figura 24.5: Control de acceso por IP

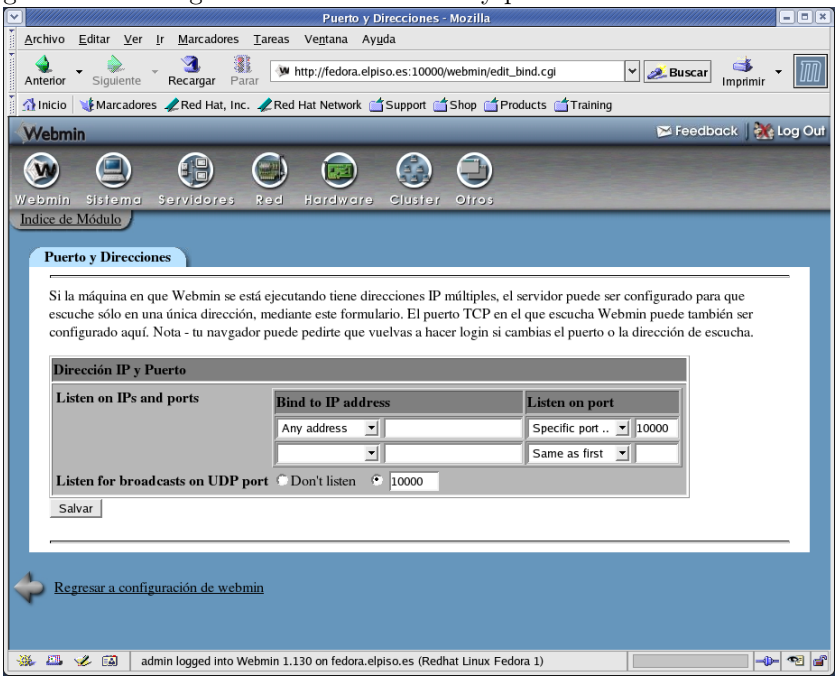


Por defecto está configurado para permitir el acceso desde cualquier dirección IP. Se indicará si se permite o niega el acceso desde las direcciones o redes que se especifiquen en la configuración de este módulo[4].

Otro módulo que aparece en esta sección es el relativo a la dirección IP y el puerto en el que estará a la escucha de peticiones. Este módulo es Puerto y Direcciones y, por defecto, configura Webmin a la escucha en todas las direcciones IP que tenga configuradas el sistema. En caso de tener varias interfaces de red con acceso a distintas redes, sería recomendable, por motivos de seguridad, configurar Webmin para que únicamente esté escuchando en la dirección IP que pertenezca a la red más segura.

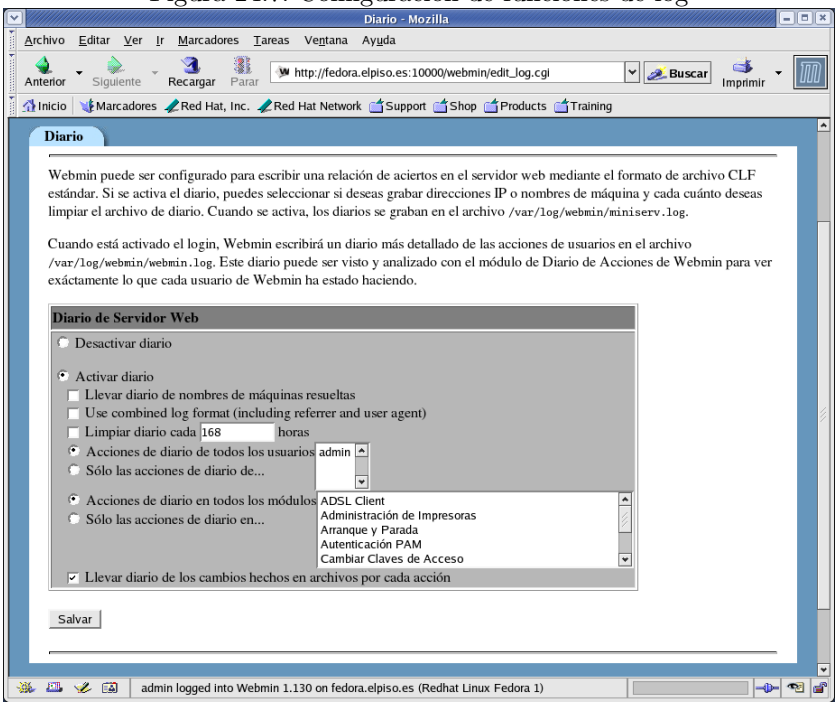
[4]En caso de que alguien obtenga alguna de las claves de acceso a Webmin, con este módulo se establece un nivel más de seguridad.

Figura 24.6: Configuración de dirección IP y puerto donde escucha Webmin



Como ya se dijo, es posible cambiar desde este módulo el puerto donde escucha Webmin. El valor que tiene establecido por defecto es el 10000, aunque puede aparecer otro, si así se lo hemos indicado durante el proceso de instalación.

Figura 24.7: Configuración de funciones de log



Otro módulo interesante y que ayudará a controlar el uso que se hace de Webmin es DIARIO. Hay que recordar que Webmin proporciona funcionalidades de log. Esto permite monitorizar fácil-

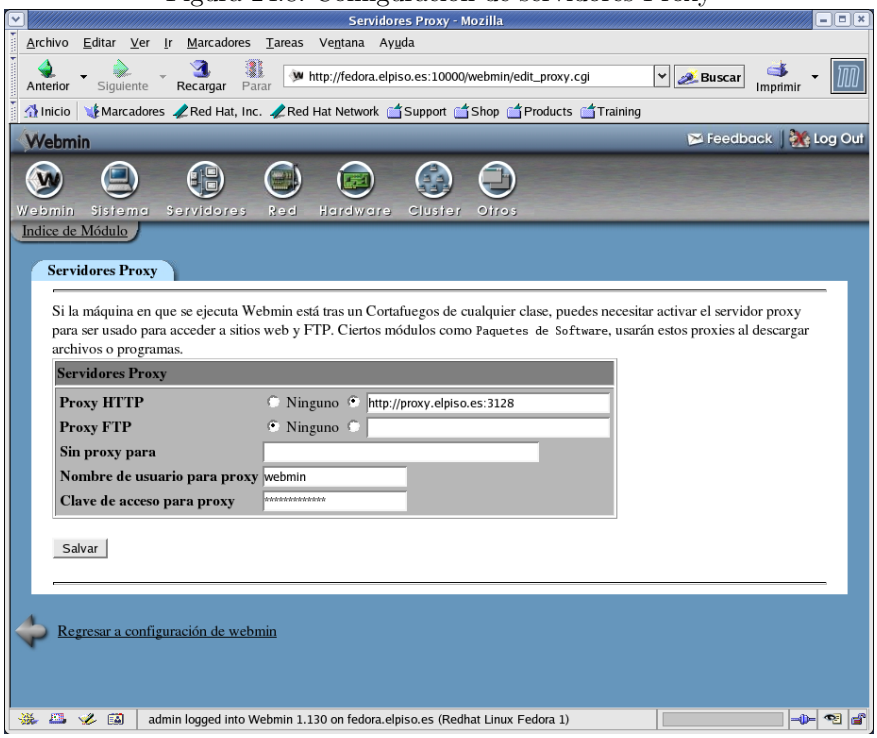
mente las acciones que realizan los distintos usuarios de Webmin, conociendo el momento concreto en que se ha llamado a un módulo y qué acción se ha realizado en el mismo.

Como puede verse, es posible tener un registro detallado de las acciones realizadas basado en el módulo donde se realizaron las acciones. La opción "**Llevar diario de nombres de máquinas resueltas**" hará que Webmin muestre el nombre de la máquina que se ha conectado en lugar de su dirección IP. También puede vaciar los ficheros de log en el intervalo que se establezca, efectuándose una rotación de los mismos.

Para evitar que el disco se llene de forma innecesaria, es conveniente realizar un estudio previo de la situación del sistema y establecer los módulos que es preciso monitorizar y excluir el resto de los logs de Webmin.

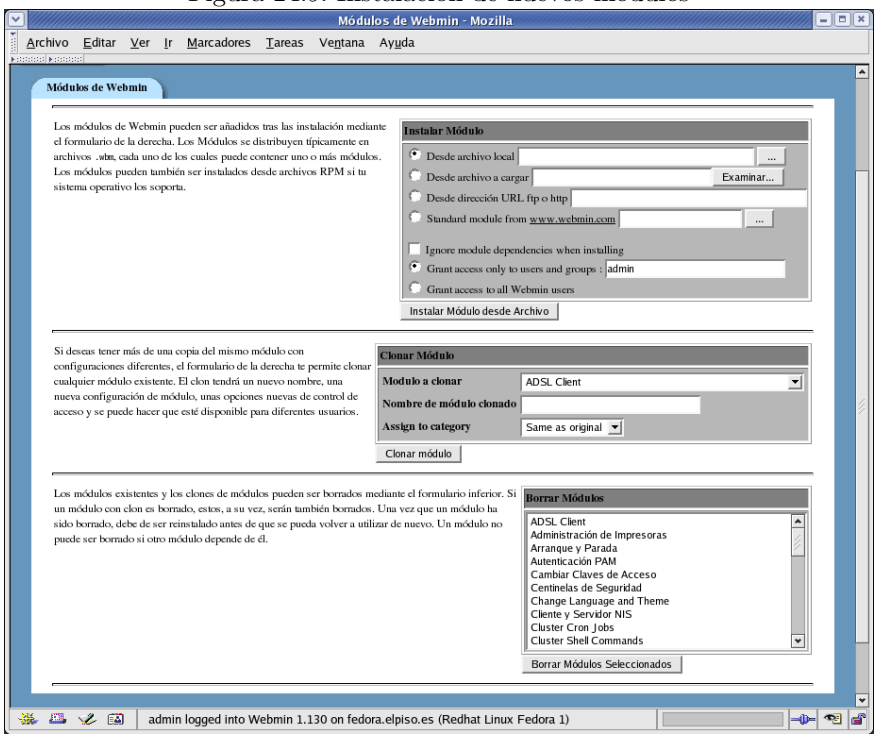
Puede que el servidor donde se encuentra Webmin alojado esté detrás de un cortafuegos o utilice un proxy para salir a internet. Será necesario configurar Webmin para permitirle la conexión a internet en el caso que requiera bajarse alguna actualización o algún módulo adicional, indicándole un proxy a través del cual pueda descargarse los módulos o actualizaciones necesarias.

Figura 24.8: Configuración de servidores Proxy



Una de las principales características de Webmin es su diseño modular. Así, es posible crear nuevos módulos que se integren por completo en la estructura de esta herramienta. Desde MÓDULOS DE WEBMIN es posible instalar nuevos módulos, ya sea desde un fichero local o desde una localización de otro servidor en internet. Los módulos de Webmin son paquetes tar que contienen la estructura completa del módulo. Estos módulos tienen la extensión .wbm.

Figura 24.9: Instalación de nuevos módulos



Pueden encontrarse bastantes módulos para Webmin hechos por otras personas en
`http://webmin.thirdpartymodules.com/`

La mayoría de ellos son gratuitos y están organizados por categorías para una búsqueda más eficiente.

Dentro de la administración de módulos existe la posibilidad de clonar módulos. Esto es especialmente útil para el caso en que sea necesario disponer de varios módulos de gestión de un servicio del sistema y que se ejecuten con distintas configuraciones. Los ficheros de configuración de los servicios debemos gestionarlos nosotros, haciendo copias de los ficheros de configuración para que cada una de las instancias del módulo utilice una configuración distinta.

Pueden también borrarse los módulos que no vayan a utilizarse o que no quieran gestionarse desde Webmin. Hay que tener en cuenta que esto borrará el módulo completamente del sistema, siendo necesario bajarlo de nuevo e instalarlo si posteriormente lo queremos utilizar. Una opción mejor es quitarlo del perfil de los usuarios que no queremos que lo utilicen, tal como veremos en la gestión de los usuarios.

Además de la instalación de nuevos módulos, la actualización de Webmin se realizará desde Mejorar Webmin. La actualización se refiere tanto a la herramienta Webmin como a los módulos que trae en la distribución estándar. Realizando regularmente este proceso de actualización, mantendremos Webmin en un estado óptimo, corrigiendo posibles errores o agujeros de seguridad que se detecten.

Esta visión general de algunos de los módulos de configuración de Webmin debe ser suficiente para empezar a utilizarlo. Hay más módulos que es recomendable investigar y ver las funcionalidades que presentan en lo referente a personalizar Webmin.

Gestión de usuarios

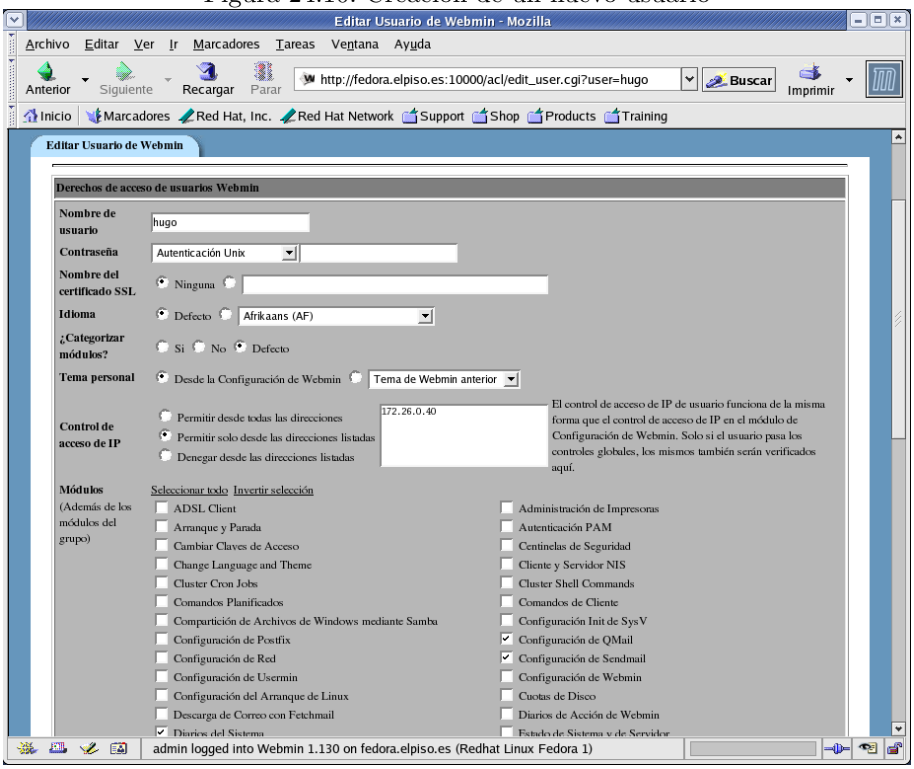
La gestión de usuarios de Webmin permite tener varios perfiles de administradores de sistemas. Por ejemplo, se puede disponer de un usuario en Webmin que se encargue de gestionar el servidor de correo, otro para gestionar el servidor web, otro para gestionar el servidor DNS. Así

el usuario que encargado de gestionar el correo no podrá interferir en las operaciones que realicen los administradores de web y DNS ya que no tendrá permisos para utilizar otros módulos que no sean los relacionados con el correo y para los que tiene acceso.

Dentro de la categoría Webmin entraremos en el módulo Usuarios de Webmin. El único usuario que existe es el usuario admin, con el que nos hemos validado en el sistema. Este usuario tiene acceso a todos los módulos instalados en el sistema.

Crearemos un nuevo usuario que se encargue de la administración del correo.

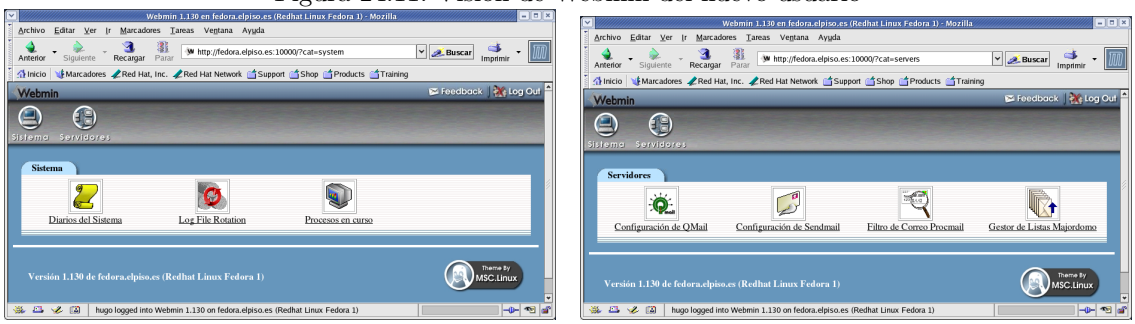
Figura 24.10: Creación de un nuevo usuario



A este usuario se le han asignado permisos para utilizar únicamente un grupo reducido de módulos. Es interesante también comprobar que en el apartado **Contraseña** hemos indicado que se utilice el método de autenticación Unix. Webmin puede gestionar los usuarios de forma independiente del sistema o basarse en los que hay creados. Cada una de estas opciones tiene sus ventajas y sus inconvenientes. La gestión de usuarios Unix es especialmente indicada para el caso que el usuario que va a utilizar Webmin también tenga definido un usuario en el sistema. Si esta circunstancia no se produce es cómoda la gestión interna por parte de Webmin de los usuarios. De esta forma se tiene la certeza que un usuario que tenga acceso a Webmin para labores de monitorización no tendrá acceso directo al sistema. Se logra así aislar a este usuario.

Tras este breve paréntesis en el que se ha visto cómo gestionar el sistema de usuarios, continuemos con la creación del usuario. Si pulsamos sobre **Logout** en la esquina superior derecha, saldremos del sistema y podremos entrar de nuevo con el usuario que acabamos de crear.

Figura 24.11: Visión de Webmin del nuevo usuario



La visión obtenida ahora con este nuevo usuario no tiene nada que ver con la anterior, desde el punto de vista del usuario admin.

Pueden crearse tantos usuarios como sea necesario y asignarles los módulos que se estime necesarios para su labor de administración.

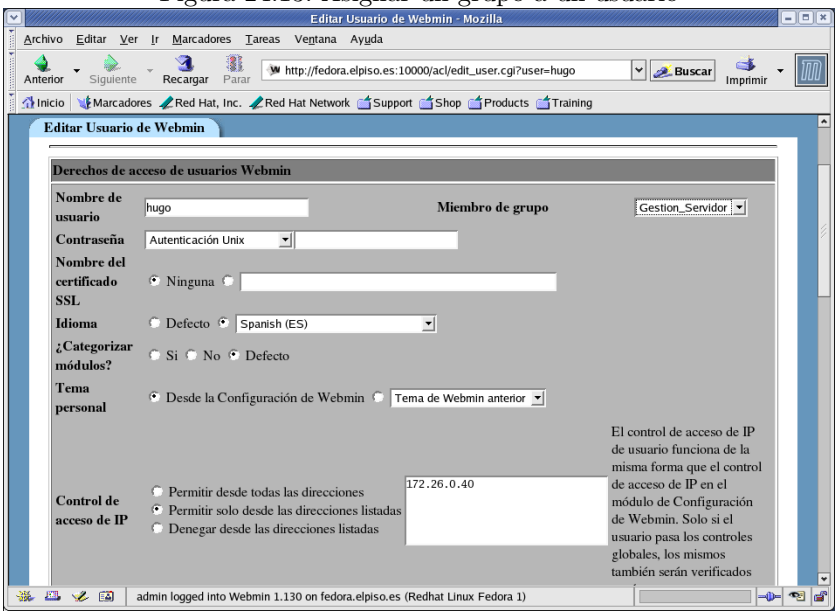
Al igual que ocurre en Linux, Webmin entiende el concepto de grupos. Los grupos en Webmin son similares a los de Linux y permiten simplificar la administración. Pueden crearse usuarios que tengan los mismos permisos y control de acceso. Cuando se crea un grupo en Webmin, se le asigna un nombre y se seleccionan los módulos a los que los miembros de este grupo tendrán acceso. Una vez creado el grupo, cualquier nuevo usuario podrá asignarse al grupo y automáticamente recibirá el acceso a los módulos del grupo, junto a los módulos a los que tenga acceso el usuario. En la implementación actual, un usuario únicamente puede pertenecer a un grupo, a diferencia de los grupos de Linux, en los que un usuario puede tener un grupo primario y varios grupos secundarios.

Figura 24.12: Grupos en Webmin

Hemos creado un grupo con acceso a todos los módulos relacionados con la administración general del sistema. De esta forma, cada vez que se cree un usuario para administrar un servicio concreto, es posible asignarle este grupo de administración general del servidor. Éste sería un ejemplo sencillo de uso de grupos en Webmin.

Es posible editar ahora el usuario recién creado anteriormente y asignarle el grupo que se acaba de crear.

Figura 24.13: Asignar un grupo a un usuario



24.6. Un ejemplo de configuración de servicio: Apache

24.6.1. Dónde configurar los servicios de nuestro sistema

Pulsando sobre la categoría SERVIDORES, se accede a una de las funcionalidades más interesantes de Webmin. Dentro de la categoría Servidores se encuentran los módulos encargados de la configuración de la mayoría de los servidores más utilizados.

El que un módulo esté instalado no quiere decir que el servicio que gestiona ese módulo esté instalado. Únicamente informa que existe soporte en Webmin para la configuración de un servicio. Es responsabilidad del administrador el tener instalado el servicio en el sistema.

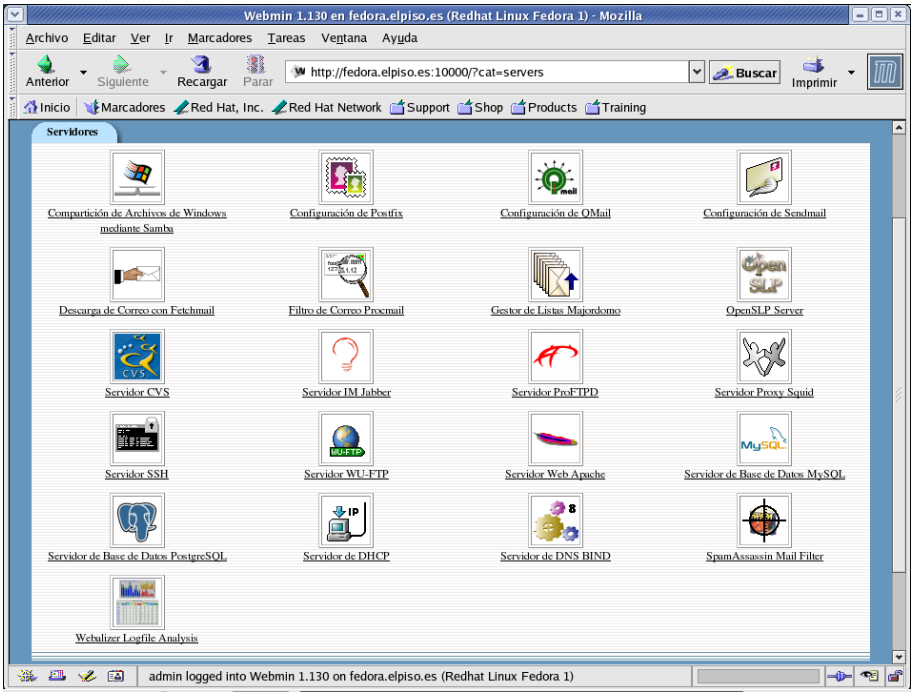
Una de las causas de la popularidad de Webmin es la posibilidad para el administrador de realizar algunas tareas de administración a través de Webmin, sin estar forzado a realizarlas todas. Además:

- a diferencia de otras herramientas gráficas de configuración para sistemas Linux, Webmin intenta dejar intacto el fichero editado.

- el hecho de configurar un servicio desde Webmin no quita que puedan editarse a mano los ficheros de configuración correspondientes.

Podrá utilizarse uno u otro método indistintamente, ya que una de las virtudes de Webmin es que no se dañarán las configuraciones hechas manualmente. Los comentarios y el orden de las directivas no son modificados por lo que no se producen conflictos entre ambos métodos.

Figura 24.14: Categoría Servidores



A continuación se verá uno de los módulos de configuración más utilizados. En concreto se verá qué funciones proporciona Webmin para configurar un servidor web. Pulsando sobre el icono de Apache se van editar los ficheros de configuración de Apache, normalmente[5] localizados en `/etc/httpd/conf`. La mayoría de los módulos de esta categoría pueden editar los ficheros de configuración bajo el directorio `/etc`.

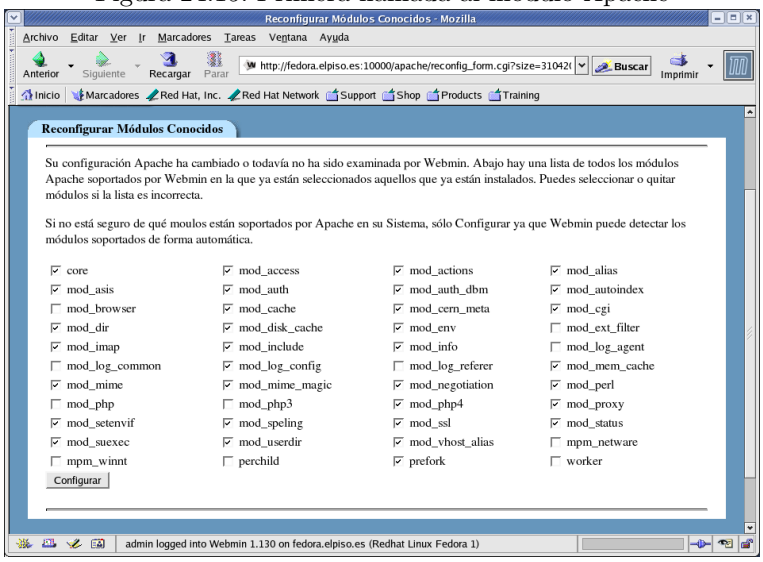
24.6.2. Módulo de configuración de Apache

El servicio que proporciona Apache es uno de los primeros que quiere verse en funcionamiento al instalar Linux. El módulo que proporciona Webmin es bastante completo y permite configurar Apache de la misma manera que manualmente, editando el fichero de configuración.

La primera vez que se ejecuta este módulo, Webmin detectará que anteriormente no se había ejecutado y preguntará por los módulos de Apache que hay instalados en el sistema. Al realizar una detección automática, en caso de duda, es recomendable aceptar la lista de módulos propuesta.

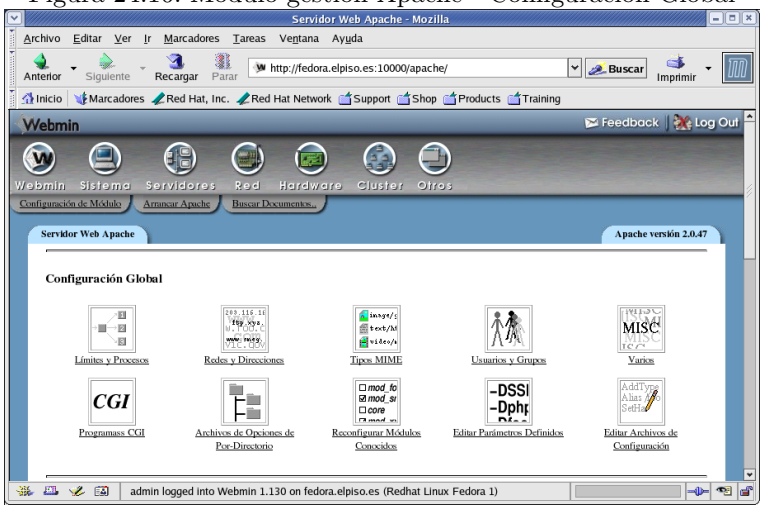
[5] En Debian: `/etc/apache2`

Figura 24.15: Primera llamada al módulo Apache



El módulo de gestión de Apache está dividido en varias secciones que cubren distintos aspectos de la configuración. En la página principal, estas secciones se agrupan en **Configuración Global** y **Servidores Virtuales**.

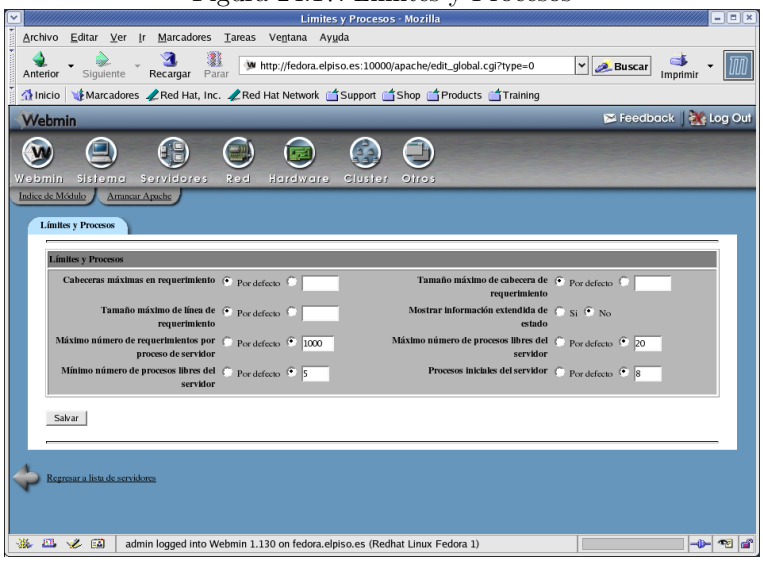
Figura 24.16: Módulo gestión Apache - Configuración Global



La **Configuración Global** proporciona acceso a las opciones que serán compartidas por todos los servidores virtuales que haya definidos. Las opciones configuradas aquí se aplicarán a todos los servidores virtuales y al servidor por defecto. Para configurar una opción para un servidor virtual específico, deberá configurarse en la sección **Servidores Virtuales**.

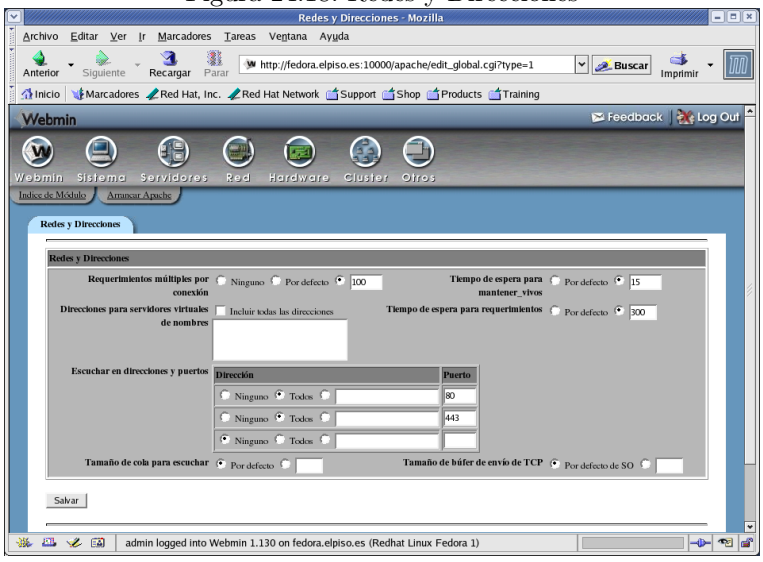
Límites y Procesos. Este módulo permite configurar algunos de los límites establecidos por defecto en Apache. El primer conjunto de límites es el relacionado con la longitud de las *request-headers* que serán aceptadas por el servidor. El segundo conjunto hace referencia a los límites de Apache en lo referente a conexiones y procesos.

Figura 24.17: Límites y Procesos



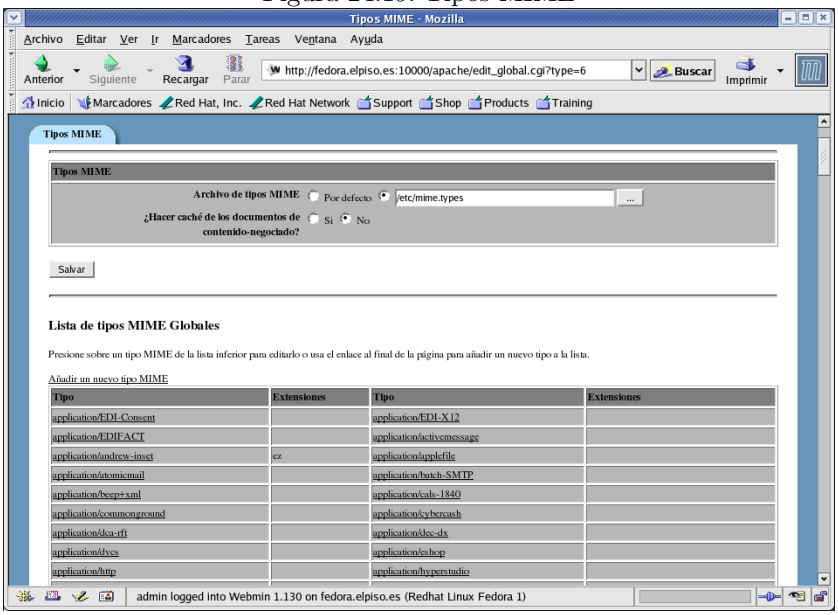
Redes y Direcciones. En este caso los parámetros disponibles son los relativos a las redes y puertos en los que puede configurarse Apache, así como algunos relativos a las conexiones.

Figura 24.18: Redes y Direcciones



Tipos MIME. Los tipos MIME proporcionan un método por el que el servidor y sus clientes conocerán el tipo de dato de un objeto determinado. Este módulo proporciona un mecanismo mediante el que podemos añadir nuevos tipos MIME a los ya definidos en Apache.

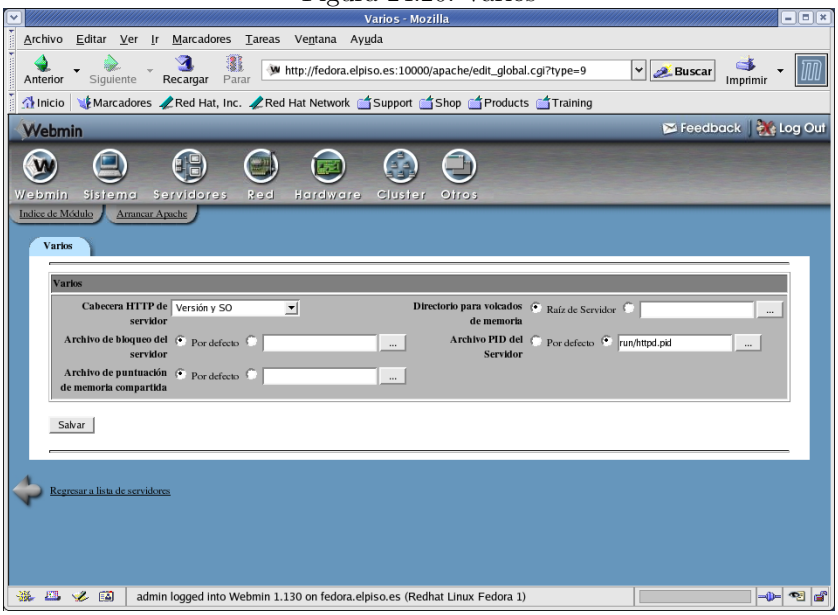
Figura 24.19: Tipos MIME



Usuarios y Grupos. Para cambiar el usuario y grupo bajo el cual se ejecutará Apache.

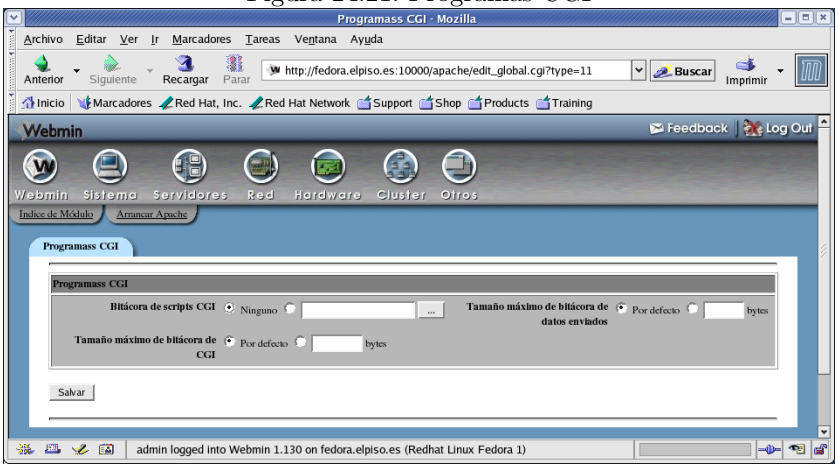
Varios. Como indica el nombre del módulo, aquí se configuran aspectos de Apache que no han sido incluidos en otros módulos.

Figura 24.20: Varios



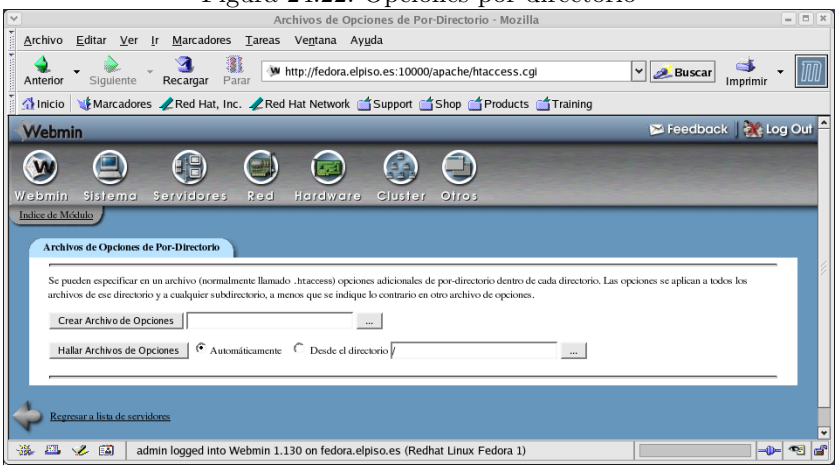
Programas CGI. Este módulo es un interfaz a las opciones de configuración global de los programas CGI en Apache.

Figura 24.21: Programas CGI



Archivo de Opciones de Por-Directorio. Las opciones adicionales de configuración para directorios específicos de la ruta del servidor web pueden realizarse mediante este módulo.

Figura 24.22: Opciones por-directorio

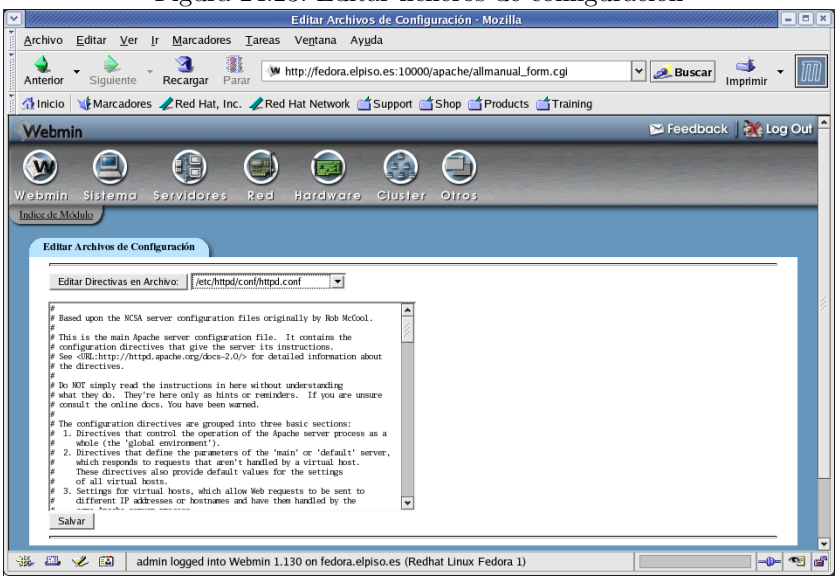


Reconfigurar Módulos Conocidos. Este módulo permite seleccionar los módulos que Apache va a arrancar en el servidor web. Este módulo se llama la primera vez que se accede a la configuración de Apache en Webmin.

Editar Parámetros Definidos. Mediante este módulo se cambia los parámetros con los que se arranca Apache.

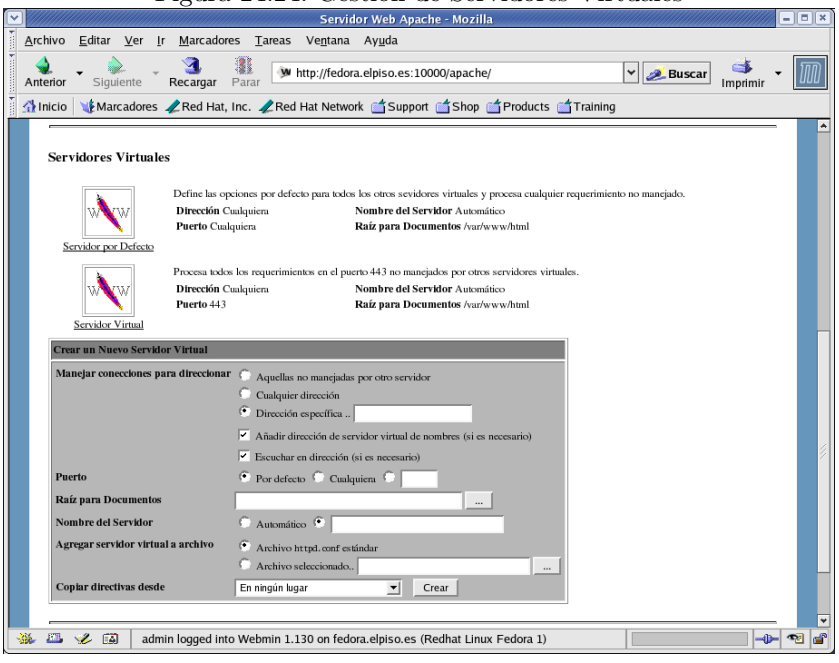
Editar Ficheros de Configuración. Si, a pesar de los módulos existentes, quiere editarse manualmente el fichero de configuración de Apache, también es posible hacerlo desde Webmin.

Figura 24.23: Editar ficheros de configuración



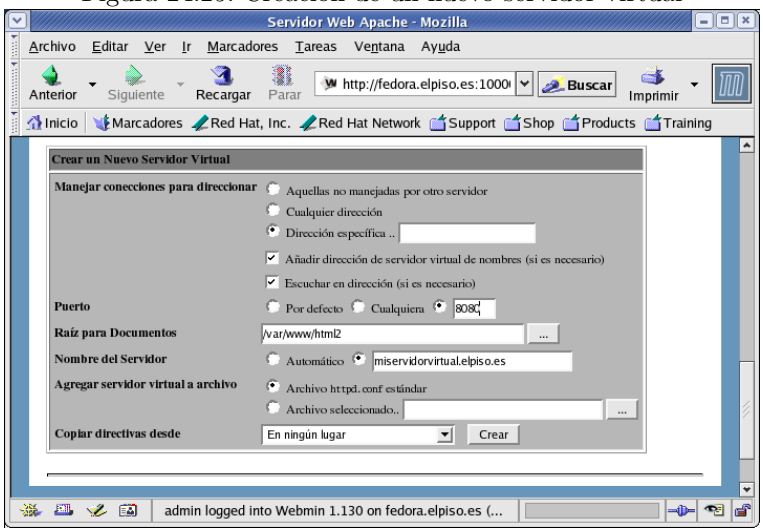
Gestión de servidores virtuales

Figura 24.24: Gestión de Servidores Virtuales



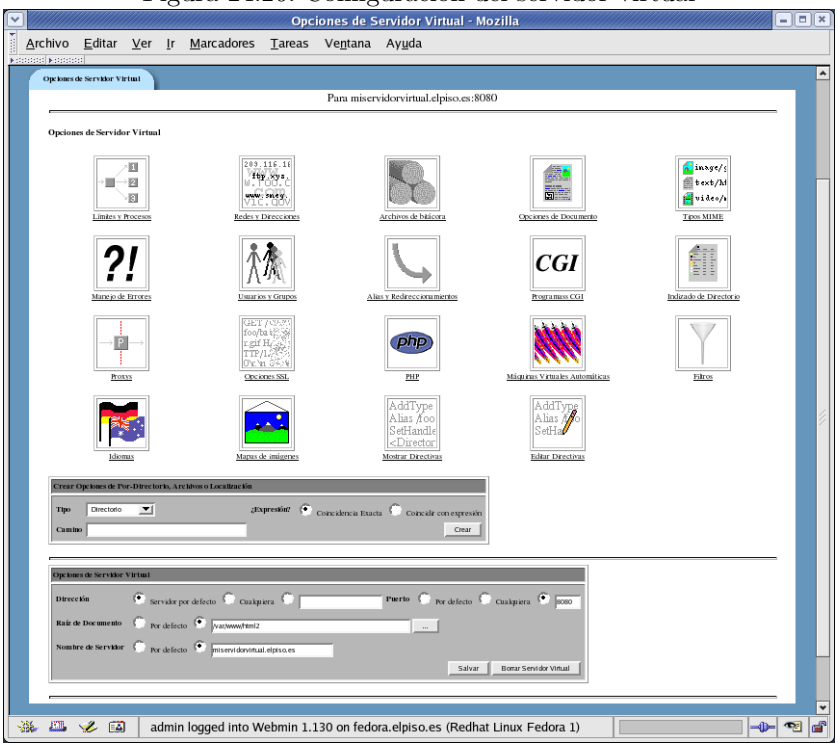
En el apartado referente a Apache ya se comentó algo al respecto de los servidores virtuales. Entremos en materia y veamos qué tendríamos que hacer para crear un nuevo servidor virtual. Supongamos que el nombre del nuevo servidor virtual será `miservidorvirtual.midominio.org` y estará a la escucha por el puerto 8080.

Figura 24.25: Creación de un nuevo servidor virtual



Con esta sencilla operación desde Webmin ya estaría configurado un nuevo servidor virtual. El siguiente paso sería personalizar la configuración del mismo. Para ello únicamente hay que pulsar sobre el icono correspondiente a `miservidorvirtual.midominio.org` que aparece en la pantalla principal del módulo Apache.

Figura 24.26: Configuración del servidor virtual



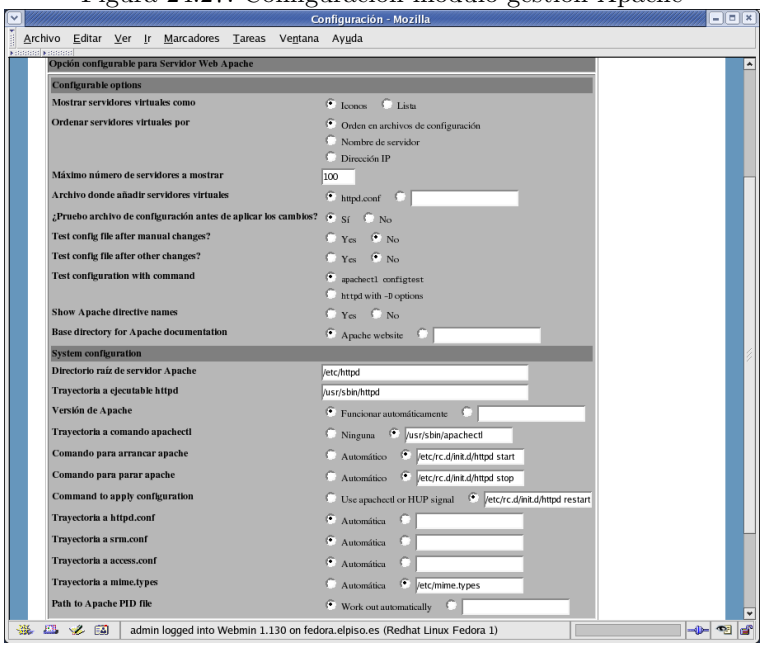
24.6.3. Consideraciones finales

La configuración desde Webmin de Apache puede comprobarse que es bastante sencilla, pero hay que tener en cuenta que son necesarios conocimientos previos sobre el servicio que se desea

configurar.

Hasta ahora se ha tratado de la configuración del servicio que proporciona Apache, pero es posible configurar este módulo de Webmin para modificar su comportamiento.

Figura 24.27: Configuración módulo gestión Apache



También es útil en este caso el uso de la funcionalidad tratada anteriormente **Clonar Módulos**. Normalmente no es necesario ejecutar más de un demonio `httpd` en el mismo servidor, pero en el caso que fuera necesario podemos utilizar la funcionalidad de clonar módulos.

24.7. Gestión de varios servidores Webmin

Dentro de la categoría Webmin existe un módulo denominado **Indice de Servidores Webmin**. Esta página nos da acceso a la lista de servidores Webmin existentes en la red local donde se encuentra instalado. Pulsando sobre el icono de cualquiera de los servidores Webmin que se defina se accede a la pantalla de login de dicho servidor Webmin.

Para añadir nuevos servidores Webmin de forma automática se habilitan los siguientes procedimientos:

Retransmisión para servidores. Hace que Webmin envíe una petición de broadcast al puerto 10000 de nuestra red local. Cualquier servidor Webmin existente en nuestra red responderá identificándose a sí mismo. De esta forma Webmin lo añade a la lista de servidores.

Explorar por servidores. También podemos especificar la red en la que queremos realizar la búsqueda de servidores Webmin.

Existe también la opción de añadir los servidores Webmin de forma manual.

Capítulo 25

Monitorización de Sistemas

25.1. Nagios

25.1.1. ¿Qué es Nagios?

Nagios es un sistema de monitorización de redes y servidores. Chequea de forma periódica los nodos y los servicios que se especifiquen a través de la red, alertando cuando se superan los indicadores definidos y cuando se vuelve de nuevo a una situación estable. Su gran versatilidad permite a Nagios monitorizar prácticamente cualquier cosa que esté en la red.

Originalmente fue diseñado para ejecutarse bajo sistemas Linux, aunque en la actualidad es posible su instalación en otros sistemas. Algunas de las características que incluye Nagios son:

- Monitorización de servicios (SMTP, POP3, HTTP, NNTP, PING, etc.).
- Monitorización de recursos del nodo (carga de procesador, uso de disco, etc).
- Posibilidad de creación de plugin personalizados que permiten el chequeo de servicios o parámetros no contemplados.
- Chequeo de los servicios de forma paralelizada.
- Notificaciones a los responsables cuando cambia el estado del sistema.
- Posibilidad de definir controles proactivos como respuesta a un estado.
- Rotado de log de forma automática.
- Interfaz web para visualizar el estado actual de los controles definidos.

25.1.2. Instalación de Nagios

Guadalinex

```
apt-get install nagios-text
```

Fedora

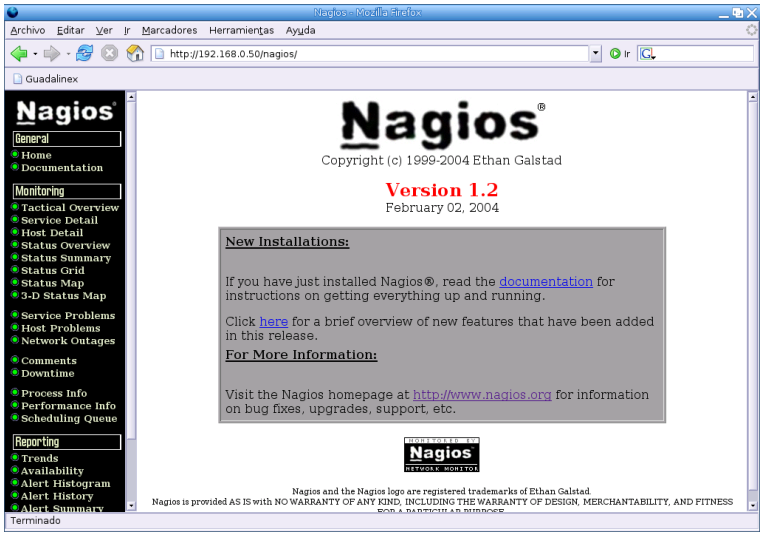
La podemos bajar, por ejemplo, de `http://www.nagios.org/download/` e instalarlos con

```
rpm -i nagios*
```

Durante la instalación será necesario indicarle el tipo de servidor web que se utilizará. También se solicitará la contraseña para acceder con el usuario `nagiosadmin` a la monitorización.

Finalizado el proceso de instalación únicamente hay que iniciar el servicio de Nagios con el script `/etc/init.d/nagios start` y acceder a la consola web de monitorización (`http://localhost/nagios`).

Figura 25.1: Pantalla inicial de Nagios



Tendremos que autenticarnos como usuario `nagiosadmin` y password la establecida en el proceso de instalación.

No será necesario modificar los ficheros principales de configuración de Nagios `/etc/nagios/nagios.cfg` y `/etc/nagios/cgi.cfg`.

25.1.3. Configuración de Nagios

La configuración de Nagios se basa en la definición de una serie de objetos. Mediante estos objetos se definirán los hosts, servicios, grupos de host, contactos, grupos de contactos, comandos, etc. Con estos elementos se define qué es lo que quiere monitorizarse y cómo quiere monitorizarse.

Servicios

Se refiere a los servicios que se van a monitorizar (SMTP, FTP, HTTP, ...) en los distintos servidores de la red. También entran en esta definición cualquier tipo de valor o métrica a aplicar sobre una medida realizada sobre un servicio. Junto con un servicio se define también el equipo que proporciona este servicio. El fichero que almacena la configuración de los servicios es `/etc/nagios/services.cfg`.

Algunos de los parámetros que definen un servicio en Nagios:

- `service_description`
- `host_name`
- `check_period`
- `contact_groups`
- `notification_options`
- `check_command`

Equipos

En el fichero `/etc/nagios/hosts.cfg` se incluirán todos los servidores que se van a monitorizar. Cada servidor tendrá al menos un servicio asociado, que será el que se monitorice.

Algunos de los parámetros que definen un equipo en Nagios:

- `host_name`
- `alias`
- `address`
- `check_command`

Grupos de equipos

Los grupos de equipos son conjuntos de equipos que tienen algo en común y que se agrupan para facilitar la administración. El fichero `/etc/nagios/hostgroups.cfg` define estos conjuntos de equipos. Relacionado con este concepto está los grupos de contactos.

Un equipo siempre pertenece a un grupo, siendo posible que un mismo equipo pertenezca a varios grupos.

Algunos de los parámetros que definen a un grupo de equipos en Nagios:

- `hostgroup_name`
- `alias`
- `contact_groups`
- `members`

Contactos

El fichero `/etc/nagios/contacts.cfg` define las personas, a través de las direcciones de correo electrónico, a las cuales se van a enviar las notificaciones que genera Nagios. Normalmente serán los responsables de los equipos o los encargados de su administración. Junto a la definición del contacto se definen también las condiciones en que la notificación se hará efectiva.

Algunos de los parámetros que definen a un contacto en Nagios:

- `contact_name`
- `alias`
- `host_notification_period`
- `service_notification_period`
- `service_notification_commands`
- `host_notification_commands`

Grupos de contactos

Al igual que ocurre con los equipos y los grupos de equipos, un contacto tiene que pertenecer a un grupo de contactos. Un grupo de contactos es un conjunto de personas (contactos) que se agrupan a la hora de recibir notificaciones. La configuración de un grupo de contactos se define en `/etc/nagios/contactgroups.cfg`.

Algunos de los parámetros que definen a un grupo de contactos en Nagios:

- `contactgroup_name`
- `alias`
- `members`

Comandos

Un comando es una tarea utilizada para chequear el estado de un servicio o de un aspecto concreto de un servidor. Se especifica la línea de comando y a partir de este momento se puede hacer referencia a este comando desde Nagios. Esto permite añadir nuevas funcionalidades a Nagios simplemente creando nuevos comandos. Los comandos se declaran en los ficheros `/etc/nagios/misccommands.cfg` y `/etc/nagios/checkcommands.cfg`

Algunos de los parámetros que definen a un comando en Nagios:

- `command_name`
- `command_line`

Periodos de tiempo

Se define un periodo de tiempo como un rango horario que se asigna para cada día de la semana. Este periodo de tiempo que se ha creado se puede asignar a una tarea concreta y formando así una planificación. Lo normal es definir el horario laboral para cada día de la semana y utilizarlo para el envío de notificaciones, de forma que solo se avise de un determinado problema si se produce en dicho intervalo de tiempo. La configuración de los periodos de tiempo se realiza en el fichero `/etc/nagios/timeperiods.cfg`.

Algunos de los parámetros que definen a un periodo de tiempo en Nagios:

- `timeperiod_name`
- `alias`
- `monday`, `tuesday`, `wednesday`, `thursday`, `friday`, `saturday`, `sunday`

25.1.4. Monitorizar un nuevo host

La configuración por defecto aparece con un nodo denominado `gw` y que se corresponde con el *gateway*. La mejor forma de comprender el funcionamiento de Nagios es añadir un nuevo nodo.

Es necesario añadir la definición del nuevo nodo a `/etc/nagios/hosts.cfg`. Por ejemplo, si nuestra IP es 192.168.0.50, quedaría

```
# 'guadalinex'
define host{
    use                     generic-host    ; Name of host template to use
    host_name               guadalinex
    alias                   Servidor Guadalinex
    address                 192.168.0.50
    check_command           check-host-alive
    max_check_attempts      10
    notification_interval   480
    notification_period     24x7
    notification_options    d,u,r
    }
```

A continuación se modifica el fichero `/etc/nagios/hostgroups.cfg`

```
# Definicion del grupo de nuestros servidores
define hostgroup{
        hostgroup_name  servidores
        alias           Servidores Linux
        contact_groups  linux-admin
        members         guadalinex
        }
```

Se modifica el fichero `/etc/nagios/contactgroups.cfg`

```
# 'linux-admin' contact group definition
define contactgroup{
        contactgroup_name       linux-admin
        alias                   Administradores Linux
        members                 administradores
        }
```

Se modifica el fichero `/etc/nagios/contacts.cfg`

```
# 'administradores' contact definition
define contact{
        contact_name                    administradores
        alias                           Administradores Linux
        service_notification_period     24x7
        host_notification_period        24x7
        service_notification_options    w,u,c,r
        host_notification_options       d,u,r
        service_notification_commands   notify-by-email
        host_notification_commands      host-notify-by-email
        email                           hugo@midominio.org
        }
```

Se modifica el fichero `/etc/nagios/services.cfg`

```
define service{
        use                             generic-service         ; Name of
            service template to use
        host_name                       guadalinex
        service_description             PING
        is_volatile                     0
        check_period                    24x7
        max_check_attempts              3
        normal_check_interval           5
        retry_check_interval            1
        contact_groups                  linux-admin
        notification_interval           240
        notification_period             24x7
        notification_options            c,r
        check_command                   check_ping!100.0,20%!500.0,60%
        }
```

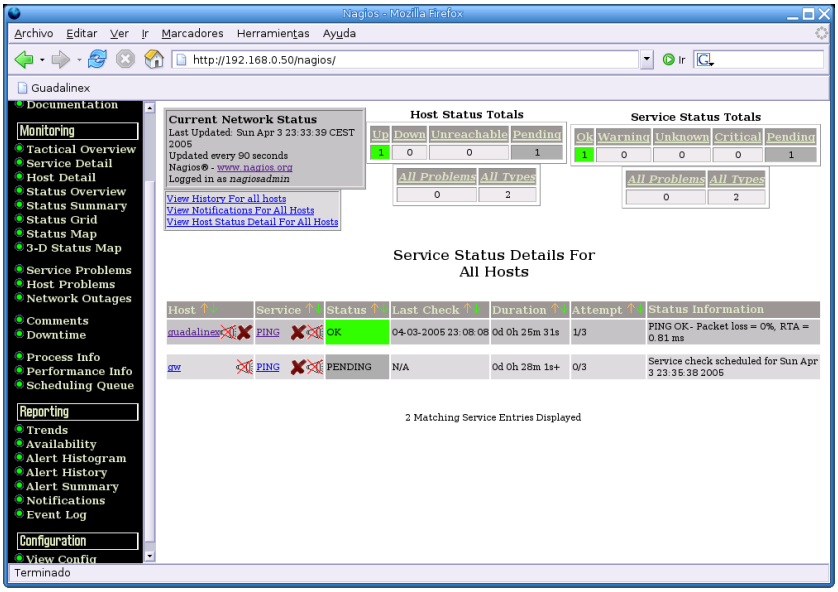
Una vez realizadas estas modificaciones:

```
root@guadalinex:~# /usr/sbin/nagios -v /etc/nagios/nagios.cfg
Nagios 1.2
Copyright (c) 1999-2004 Ethan Galstad (nagios@nagios.org)
Last Modified: 02-02-2004
License: GPL
Reading configuration data...
Running pre-flight check on configuration data...
Checking services...
        Checked 2 services.
Checking hosts...
        Checked 2 hosts.
Checking host groups...
        Checked 2 host groups.
Checking contacts...
```

```
        Checked 2 contacts.
Checking contact groups...
        Checked 2 contact groups.
Checking service escalations...
        Checked 1 service escalations.
Checking host group escalations...
        Checked 0 host group escalations.
Checking service dependencies...
        Checked 0 service dependencies.
Checking host escalations...
        Checked 0 host escalations.
Checking host dependencies...
        Checked 0 host dependencies.
Checking commands...
        Checked 88 commands.
Checking time periods...
        Checked 4 time periods.
Checking for circular paths between hosts...
Checking for circular service execution dependencies...
Checking global event handlers...
Checking obsessive compulsive service processor command...
Checking misc settings...
Total Warnings: 0
Total Errors:   0
Things look okay - No serious problems were detected during the pre-flight
    check
```

Para que la nueva configuración tome efecto es necesario reiniciar el servicio mediante la línea `/etc/init.d/nagios reload`. Una vez cargada la nueva configuración puede chequearse el estado del servicio a través del navegador, pulsando sobre SERVICE DETAILS:

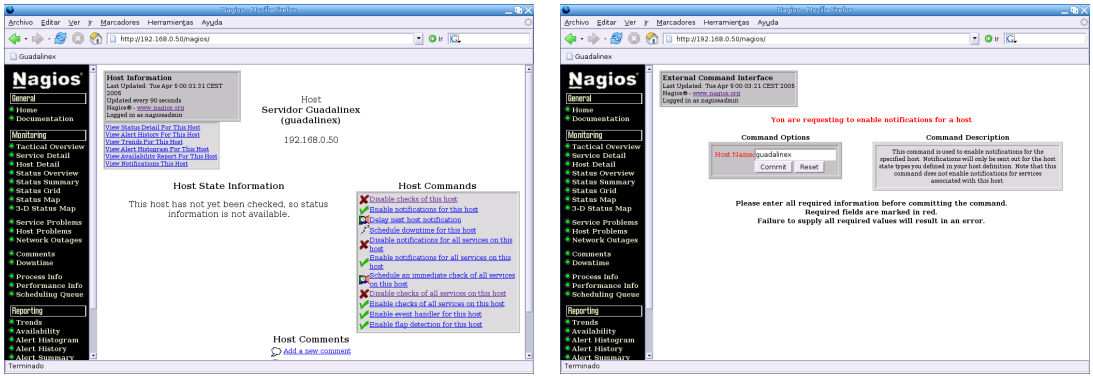
Figura 25.2: Monitorizar el nuevo host



Aquí no acaba el trabajo, existe un nuevo nodo definido en Nagios. Es necesario habilitar los chequeos y las notificaciones para este nodo, que aparecen deshabilitados (marca de color rojo

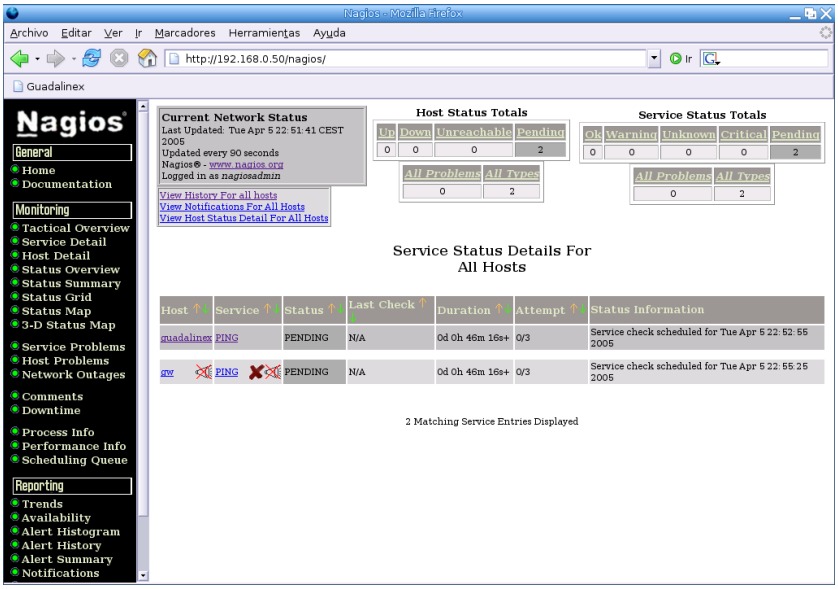
y altavoz tachado) en la consola web. Pulsando sobre el nombre del nuevo nodo, guadalinex, aparecen las propiedades del nodo y se permite habilitar todos los chequeos del nodo, de los servicios asociados al nodo y de las notificaciones. También se pueden habilitar los chequeos y notificaciones pulsando directamente sobre los símbolos.

Figura 25.3: Habilitar chequeos y notificaciones



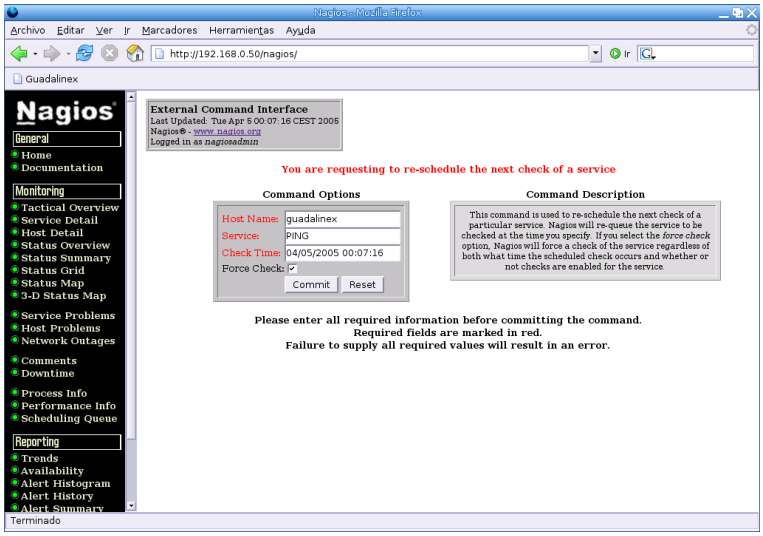
Una vez habilitados, desaparecen los indicadores anteriores.

Figura 25.4: Chequeos y notificaciones habilitados



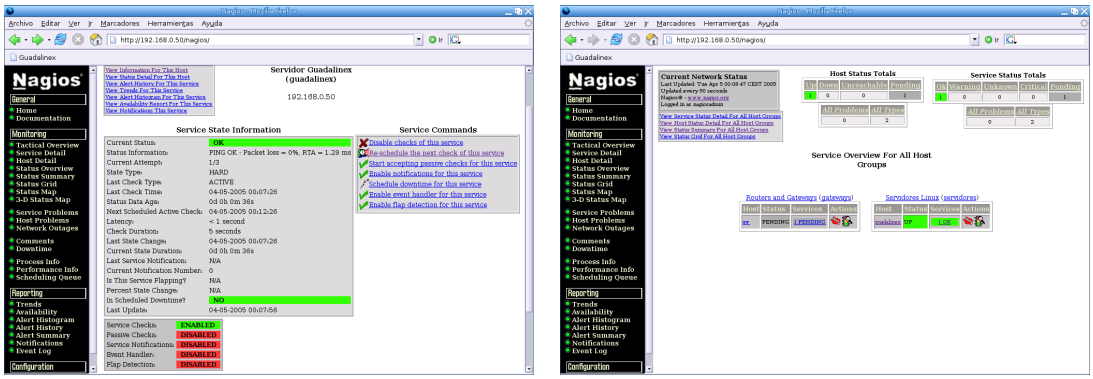
Por último, solo queda empezar a monitorizar el sistema. Si no queremos esperar a que se produzca el primer chequeo de forma automática, es posible forzarlo.

Figura 25.5: Forzar chequeo de un servicio



Una vez chequeado, Nagios mostrará el estado del servicio.

Figura 25.6: Estado de servicio OK



25.2. Monitorización de redes con ntop

Network top (`ntop`) es un software que nos permite monitorizar el tráfico de la red, con una gran potencia debido al gran número de protocolos que soporta. La utilidad `ntop` nos permite realizar un análisis de todo el tráfico que pasa a través de nuestro interfaz, por defecto el eth0. Si queremos sacar el máximo rendimiento a esta herramienta un buen sitio para hacerla funcionar es en una máquina por donde pase casi todo el tráfico de nuestra red: un cortafuegos o un proxy. Cuanto mayor porcentaje del tráfico de nuestra red pase por el interfaz o interfaces monitorizados por `ntop`, mayor y más real será la información.

Anteriormente hemos visto algún software de monitorización pero éstos no nos proporcionaban determinadas características que nos da `ntop`, por ejemplo: modelo de los equipos y S.O., trafico acumulado entre dos equipos, equipos que generan mayor cantidad de tráfico,...

Es fácil de instalar y de usar, ya que es vía web y se le pueden incorporar otras herramientas como el `rrdtools` para la generación de gráficas que facilitan la obtención de resultados.

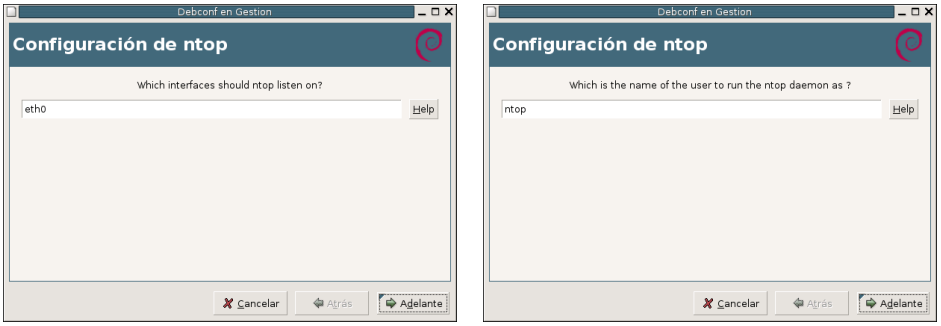
25.2.1. Instalación

Para la instalación de `ntop` vamos a utilizar Guadalinex 2004[1], de tal forma que es suficiente con:

```
root@Gestion:~# apt-get install ntop
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se áinstalarn los siguientes paquetes NUEVOS:
ntop
0 actualizados, 1 se áinstalarn, 0 para eliminar y 126 no actualizados.
3 no instalados del todo o eliminados.
Necesito descargar 2402kB de archivos.
Se áutilizarn 6525kB de espacio de disco adicional édespus de desempaquetar.
Des:1 http://http.guadalinex.org sarge/main ntop 2:3.0-3 [2402kB]
Descargados 2402kB en 45s (53,3kB/s)
Preconfiguring packages ...
```

A continuación nos aparece el asistente de configuración:

Figura 25.7: Configuración de ntop



(a) Interfaz de red (b) Usuario

Elegimos el interfaz por el que monitorizar y a continuación seleccionamos el usuario para ejecutar `ntop`.

Así se configurará de forma automática el *Network top* y todos aquellos paquetes de los que dependa (libgd2, apache2,...).

Una vez instalado y antes de arrancarlo debemos establecer la contraseña del usuario admin, para esto utilizaremos el comando `ntop -A`. Con esta opción nos pedirá introducir dicha contraseña y una vez realizado podremos arrancar el demonio con `/etc/init.d/ntop start`.

Ntop tiene su propio servicio http y https, por defecto se habilita el primero en el puerto 3000, si queremos habilitar otro puerto para http utilizaremos la opción `-w puerto` y si queremos habilitar el https utilizaremos la opción `-W puerto`. Para que estas opciones las tome al arrancar el demonio de ntop debemos editar el fichero `/etc/default/ntop` y descomentar el parámetro `OPTGET` y darle los valores que queramos, por ejemplo `OPTGET=" -w 2000 -WWW 2002"`.

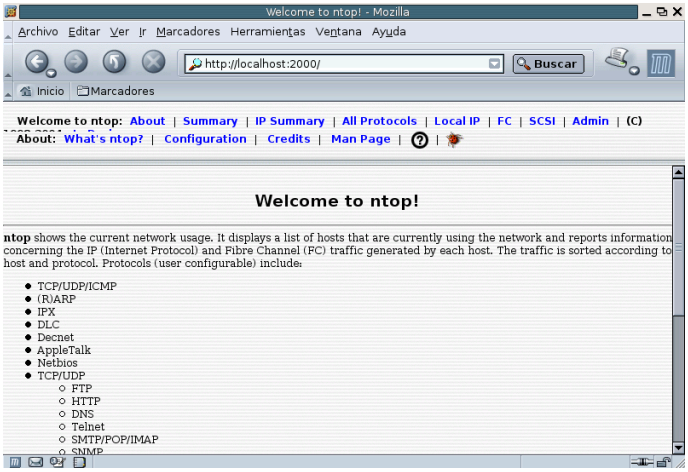
Con esto podemos empezar a trabajar con `ntop` mediante un navegador y podemos acceder al manual html para ver el resto de diferentes opciones.

[1]El paquete rpm para Fedora lo podemos bajar desde:
`http://www.ntop.org/ntop.html`

25.2.2. Datos en ntop

Una vez arrancado el demonio de `ntop`, podemos ver los primeros resultados directamente[2] en la url `http://localhost:2000`.

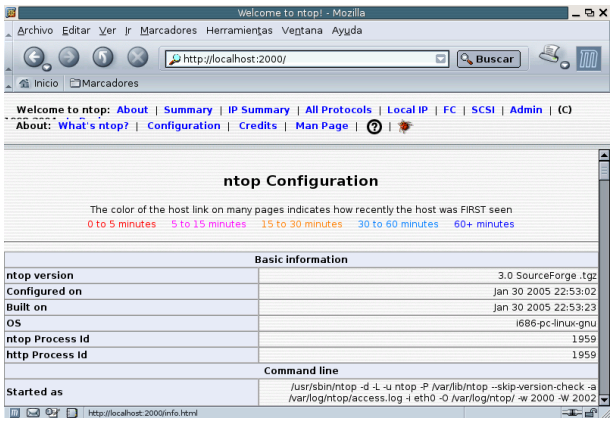
Figura 25.8: Inicio de ntop



En la pantalla de bienvenida, y según la distribución que estemos utilizando, nos aparecerá un menú con los distintos apartados en los que podemos entrar:

About: Muestra una explicación del programa, la configuración de ntop, los créditos y el manual en formato html.

Figura 25.9: Usando ntop -About



Summary: Nos muestra un resumen de las estadísticas obtenidas de forma global. Muestra los diferentes tipos de tráfico y sus carácterísticas, lista todos los equipos y el ancho de banda utilizado, la carga de la red y otras características como las vlans.

[2]Si no hemos cambiado los puertos será `http://localhost:3000`

Figura 25.11: Usando ntop - IP Summary

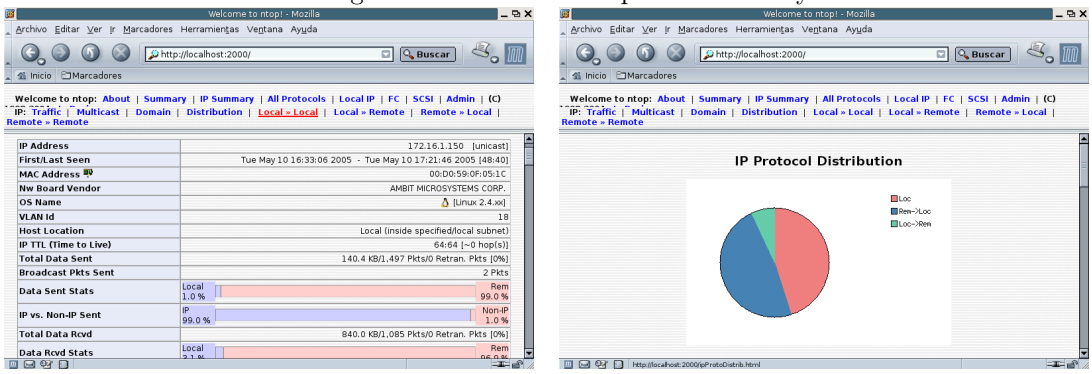
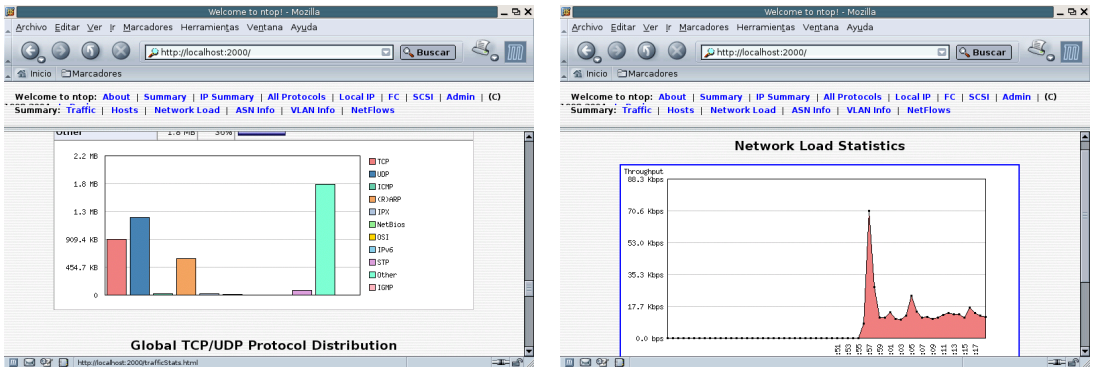


Figura 25.10: Usando ntop - Summary



IP Sumary Podemos obtener el tipo de tráfico cursado por cada host y acceder a la información del equipo. Nos permite ver las estadísticas distinguiendo entre tráfico local y remoto.

All Protocols En este apartado encontramos estadísticas de paquetes enviados y recibidos, así como una distribución de la actividad por franja horaria.

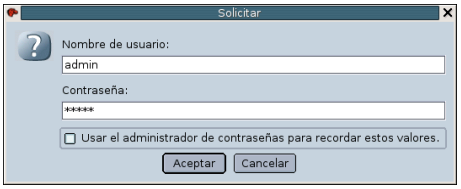
Local IP En este apartado se muestran diferentes cuadros comparativos de puertos utilizados por clientes y servidores, los sistemas operativos usados por las máquinas, los servicios de cada máquina o un cuadro con el tráfico entre máquinas.

FC Muestra los datos de las interfaces Fibre Channel.

SCSI Muestra los datos de las conexiones SCSI.

Admin Nos permite gestionar `ntop` vía web. Utilizando la primera vez el usuario admin con la contraseña inicial podemos dar de alta nuevos usuarios, crear nuevos filtros, añadir y configurar plugins, resetear las estadísticas, cambiar el interface e incluso parar el servicio.

Figura 25.12: Usando ntop - Admin



Estos apartados descritos anteriormente se corresponden con la distribución de Guadalinex 2004, en otras distribuciones la presentación de los menús puede variar un poco pero los datos siguen siendo los mismos. Como siempre lo mejor es instalarlo, ponerlo en funcionamiento e ir viendo el partido que podemos sacarle. Seguro que gran parte de la información nos puede resultar de bastante utilidad, sobre todo si tenemos que detectar máquinas que generen grandes cantidades de tráfico o tráfico no deseado.

Prácticas

Tipo I

E5-I-1

Realizar correctamente todas las preguntas del cuestionario que para esta entrega se ha habilitado en Moodle.

1. El disco duro puede utilizarse para realizar copias de seguridad:

   a) Solo de forma local, instalando el disco en el servidor del cual se quiere hacer copia de seguridad.

   b) Para hacer copias de seguridad solo se pueden utilizar cintas magnéticas.

   c) Sólo de forma remota, instalando el disco en un servidor remoto al que se vuelca la copia de seguridad del servidor.

   d) De forma local (instalando el disco en el servidor del cual se quiere hacer copia de seguridad) y de forma remota (instalando el disco en un servidor remoto al que se vuelca la copia de seguridad del servidor).

2. Las utilidades `dump/restore` nos permiten:

   a) Realizar copias de seguridad pero sin la opción de restauración de los datos guardados.

   b) Únicamente realizan restauraciones de las copias de seguridad que se han realizado con la utilidad `tar`.

   c) Permiten realizar copias/restauraciones de un sistema.

   d) Son utilidades para comprobar la integridad de un sistema de ficheros.

3. Indicar cómo se realizaría, mediante `rsync`, la copia de seguridad del directorio `/home/html` en la máquina remota `guadalinex.midominio.org` dentro del directorio `/backup`, siendo el tráfico encriptado.

   a) `rsync -a ssh /home/html root@guadalinex.midominio.org:/backup`

   b) `rsync -a /home/html root@guadalinex.midominio.org:/backup`

   c) `rsync -a -e encriptar /home/html root@guadalinex.midominio.org:/backup`

   d) `rsync -a -e ssh /home/html root@guadalinex.midominio.org:/backup`

4. ¿Cómo puede chequearse la política creada en `/etc/amanda/Mensual` para su uso con Amanda?

   a) `/usr/sbin/amanda check Mensual`

   b) `/usr/sbin/amandacheck Mensual`

   c) `/usr/sbin/amcheck Mensual`

d) `/usr/sbin/amcheck Daily`

5. ¿Cómo puede visualizarse la información sobre los últimos accesos de los usuarios contenida en `/var/log/lastlog`?

   a) `more /var/log/lastlog`
   b) `cat /var/log/lastlog`
   c) `lastlog`
   d) `lastlog /var/log/lastlog`

6. Para controlar qué y dónde se registran los eventos producidos en el sistema es necesario:

   a) Modificar el fichero `/etc/syslog.conf` y reiniciar el servicio de log
   b) Modificar el fichero `/var/log/messages`
   c) Modificar el fichero `/etc/syslog.conf`
   d) No se pueden modificar los valores que tiene el sistema por defecto.

7. Acabamos de instalar Webmin y no hemos tocado aún nada de la configuración, ¿cómo puede acceder a él?

   a) Mediante el cliente webmin, disponible únicamente para Linux.
   b) Mediante cualquier navegador web y desde cualquier sistema.
   c) Solo puede accederse en modo texto desde localhost.
   d) Mediante cualquier navegador web y desde un sistema para el cual se haya definido el permiso de acceso en Webmin.

8. ¿Qué servicios permite gestionar Webmin en un sistema?

   a) Cualquier servicio instalado en el sistema es posible administrarlo mediante Webmin.
   b) Únicamente se pueden administrar con Webmin los servicios web.
   c) Se pueden administrar con Webmin los servicios de correo y web.
   d) Webmin no sirve para administrar servicios, sirve para monitorizarlos.

9. Una vez instalado ntop:

   a) Debe arrancarse utilizando `/etc/init.d/ntop init`
   b) Debe arrancarse utilizando `/etc/init.d/ntop start` y poner la password del usuario admin
   c) Debe ejecutarse `ntop -A` y luego arrancar el demonio
   d) Debe crearse el fichero `ntop.config` con el usuario ntop

10. Para que se acceda a ntop mediante un puerto seguro:

    a) Por defecto se accede a un puerto seguro
    b) Hay que utilizar los parámetros `-w 0` y `-W 443`
    c) Debemos utilizar apache para redirigir el contenido
    d) No se puede acceder usando ssl

E5-I-2 AMANDA

Crear con AMANDA una política de copia de seguridad en la cual se copiarán los ficheros del directorio `/home/`. El nombre de esta política será Diaria.

A diferencia del ejemplo de los apuntes, el número de días definidos en esta política será de 7, por lo que las cintas utilizadas también serán 7.

Se realizarán las copias a disco simulando cintas, tal como se ha visto en los apuntes. El patrón que se sigue para definir las cintas será DIARIAXX, siendo necesario definirlo en `amanda.conf`.

La ruta donde están definidas las cintas debe ser `/backups/Diaria/tapeXX` y se etiquetarán tal como se muestra a continuación:

```
/usr/sbin/amlabel Diaria DIARIA01 slot 1
    /usr/sbin/amlabel Diaria DIARIA02 slot 2
    ...
    /usr/sbin/amlabel Diaria DIARIA07 slot 7
```

Se pide mostrar la salida del comando `amcheck` para la política Diaria sin errores, así como la salida de ejecutar un backup con esta política (contenido del correo que se envía con el resultado del backup).

Pistas: Pueden copiarse los ficheros de configuración del ejemplo de los apuntes, siendo necesario modificar `amanda.conf` y `changer.conf`. No debe olvidarse tampoco la creación de los directorios que albergan las cintas.

El resultado de la práctica (ficheros de configuración, explicaciones, etc) debe mandarse en formato OpenOffice en un fichero de nombre `e5-i-2.sxw`

Tipo II

E5-II-1 NAGIOS

Modificar los ficheros de configuración de NAGIOS del ejemplo de los apuntes de forma que monitorice vuestro ordenador.

Se pide mostrar la salida del comando

`/usr/sbin/nagios -v /etc/nagios/nagios.cfg`

así como incluir los ficheros que ha sido necesario modificar para que monitorice vuestro servidor.

El resultado de la práctica (ficheros de configuración, explicaciones, etc) debe mandarse en formato OpenOffice en un fichero de nombre `e5-ii-1.sxw`

E5-II-2 NTOP

Instalar `ntop` en el equipo. Acceder a la página web del curso con `ntop` ejecutándose. Se visualizarán los datos de las características del equipo donde se está ejecutando `ntop` y del host `mileto.cica.es`.

El resultado de las pantallas que presenta el navegador y las explicaciones, deben mandarse en formato OpenOffice en un fichero de nombre `e5-ii-2.sxw`

Bibliografía

[1] *Securing and Optimizing Linux: The ultimate solution.* Gerhard Mourani

[2] *A brief tutorial on dump and restore.* `http://www.nethamilton.net/docs/`

[3] *Página oficial de AMANDA.* `http://www.amanda.org/`

[4] *Backups en disco duro con AMANDA* `http://www.kleenux.org/articulos/amanda-tapeless/amanda-tapeless.html`

[5] *Administración de sistemas Linux.* Dee-Ann LeBlanc

[6] *Seguridad Pactica en UNIX e Internet.* Simson Garfinkel y Gene Spafford

[7] *System Administation with Webmin.* Joe Cooper

[8] Página oficial de Nagios. `http://www.nagios.org/`

[9] Página oficial de ntop. `http://www.ntop.org/`

Parte VI

Seguridad

Capítulo 26

Blindaje del sistema

Intruso (hacker): sustantivo.

1)Se dice de quien goza averiguando los detalles de sistemas de cómputo y cómo llevarlos a su límite, en contraposición a la mayoría de usuarios que prefiere aprender lo mínimo necesario.

2)Se dice de quien escribe programas en forma entusiasta o goza programando en lugar de pensar cómo programar.

Guy L. Steele y cols. *The Hacker's Dictionary*

UNIX no se diseñó para ser seguro. Se diseñó para que se pudiera usar la seguridad

Dennis Ritchie

26.1. Seguridad en UNIX

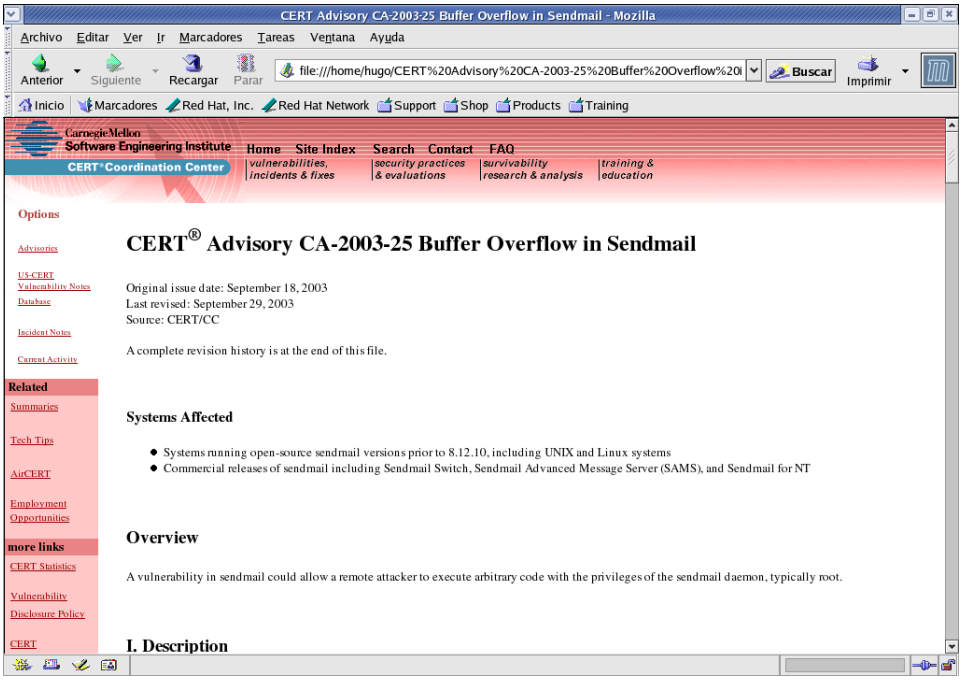
En los años 90 mucha gente pensaba que la seguridad en UNIX era una fantasía. Con la facilidad con que un gurú UNIX podía irrumpir en un sistema y tomar el control del mismo, no concebían la idea de seguridad en estos sistemas.

Desde entonces, las cosas han cambiado, pasando a considerarse los sistemas UNIX como sistemas operativos relativamente seguros. Esto es debido a que en el transcurso de los años se ha avanzado en el estudio de la defensa preventiva de los sistemas UNIX. Cada vez pasa menos tiempo desde que se descubre un agujero de seguridad en algún componente, hasta la aparición del parche o actualización correspondiente.

Sin embargo, por su diseño particular, sigue teniendo fallos. Uno de ellos y quizás el más importante es el *superusuario*, que sigue constituyendo un punto único de ataque. Una vez logrado el acceso a esta cuenta, el atacante tiene control absoluto sobre el sistema.

Fue a raíz del ataque protagonizado por Robert T. Morris en 1988 cuando el tema de la seguridad en sistemas operativos y redes se convirtió en un factor a tener muy en cuenta por cualquier administrador de sistemas. Poco después de este incidente, la agencia DARPA (*Defense Advanced Research Projects Agency*) creó el CERT (*Computer Emergency Response Team*) `http://www.cert.org`, un grupo formado en su mayor parte por voluntarios cualificados de la comunidad informática. El objetivo era facilitar la respuesta rápida ante los problemas de seguridad.

Figura 26.1: Ejemplo de vulnerabilidad aparecida en CERT



Han pasado más de 15 años desde la creación del primer CERT y sigue siendo patente la preocupación por los temas relativos a la seguridad y, sobre todo, se hace patente la necesidad de esa seguridad.

En la actualidad, donde el comercio electrónico y las redes internacionales están a la orden del día, todos los sistemas son potenciales víctimas de un intruso. Prácticamente todos los meses aparece alguna noticia sobre la intrusión en una gran compañia. Estas intrusiones pueden ser totalmente inofensivas y promovidas por la curiosidad o, en el lado opuesto, promovidas por intenciones de lo más siniestras. En este último caso las consecuencias suelen ser desastrosas.

Aunque no se borre o modifique ningún archivo del sistema, es obligación de los administradores una vez detectada la intrusión, chequear el sistema en busca de posibles destrozos o, más importante aún, la colocación de algún programa que realice las funciones de puerta trasera por parte de los intrusos.

Se han desarrollado una gran cantidad de herramientas y técnicas con objeto de ayudar a los administradores de sistemas y servir de medidas preventivas frente a intrusiones.

26.2. Conceptos sobre seguridad

Antes de empezar a hablar de la seguridad de sistemas, es conveniente dejar claro qué se entiende por seguridad. Seguridad es una característica de cualquier sistema informático, que indica que ese sistema está libre de todo peligro de accesos no permitidos que puedan provocar un daño en él. Podría denominarse infalible a un sistema seguro según esta definición.

Particularizando en el terreno que nos ocupa, es muy difícil conseguir esta característica, prácticamente imposible. Se suaviza entonces la definición de seguridad y se habla de fiabilidad, entendiendo como tal la probabilidad de que un sistema se comporte tal y como se espera de él. Más que de sistemas seguros, se habla de sistemas fiables.

Para mantener un sistema seguro o fiable, deben garantizarse tres aspectos:

Confidencialidad. Los objetos de un sistema han de ser accedidos únicamente por los métodos permitidos para ello, y estos métodos no harán disponible esta información a terceros.

Integridad. Los objetos sólo pueden ser modificados por elementos autorizados y de una forma controlada.

Disponibilidad. Los objetos del sistema tienen que permanecer accesibles a elementos autorizados.

Dependiendo del entorno donde se trabaja, puede darse más importancia a alguno de los aspectos anteriores. Por ejemplo, si estamos trabajando en un banco, es prioritario mantener la integridad de los datos, pasando los otros aspectos a un plano inferior.

Una vez claro el concepto de seguridad y lo que conlleva, surge una nueva pregunta ¿qué debe protegerse? Los tres elementos principales a proteger en cualquier sistema informático son:

- Software. Conjunto de programas lógicos que hacen funcionar al hardware instalado, tanto sistemas operativos como aplicaciones.
- Hardware. Conjunto formado por los elementos físicos del sistema informático.
- Datos. Conjunto de información lógica que maneja el hardware y el software.

Una vez claros los conceptos relativos a seguridad de sistemas, es hora de pasar a la acción. Para proteger nuestro sistema la primera tarea a realizar es un análisis de las amenazas potenciales que puede sufrir[1] y, a partir de la información obtenida, diseñar una política de seguridad que defina responsabilidades y reglas a seguir para evitar las amenazas o minimizar su incidencia. A los mecanismos utilizados para la implementación de esta política, se les denomina mecanismos de seguridad. Estos mecanismos serán la parte más visible del sistema de seguridad.

Los mecanismos de seguridad se dividen en tres grupos:

Prevención. Son mecanismos que aumentan la seguridad de un sistema durante el funcionamiento normal de éste, previniendo la ocurrencia de violaciones a la seguridad.

Detección. Son aquellos que se utilizan para detectar violaciones o intentos de violación del sistema.

Recuperación. Son los que se aplican cuando se ha detectado una violación del sistema y quiere devolverse el mismo a un estado estable. Dentro de estos mecanismos está el denominado análisis forense. El objetivo no sólo es retornar a una situación segura y estable, sino también conocer el alcance real de la violación, las actividades llevadas a cabo por el intruso y la forma de entrada.

Debe emplearse el máximo esfuerzo en implementar unos mecanismos de prevención lo suficientemente robustos en nuestro sistema. Se hace patente el dicho "más vale prevenir que curar" ya que es mejor dedicar tiempo a evitar los ataques que a recuperar un sistema que han violado.

26.3. Planificación de Seguridad

La planificación de la seguridad de un sistema puede dividirse en seis etapas diferentes:

1. Planificación de las necesidades de seguridad
2. Análisis de riesgos
3. Análisis de costo-beneficio

[1] Si tenemos un ordenador para uso personal que no tiene acceso a ninguna red, da igual que tenga todas las vulnerabilidades posibles referentes a acceso remoto, no es posible explotarlas. Aunque de qué nos sirve...

4. Creación de políticas de seguridad
5. Implementación
6. Auditoría y respuesta ante incidentes

Un servidor será seguro si se comporta de la manera esperada. Sin embargo, hay que tener en cuenta que pueden considerarse muchos tipos de seguridad:

- Confidencialidad. Proteger la información para que nadie pueda leerla o copiarla sin autorización del dueño.
- Integridad de los datos. Proteger la información (datos y programas) para evitar el borrado o alteración de la misma sin el permiso del dueño.
- Disponibilidad. Proteger los servicios para que no se degraden o dejen de estar disponibles sin autorización.
- Consistencia. Asegurar que el sistema se comporta como esperan los usuarios autorizados.
- Control. Reglamentar el acceso al sistema.
- Auditoría. Registro de las acciones que realizan tanto los usuarios autorizados, como los intentos de acceso de los no autorizados.

26.4. Mecanismos de prevención

Lamentablemente, muchos administradores de equipos UNIX no disponen de los conocimientos o simplemente deciden no dedicar tiempo a la seguridad del sistema del cual son responsables. Esto hace que sean servidores abiertos a cualquier ataque.

Es necesario marcar una serie de pautas o recomendaciones mínimas que ayuden a minimizar este problema al máximo con técnicas de prevención. Estos mecanismos deben ser importantes para cualquier administrador y deben tenerse en cuenta en todas y cada una de las máquinas que administre. Sin embargo, esto no significa que el sistema esté a salvo, son simplemente actuaciones básicas.

26.4.1. Cierre de servicios innecesarios

Cada uno de los servicios ofrecidos en el sistema se convierte en una potencial puerta de entrada a nuestro sistema. Uno de los primeros puntos a comprobar es el superdemonio de red en cualquiera de sus versiones (`xinetd` o `inetd`). Se cerrará cada uno de los servicios que no se utilice o que no se conozca para qué sirve. En el caso de desactivar un servicio que sea necesario, simplemente se activa de nuevo, modificando los ficheros necesarios.

En el caso de `xinetd` se accederá al directorio `/etc/xinetd.d` editando los ficheros que correspondan a servicios que no sean necesarios. Únicamente se establece el atributo `disable=yes` tal como se vio en las primeras entregas del curso.

Para `inetd` únicamente hay que editar el fichero `/etc/inetd.conf`, comentando las entradas correspondientes a servicios que no se vayan a utilizar.

Una vez comprobados los servicios arrancados por `xinetd/inetd` se verán los servicios que se inician al arrancar el servidor. Estos servicios arrancarán procesos independientes que estarán a la escucha de peticiones desde el exterior. La localización de los scripts de arranque es `/etc/rc?.d` o `/etc/rc.d`. Supongamos que queremos desactivar el servicio de correo proporcionado por Sendmail.

```
root@guadalinex:~# find /etc/rc* -name "*sendmail"
/etc/rc0.d/K19sendmail
/etc/rc1.d/K19sendmail
/etc/rc2.d/S21sendmail
/etc/rc3.d/S21sendmail
/etc/rc4.d/S21sendmail
/etc/rc5.d/S21sendmail
/etc/rc6.d/K19sendmail
root@guadalinex:~# find /etc/init.d/ -name "sendmail"
/etc/init.d/sendmail
```

Bastará con renombrar los archivos `S21sendmail` a `noS21sendmail` para que en el próximo arranque del sistema, este servicio no se inicie. Hay que tener en cuenta que el renombrar estos ficheros hace que no se arranque en el próximo inicio de la máquina, pero seguirá activo hasta que se produzca ese evento. Pararemos el servicio con:

```
root@guadalinex:~# /etc/init.d/sendmail stop
```

26.4.2. Instalación de envolventes (*wrappers*)

A pesar de seguir las recomendaciones anteriores y suponiendo que nuestro sistema únicamente tiene activos los servicios necesarios, aún puede hacerse más. El siguiente paso es securizar los servicios que se ha decidido mantener. Es muy conveniente el uso de *wrappers* ya que nos permiten restringir el acceso a los servicios, aceptando únicamente las conexiones que se definan.

Anteriormente se vio uno de los más extendidos, `tcp-wrappers`, por lo que éste es un buen momento para poner en acción los conocimientos adquiridos y empezar a restringir el acceso a los servicios activos. Es conveniente que se configuren `tcp-wrappers` para que únicamente accedan a los servicios las direcciones IP que se indiquen. Dejar que una persona externa a nuestra organización pueda acceder a un servicio es dejarle la puerta abierta para que acceda de forma incontrolada al mismo.

26.4.3. Seguridad de las claves

Claves en Linux

Las claves en Linux no se almacenan en un formato legible. La contraseña se convierte en una cadena de texto mediante un algoritmo criptográfico que proporciona una cadena totalmente distinta de la original. La función que utiliza para realizar esta tarea es `crypt()`. Es una función de un único sentido[2], lo que evita que puede utilizarse para averiguar la clave, cuyo resultado se almacenará en los archivos `/etc/passwd` o `/etc/shadow`.

Cuando un usuario se conecta al sistema con `/bin/login,` lo que se hace realmente es generar de nuevo la clave modificada a partir de la que se introduce por el teclado. La función `crypt()` la transforma y la compara con la que hay almacenada en el sistema. La función `crypt()` se ha demostrado que es suficientemente sólida. Es una mala elección de contraseña lo que provoca los problemas que se verán a continuación.

A pesar de que el código fuente de `crypt()` está disponible, no se ha descubierto ninguna técnica para convertir la contraseña cifrada de nuevo en la contraseña original. Posiblemente, la traducción inversa no sea posible. La única forma conocida de vencer la seguridad de las contraseñas es mediante un ataque de fuerza bruta o mediante un ataque de diccionario.

Elección de las contraseñas

Una contraseña mala es una posible puerta abierta al sistema. Aunque son una de las partes más importantes del sistema, normalmente no se proporciona a los usuarios instrucciones concretas

[2]Puede utilizarse dicha la función, pero no existe la función inversa.

sobre cómo elegir y guardar la contraseña. Todo usuario debe saber que si elige una mala contraseña o se la comunican a alguien que no sea de fiar, comprometen toda la seguridad del sistema.

Una contraseña mala es aquella que se puede adivinar fácilmente. En el mundo real, los intrusos en lugar de comprobar las contraseñas a mano, utilizan sus propios ordenadores para comprobarlas de forma automática. En lugar de probar todas las combinaciones de letras lo que hacen es probar las contraseñas más comunes.

Surge la pregunta ¿cuáles son las malas contraseñas?, veremos algunos ejemplos:

- Nombre de usuario o el de un conocido o familiar
- Nombres escritos al revés, incluso si mezclan mayúsculas y minúsculas
- Contraseñas cortas de cualquier tipo
- Números telefónicos
- Personajes de películas
- Basadas en modificaciones simples de una palabra (sustituir I por 1, E por 3, ...)
- Palabras en otros idiomas

Es recomendable utilizar mayúsculas y minúsculas, además de dígitos y símbolos de puntuación junto con letras. Sin embargo, no debe caerse en la elección de contraseñas difíciles de recordar y que obliguen a escribirlas en algún sitio, no siendo recomendable que superen los 7 u 8 caracteres.

Para elegir una buena contraseña, se pueden tomar dos palabras cortas y combinarlas intercalando un carácter especial o un número. Otra opción puede ser componerla como un acrónimo de una frase o poema que nos guste, por ejemplo, “Esta contraseña es lo suficientemente segura” generaría la clave E2c0E0l4Ss. Hemos tomado las primeras letras alternando mayúsculas y minúsculas e intercalando el número 2004[3].

En caso que el usuario maneje varias cuentas a la vez en el mismo o distintos sistemas, sería un error utilizar la misma contraseña en todos los sistemas, ya que si averiguan la contraseña de una de las cuentas, la seguridad del resto de sistemas y cuentas se verá comprometida. Un enfoque muy válido es utilizar una contraseña básica y modificarla en función de la cuenta o del sistema en el que estemos accediendo.

Hay una película bastante conocida llamada “Juegos de Guerra” en la que un joven realizaba sus primeros pinitos en la infiltración clandestina de sistemas. Accedía al ordenador central de su instituto para cambiarse las calificaciones, gracias al listado de claves que existía en un papel en la secretaría del centro. Lamentablemente, esto sucede cientos de veces. Una advertencia básica es que los usuarios no anoten la contraseña nunca. Aún así, si un usuario quiere escribir su contraseña en algún sitio debe seguir las siguientes recomendaciones:

- Al escribirla, no identificarla como contraseña
- No incluir el nombre de usuario ni los datos del servidor al que se accede
- Guardar en lugar seguro

26.4.4. Seguridad de los usuarios

Usuarios normales

Se asumirá como práctica de seguridad habitual, en lo referente a usuarios, el uso de las utilidades de clave `shadow`. Los ficheros que se utilizan para la gestión de los usuarios son:

[3]Se aconseja encarecidamente no utilizar este ejemplo concreto en nuestro sistema ya que cualquiera que lea esta documentación conocerá la clave.

```
/etc/passwd
/etc/shadow
/etc/group
/etc/gshadow
```

Para prevenir un borrado o sobreescritura accidental de estos ficheros, es recomendable activar el bit inmutable. Es una forma de protección que también previene de la creación de enlaces simbólicos sobre estos ficheros con objeto de realizar posteriormente un ataque.

```
[root@fedora root]# lsattr /etc/passwd
----------------- /etc/passwd
[root@fedora root]# chattr +i /etc/passwd
[root@fedora root]# chattr +i /etc/shadow
[root@fedora root]# chattr +i /etc/group
[root@fedora root]# chattr +i /etc/gshadow
```

Hay que tener en cuenta que una vez establecido este bit cualquier modificación sobre estos ficheros dará error. De esta forma, para añadir un nuevo usuario es necesario deshabilitar el bit de nuevo. Lo mismo ocurrirá con cualquier paquete de software que durante el proceso de instalación requiere la creación de nuevos usuarios o grupos.

```
[root@fedora root]# lsattr /etc/passwd
---i------------- /etc/passwd
[root@fedora root]# adduser gandalf
adduser: unable to open password file
```

Un punto fundamental en la seguridad de los accesos a un sistema, es el referido a las claves de usuarios. Muchos usuarios piensan que sus ficheros y programas están lo suficientemente protegidos por la elección de una "buena" clave, pero esto no es así. No existen claves irrompibles. Si se da suficiente tiempo y recursos a un intruso, tarde o temprano conseguirá averiguar la clave de un usuario, ya sea por fuerza bruta o por ingeniería social.

En primer lugar, la clave que se elige debe ser segura y no adivinable fácilmente. Para ello las siguientes recomendaciones pueden ser de utilidad:

- La longitud de la clave debe establecerse entre 5 y 8 caracteres, incluyendo algún número o carácter especial en la misma.
- No debe ser trivial (basada en el nombre, familia, lugar de trabajo, ...).
- Debe ser revocada y reseteada después de un número determinado de intentos.

El fichero de sistema `/etc/login.defs` es el fichero de configuración para la utilidad `login`. Es necesario revisarlo una vez que se ha instalado el sistema para revisar los valores que tiene configurados por defecto y establecerlos de acuerdo a las consideraciones de seguridad que se hayan adoptado.

Es recomendable cambiar los parámetros por defecto que se utilizan en la creación de nuevos usuarios. Estos parámetros afectan a la caducidad de las claves y a los grupos a los que pertenecen los nuevos usuarios cuando son creados entre otros[4].

4

Debian:

- Los campos del fichero `/etc/login.defs` cambian los valores por defecto.
- Si bien se puede crear el fichero `/etc/default/useradd`, es mejor usar el comando `adduser`. Su fichero de configuración es `/etc/adduser.conf` y está muy bien documentado.

Fichero configuración	Campo	Valor por defecto (Fedora)	Descripción
/etc/login.defs	PASS_MAX_DAYS	99999	Máximo número de días que una clave es válida
/etc/login.defs	PASS_MIN_DAYS	0	Máximo número de días permitido entre el cambio de clave
/etc/login.defs	PASS_WARN_AGE	7	Número máximo de días antes de forzar un cambio de clave
/etc/login.defs	UID_MIN	500	Mínimo valor para el UID automático
/etc/login.defs	GID_MIN	500	Mínimo valor para el GID automático
/etc/login.defs	PASS_MIN_LEN	5	Longitud máxima de clave aceptable (El módulo `pam_cracklib` sobreescribe este valor con el parámetro `minlen`)
/etc/default/useradd	GROUP	100	Grupo por defecto
/etc/default/useradd	HOME	/home	Directorio local de usuario
/etc/default/useradd	INACTIVE	-1	Máximo número de días después de la expiración de la clave en que el usuario puede cambiar la clave
/etc/default/useradd	EXPIRE	n/a	Fecha de expiración de una cuenta en el formato AAAA-MM-DD
/etc/default/useradd	SHELL	/bin/bash	Shell por defecto
/etc/default/useradd	SKEL	/etc/skel	Directorio de perfil por defecto

Otra consideración a tener en cuenta es la variable de entorno `$PATH`. Cuando un usuario ejecuta un comando, el *shell* buscará en cada uno de los directorios existentes en el *path,* hasta encontrar un comando con el mismo nombre que el tecleado, en cuyo caso lo ejecuta sin más. En caso de no encontrarlo dará un mensaje de error.

¿Qué problemas de seguridad presenta esta variable? Es muy recomendable comprobar que ninguno de los directorios que aparecen en `$PATH` del superusuario tienen permiso de escritura para los usuarios normales. Esto incluye a directorios como `/tmp/` o "`.`".

Imaginemos la siguiente situación. El usuario `root` de nuestro sistema tiene incluído en su variable `$PATH` el directorio actual como uno más donde buscar ejecutables. Si este usuario desea comprobar el contenido del directorio `/tmp/` o el de `$HOME` de alguno de sus usuarios, seguramente cambiará su directorio actual al del usuario en cuestión. ¿Qué sucede si "`.`" está en primer lugar en la variable `$PATH`? El *shell* buscará primero en el directorio actual (recordemos que es el del usuario) y podríamos encontrarnos con:

```
[root@fedora root]# cd /home/hugo/
[root@fedora hugo]# cat ls
#!/bin/bash
cd /
rm -rf *
```

Si tecleamos en la línea de comandos `ls,` se ejecutará el *script* anterior, borrando todo el sistema. Un simple `ls` habría conseguido que se borrase parte del sistema o todo, simplemente porque el administrador no ha tenido la precaución de configurar convenientemente la variable `$PATH`.

Surge entonces la pregunta ¿si ponemos el directorio "." al final de `$PATH` se soluciona el problema? La respuesta es contundente, NO, el problema sigue existiendo. Supongamos ahora que el script se llama *`moer`*. No es un comando que exista, pero ¿cuántas veces nos hemos equivocado al escribir el comando `more` y hemos tecleado `moer`? Si tecleamos `moer`, el shell buscará sin encontrarlo hasta llegar al ultimo directorio de `$PATH` y ejecutará el script. Tenemos el mismo resultado, borrado total o parcial del sistema. Parece claro, tras estos ejemplos que no es una buena práctica poner el directorio "." en la variable de entorno `$PATH` del superusuario.

Usuario root

La cuenta de usuario `root` es la que tiene más privilegios en un sistema UNIX. Este usuario no tiene restricciones de seguridad por lo que hay que operar con esta cuenta con mucha precaución.

Es importante no dejarse nunca esta cuenta abierta en la consola. En caso de que esto ocurra, puede establecerse un tiempo de desconexión del sistema si no se registra ninguna actividad desde esta cuenta. Así, puede establecerse que a los 3 minutos desde la última operación hecha desde una sesión del usuario `root` se desconecte dicha sesión[5].

Como medida de seguridad adicional también puede restringirse el uso del comando `su` para acceder a la cuenta `root`. Puede configurarse el sistema para que sólo los usuarios de un determinado grupo accedan a la cuenta de `root` mediante `su`. El fichero de configuración implicado es `/etc/pam.d/su`

```
auth sufficient /lib/security/pam_rootok.so
auth required /lib/security/pam_stack.so service=system-auth
account required /lib/security/pam_stack.so service=system-auth
password required /lib/security/pam_stack.so service=system-auth
session required /lib/security/pam_stack.so service=system-auth
session optional /lib/security/pam_xauth.so
```

Las líneas que es necesario añadir para reflejar esta configuración dejan el fichero `/etc/pam.d/su` como puede verse a continuación:

```
#Uncomment the following line to implicitly trust users in the 'wheel' group
    .
#auth sufficient /lib/security/pam_wheel.so trust use_uid
#Uncomment the following line to require a user to be in the 'wheel' group.
#auth required /lib/security/pam_wheel.so use_uid
auth sufficient /lib/security/pam_rootok.so
auth sufficient /lib/security/pam_wheel.so trust use_uid
auth required /lib/security/pam_wheel.so use_uid
auth required /lib/security/pam_stack.so service=system-auth
account required /lib/security/pam_stack.so service=system-auth
password required /lib/security/pam_stack.so service=system-auth
session required /lib/security/pam_stack.so service=system-auth
session optional /lib/security/pam_xauth.so
```

Usuarios especiales

Es importante deshabilitar todas las cuentas creadas por defecto en la instalación por los desarrolladores de algunos productos, si no se van a usar en el sistema. Algunas de estas cuentas existen por defecto, incluso si no se han instalado los programas o utilidades asociados. Cuantas más cuentas tenga un sistema más posibilidades hay de encontrar un acceso al mismo a través de una de estas cuentas.

[5] Por ejemplo, añadiendo dentro del fichero `/etc/profile` una variable `TMOUT` a la que se le da un valor en segundos que indica el tiempo de desconexión por inactividad. Podemos incluirla tambien en `.bashrc` de los usuarios para controlar el tiempo de inactividad.

El mismo procedimiento se seguirá con los grupos genéricos instalados en el sistema por defecto. En definitiva, todo aquello que no se use debe ser desinstalado. Esto es una norma general de seguridad del sistema que debe llevarse a cabo siempre.

26.5. SELinux

26.5.1. ¿Qué es SELinux?

A continuación, se explica brevemente en qué consiste SELinux. Sólo queremos que tengáis una pequeña noción de este nuevo Módulo de Seguridad que con el tiempo irá teniendo un mayor peso en las nuevas distribuciones. No entraremos en instalación ni configuración y solo mostraremos algún pequeño ejemplo que ayude a comprender los conceptos básicos, quien esté más interesado en los enlaces de la NSA y Fedora puede encontrar casi toda la información disponible.

Security Enhanced Linux, SELinux, es un sistema que modifica el núcleo de Linux, fortaleciendo los mecanismos de control de acceso y forzando la ejecución de los procesos dentro de un entorno con los mínimos privilegios necesarios.

La primera versión de SELinux se remonta a finales del año 2000 de manos de la NSA (Agencia Nacional de Seguridad de los Estados Unidos). El objetivo del mismo es, por un lado, demostrar la posibilidad de implementar el modelo de seguridad de control de acceso obligatorio y el control de acceso basado en roles en entorno Linux. Como segundo objetivo, hacer frente a la eventualidad de que los sistemas operativos "*trusted*" (confiables y seguros) comerciales dejaran de estar disponibles.

SELinux puede considerarse como una implementación práctica del modelo de seguridad de control de acceso obligatorio basado en el núcleo del sistema operativo Linux. Un administrador de un sistema SELinux tiene la posibilidad de configurar una política donde se definen los archivos a los que tiene acceso cada programa.

Para poder realizar esto, SELinux implementa un mecanismo para establecer en cada archivo y proceso el contexto en el que está siendo utilizado.

Mediante la utilización de un módulo del sistema operativo, es posible establecer reglas para permitir o denegar el acceso a cualquier archivo del sistema (utilizando el concepto de archivo de Unix, lo que incluye a los dispositivos, ficheros...).

El sistema operativo dispone de un proceso servidor de seguridad, que se ejecuta como parte del núcleo, que decide en base a la política de seguridad definida por el administrador, si algo (un proceso o un usuario) dispone de permiso para acceder a un objeto (archivo, dispositivo...). Este mecanismo de control se denomina *Type Enforcement* (TE).

Así, ante una incidencia de seguridad como puede ser un desbordamiento de búfer en un proceso ejecutado por `root`, el atacante sólo podrá acceder a los archivos para los cuales el proceso vulnerable esté autorizado por la política del sistema. No tendrá ningún efecto sobre el resto de archivos u objetos del sistema.

SELinux también permite implementar un modelo adicional de seguridad (*Multi-Level Security*, MLS) en el que además de lo indicado hasta ahora, es posible, para cada objeto, una capa de seguridad (como "altamente secreta", "secreta", "confidencial" y "sin restricción"). En este modelo, a los mecanismos descritos anteriormente se añade la restricción de que únicamente aquellos procesos y usuarios situados en la misma capa (o una capa superior) pueden acceder a los objetos de la misma capa o inferiores, pero nunca al revés. Así un usuario o un proceso de la capa "confidencial" puede acceder a la información "confidencial" y "sin restricción", pero nunca a la información marcada como "secreta" o "altamente secreta".

Puede encontrarse más información en:

`http://www.nsa.gov/selinux/index.cfm`

`http://fedora.redhat.com/projects/selinux/`

`http://fedora.redhat.com/docs/selinux-faq-fc3/`

26.5.2. Terminología SELinux

Pasamos ahora a introducir algunos de los conceptos básicos de SELinux. Se busca facilitar al alumno la comprensión posterior de documentos más detallados.

Como ya se ha comentado, mediante SELinux se definen una serie de políticas que controlan el acceso de forma que los usuarios únicamente obtienen los privilegios necesarios para realizar su trabajo. Así, es posible reducir o eliminar los daños producidos por posibles errores en la configuración o *buffer overflows*. Este mecanismo opera de forma independiente al tradicional mecanismo de control de acceso de Linux. No existe el concepto de superusuario o `root` ni los binarios setuid/setgid.

En el caso que la distribución de linux no tenga el soporte SELinux instalado es necesario instalar un kernel modificado que incluya estas funcionalidades.

SELinux proporciona compatibilidad con las aplicaciones linux existentes y con los módulos del kernel. Sin embargo, algunos módulos del kernel puede que requieran su modificación para interactuar de forma adecuada con SELinux. Las categorías de compatibilidad son:

- Compatibilidad de aplicación. SELinux proporciona compatibilidad con las aplicaciones existentes. No se han cambiado las estructuras de datos visibles por las aplicaciones ni el interfaz de las llamadas al sistema existentes, por lo que las aplicaciones correrán sin cambios si la política de seguridad autoriza la operación.
- Compatibilidad de módulos del kernel. Originalmente, SELinux solo proporciona compatibilidad para los módulos del kernel existentes, por lo que era necesario recompilar los módulos con las estructuras de seguridad necesarias. Ahora están integrados en el kernel 2.6, proporcionando compatibilidad binaria con los módulos existentes, con algunas excepciones que precisan modificaciones en los módulos.

SELinux utiliza un sistema de ficheros especial como parte de la instalación. Será necesario modificar `/etc/fstab` para que se monte de forma correcta.

```
none /selinux selinuxfs noauto 0 0
```

El sistema de ficheros `/selinux` es similar a `/proc`, es un pseudo sistema de archivos.

```
ls -l /selinux
total 0
-rw-rw-rw- 1 root root 0 Nov 25 11:27 access
-rw-rw-rw- 1 root root 0 Nov 25 11:27 context
-rw-rw-rw- 1 root root 0 Nov 25 11:27 create
-rw------- 1 root root 0 Nov 25 14:19 enforce
-rw------- 1 root root 0 Nov 25 11:27 load
-r--r--r-- 1 root root 0 Nov 25 11:27 policyvers
-rw-rw-rw- 1 root root 0 Nov 25 11:27 relabel
-rw-rw-rw- 1 root root 0 Nov 25 11:27 user
```

Si se ejecuta `cat /selinux/enforce` se obtendrá 1 ó 0 dependiendo de si estamos en modo forzado o permisivo respectivamente. Posteriormente se indicará que implica cada uno de estos modos.

En Debian y derivados, el directorio de configuración de políticas estará en `/etc/selinux`. En el caso de Fedora es `/etc/security/selinux/src/policy`.

A continuación veremos los conceptos con los que trabaja SELinux.

Identidad

Una identidad bajo SELinux no tiene el mismo significado que un uid. Ambos pueden convivir en el mismo sistema pero son diferentes. Las identidades para SELinux forman parte de un contexto de seguridad que definirá a qué dominios se puede acceder (basicamente, qué se puede hacer). Si se ejecuta el comando `su` no se cambiaría la identidad bajo SELinux.

Dominio

Cada proceso se ejecuta en un dominio. Un dominio determina el acceso que tiene un proceso, es una lista de qué procesos puede hacer o qué acciones puede realizar un proceso. En este caso sí hay similitud con el uid. Supongamos que `root` tiene un programa al que le ejecuta `chmod 4777` haciéndolo setuid `root`. Cualquiera en el sistema puede ejecutar este programa con privilegios de `root`, lo que representa un claro agujero de seguridad. Con SELinux si se tiene un proceso que hace una transición a un dominio privilegiado, si el *role* del proceso no lo autoriza a entrar entonces no se ejecutará.

Ejemplos de dominios son `sysadm_t` que es el dominio de administración del sistema y `user_t` que es el dominio general sin privilegios.

Tipo

Un tipo es asignado a un objeto y determina quién tiene acceso a este objeto. La definición para dominio es similar, excepto que un dominio se aplica a un proceso y el tipo se aplica a objetos como pueden ser directorios, ficheros, sockets, etc.

Role

Un role determina qué dominios pueden ser usados. Los dominios que un role de usuario puede acceder están predefinidos en los ficheros de configuración de políticas. Si un role no está autorizado para entrar en un dominio, se le negará el acceso.

Así, para permitir a un usuario del dominio sin privilegios `user_t` ejecutar el comando `passwd` será necesario especificarlo:

```
role user_r types user_passwd_t
```

De esta forma un usuario en el role `user_r` se le permite entrar en el dominio `user_passwd_t` donde puede ejecutar el comando `passwd`.

Contexto de seguridad

Un contexto de seguridad tiene todos los atributos que están asociados a cosas como ficheros, directorios, procesos, sockets TCP, entre otros. Un contexto de seguridad está formado por la identidad, role y dominio o tipo. Para verificar el contexto de seguridad actual se ejecuta `id` bajo SELinux.

En el caso de crear un fichero, el contexto de seguridad variará dependiendo del dominio que lo cree. Por defecto, el nuevo fichero hereda el mismo tipo que el directorio padre, aunque este comportamiento puede cambiarse con las políticas.

Si el usuario `legolas` crea un fichero con el nombre `prueba` en su directorio `$HOME`:

```
ls --context prueba
-rw-r--r-- legolas legolas legolas:object_r:user_home_t prueba
```

Si a continuación crea un fichero en `/tmp` llamado `tmpprueba`:

```
ls --context /tmp/tmpprueba
-rw-r--r-- legolas legolas legolas:object_r:user_tmp_t /tmp/tmpprueba
```

El tipo ha cambiado dependiendo del directorio donde fue creado el fichero.

Una forma de cambiar el contexto de seguridad es usando el comando `newrole -r role`, donde `role` es el nuevo role que quiere adoptarse. De esta forma, si un usuario quiere adoptar el role `sysadm_r`:

```
newrole -r sysadm_r
```

Será necesario proporcionar la clave para la identidad del usuario, la cual puede chequearse con el comando `id`. En caso de no tener autorización para entrar en el nuevo role:

```
newrole -r sysadm_r
legolas:sysadm_r:sysadm_t is not a valid context
```

Con este mensaje se indica que el usuario `legolas` no puede entrar en el role:dominio `sysadm_r:sysadm_t` debido a que no está autorizado.

Transición

Una transición determina qué contexto de seguridad será asignado a la operación solicitada. Hay dos tipos de transición:

- Transición del dominio de un proceso que es usado, cuando se ejecuta un proceso de un tipo especificado.
- Transición de un tipo de fichero, cuando se crea un fichero bajo un directorio en particular.

Una transición de tipo es lo que se vio en el ejemplo anterior.

Políticas

Las políticas están constituidas por un conjunto de reglas que definen cosas como los roles a los que un usuario tiene acceso. Estas reglas se editan conforme a como se desee que se defina la seguridad del sistema.

26.5.3. Modos de uso de SELinux

El **modo Permisivo** es el utilizado cuando el servidor se dedica a guardar información en ficheros de log referente a SELinux. No se siguen las reglas definidas en las políticas, simplemente se almacenan los eventos que se producen relativos a SELinux. Este modo es el adecuado para depuración ya que se pueden repasar los mensajes generados y verificar que la configuración es correcta.

El otro modo existente es el **modo Reforzado**. En este modo el sistema sigue las políticas que se hayan definido en SELinux. Hay que tener cuidado cuando se active este modo, si hay algún fallo en la configuración de las políticas puede perserse parte del acceso al sistema.

Para poder pasar de un modo a otro es necesario que esté definida la opción `CONFIG_SECURITY_SELINUX_DEVELOP` en la compilación del kernel. Posteriormente, para pasar del modo permisivo al reforzado es necesario ejecutar:

```
echo "1" > /etc/selinux/enforce
```

En caso contrario se sustituye `1` con `0`. Esto proporciona un método para saber en que modo está el sistema. Basta con verificar el valor de `/etc/selinux/enforce`.

Si se compila el kernel con el modo desarrollo activado el servidor arrancará con el modo permisivo. Este comportamiento puede modificarse creando un script que se ejecute en el arranque y que cambie de modo pasando el parámetro `enforcing=1` al kernel durante el arranque.

Capítulo 27

Vulnerabilidades del sistema

> Los favorables al Open Source defienden que la naturaleza del software de fuentes abiertas lo hace más seguro. Los críticos al movimiento Open Source defienden que el software abierto es menos seguro.
>
> *Hacking Exposed Linux*

27.1. Tipos de ataques y vulnerabilidades

27.1.1. Ataques de negación de servicio (*denial of service*)

El ataque denominado *Denial of Service*[1] o de negación de servicio es un tipo de ataque cuya meta fundamental es la de negar el acceso del atacado a un recurso determinado o a sus propios recursos. Los ataques de negación de servicio pueden dejar inoperativo un servidor o una red. De esta forma, toda una organización puede quedar fuera de Internet durante un tiempo determinado. Algunos ejemplos de este tipo de ataque son:

- Intentos de inundar (*flood*) una red, evitando de esta manera el tráfico de datos en la misma.
- Intentos de interrumpir las conexiones entre dos máquinas, para evitar el acceso a un servicio por parte del resto de usuarios.
- Intentos de evitar que un determinado usuario tenga acceso a un servicio.
- Intentos de interrumpir un servicio específico a un sistema o a un usuario.

También hay que tener en cuenta que el uso ilegítimo de recursos puede dar lugar, igualmente, a la negación de un servicio. Por ejemplo, un intruso puede utilizar un área del FTP anónimo como lugar para salvar archivos, consumiendo, de esta manera, espacio en el disco y generando tráfico en la red.

Algunos ataques de negación de servicio se pueden ejecutar con recursos muy limitados contra un sitio grande y sofisticado. Este tipo de ataque se denomina ataque asimétrico. Por ejemplo, un atacante con un ordenador anticuado y un módem puede poner fuera de combate a máquinas rápidas y sofisticadas.

Hay tres tipos de ataques básicos de negación de servicio:

- Consumo de recursos escasos, limitados, o no renovables
- Destrucción o alteración de información de configuración
- Destrucción o alteración física de los componentes de la red

[1] También se hace referencia a él como ataque DoS.

Ataques que podemos sufrir

Los ordenadores y las redes necesitan para funcionar ciertos recursos: ancho de banda de la red, espacio de memoria y disco, tiempo de CPU, estructuras de datos, acceso a otros ordenadores y redes, entre otros.

Los ataques de negación de servicio se ejecutan, con frecuencia, contra la conectividad de la red. La meta del atacante es evitar que los ordenadores se comuniquen en la red. La inundación o *flooding* es uno de las formas más antiguas de ataques DoS a internet. En la actualidad el ancho de banda de la red es tan importante como cualquiera de los elementos físicos del servidor. Los ataques DoS de inundación de red pretenden evitar que los ordenadores que forman nuestra red puedan comunicarse. El ataque comienza con un proceso de establecimiento de conexión con la máquina objetivo, uno de los ejemplos más claros es el ataque `SYN` *flood*. Veamos con un poco de detalle en qué consiste este ataque.

Cuando un sistema (cliente) intenta establecer una conexión TCP con el sistema que proporciona servicios (servidor), tanto el cliente como el servidor van a intercambiar un conjunto de mensajes. Esta técnica de conexión se aplica a todas las conexiones TCP. El cliente comienza enviando un mensaje `SYN` al servidor. A su recepción, el servidor da su reconocimiento al mensaje enviando un mensaje `SYN-ACK`. El cliente finaliza la conexión respondiendo con un mensaje `ACK`. Es en este momento en el que se establece la comunicación entre cliente y servidor[2].

El potencial ataque tiene lugar en este punto donde el servidor ha enviado un reconocimiento `SYN-ACK` de vuelta al cliente, pero aún no ha recibido el `ACK`. El servidor ha construido en memoria una estructura para almacenar todas las conexiones pendientes. Esta estructura tiene un tamaño limitado y puede excederse creando muchas conexiones como la descrita. El atacante envía muchos mensajes `SYN` haciéndose pasar por clientes desconocidos que no van a responder. Logrará de esta manera que el servidor no acepte más conexiones entrantes cuando sature la estructura de datos donde almacena las comunicaciones pendientes de confirmación.

Debemos tener en cuenta que este tipo de ataque no depende del ancho de banda que disponga el atacante. En este caso, el atacante está consumiendo las estructuras de datos del kernel, implicadas en establecer una conexión TCP. Un atacante con una simple conexión telefónica puede realizar este ataque contra un servidor de los más potentes (éste último es un buen ejemplo de un ataque asimétrico).

Un atacante también puede utilizar los recursos que disponemos contra nosotros mismos, de maneras inesperadas. Por ejemplo, el caso de DoS UDP. En este ataque, el atacante utiliza los paquetes falsificados de UDP para conectar el servicio de generación de *echo* en una máquina con el servicio de *chargen* en otra máquina. El resultado es, que los dos servicios consumen todo el ancho de banda de red entre ellos. Así, la conectividad para todas las máquinas en la misma red desde cualquiera de las máquinas atacadas se ve afectada.

Un atacante puede, también, consumir todo el ancho de banda disponible en su red generando una gran cantidad de paquetes dirigidos a la misma. Típicamente, estos paquetes son de generación de *echo* de `ICMP` (`ping`), pero pueden ser cualquier otra cosa. Además, el atacante no necesita operar desde una sola máquina; podría coordinar varias máquinas en diversas redes para alcanzar el mismo efecto.

Además del ancho de banda de la red, los atacantes pueden consumir otros recursos que nuestro sistema necesite para funcionar. Por ejemplo, en muchos sistemas, un número limitado de las estructuras de datos en el kernel está disponible para almacenar información de procesos (identificadores, entradas en tablas de procesos, *slots* , etc.). Si se consumen estas estructuras de datos escribiendo un programa o un script que no haga nada, pero que cree en varias ocasiones copias de sí mismo conseguiríamos un ataque DoS. Muchos sistemas operativos modernos, aunque no la totalidad de ellos, tienen recursos para protegerse contra este problema.

Un atacante puede también consumir su espacio en disco de otras maneras, por ejemplo:

- Generar miles de correos (*Spam*, *Bombing*).

[2] Lo que conocemos como *Three way handshake* o establecimiento de conexión en tres pasos.

- Generar intencionalmente errores que deben ser reflejados en los ficheros de log. En este tipo de ataque, podemos citar también la utilización indebida de `syslog`. Es decir, utilizar el proceso `syslog` de la víctima para que registre eventos de otra máquina, llenando el espacio en disco.
- Colocar archivos en su disco, utilizando `ftp` anónimo.

En general, se puede utilizar cualquier cosa que permita que los datos sean escritos en el disco para ejecutar un ataque DoS si no hay límites en la cantidad de datos que se pueden escribir (*quotas*).

No obstante, muchos sitios tienen esquemas de *lockout* de cuenta después de un cierto número de *logins* fallados. Una configuración típica bloquea el *login* después de 3 a 5 intentos fallidos. Un atacante puede utilizar este esquema para evitar que los usuarios legítimos entren. En algunos casos, incluso las cuentas privilegiadas, tales como `root`, pueden ser víctimas de este tipo de ataque. Es fundamental disponer siempre de un método para acceder ante la emergencia de este tipo de ataques.

Algunas veces, errores en la programación del kernel de Linux pueden llevarnos a un DoS:

- Ping de la muerte (*ping of death*). Algún software permite enviar paquetes ICMP que son mayores de 65.536 bytes, valor máximo que la especificación TCP/IP permite. En la actualidad, sólo algunas pilas TCP/IP son vulnerables a este ataque y la mayoría de los routers de internet filtrarán los paquetes de este tamaño.
- *Teardrop*. Es similar al ping de la muerte, pero en este caso intenta romper la pila de la red destino proporcionándole múltiples fragmentos que no se reensamblan de forma adecuada. El resultado es un *kernel panic* y el consiguiente reinicio de la máquina.
- *Deep simlink bug*. En algunas versiones del kernel, incluyendo hasta la 2.4.9 hay un error que permite realizar un DoS local. Un atacante puede crear un directorio que contenga múltiples directorios y enlaces simbólicos que referencien a ellos mismos de forma repetida. Cuando uno de los ficheros es leído, el kernel consume un periodo largo de tiempo intentando resolver dónde se encuentra el fichero original. Durante dicho periodo ningún otro proceso puede ejecutarse y la máquina se bloquea.

Hay otros componentes que pueden ser vulnerables a la negación de servicio y que deben vigilarse. Estos incluyen:

- Impresoras
- Unidades de cinta
- Conexiones de red
- Otros recursos limitados importantes para la operación del sistema.

Un ordenador incorrectamente configurado puede no funcionar bien, o directamente no arrancar. Un atacante puede alterar o destruir la información de configuración del sistema operativo, evitando de esta forma que pueda usarse el ordenador atacado. Veamos algunos ejemplos:

- Si un atacante puede cambiar la información de enrutado de sus *routers*, la red puede ser deshabilitada.
- Si un atacante puede modificar cualquier fichero de configuración del sistema, de los existentes en `/etc` el sistema puede no arrancar. Supongamos que se borran las entradas de `/etc/fstab`.

También es muy importante la seguridad física de la red. Se debe resguardar contra el acceso no autorizado a los ordenadores, los *routers*, los *racks* de cableado de red, los segmentos del *backbone* de la red, y cualquier otro componente crítico de la red.

Prevención y respuesta a los ataques

Tal como se ha expresado anteriormente, los ataques DoS pueden dar lugar a pérdidas significativas de tiempo (y dinero) para muchas organizaciones, por lo que se recomiendan una serie de medidas:

- Colocar listas de control de acceso en los *routers*. Esto reducirá su exposición a ciertos ataques DoS
- Instalar parches a su sistema operativo contra *flooding* de `TCP SYN`. Esta acción permitirá reducir sustancialmente la exposición a estos ataques, aunque no pueda eliminar el riesgo de forma definitiva.
- Invalidar cualquier servicio de red innecesario o no utilizado. Esto puede limitar la capacidad de un atacante de aprovecharse de esos servicios para ejecutar un ataque DoS.
- Implementar sistema de cuotas.
- Observar el funcionamiento del sistema y establecer valores base para la actividad ordinaria. Utilizar estos valores para calibrar niveles inusuales de la actividad del disco, del uso de la CPU, o del tráfico de red.
- Incluir, como parte de su rutina, el examen de su seguridad física. Considerar, entre otras cosas, los servidores, *routers*, terminales desatendidos, puertos de acceso de red y los *racks* de cableado.
- Utilizar Tripwire o una herramienta similar para detectar cambios en la información de configuración u otros archivos.
- Utilizar configuraciones de red redundantes y tolerantes a fallos.

27.1.2. Cracking de passwords

En este apartado, se presentarán una serie de consideraciones referidas al *cracking* de *passwords* basadas en UNIX. El objetivo inicial consiste en entrar al servidor. Debido a que se permite el acceso a múltiples usuarios, los sistemas UNIX nos solicitarán un nombre de identificación acompañado de una clave. Dicho nombre deberá darse de alta en el sistema para que se pueda acceder.

Cuando un usuario desea entrar en una máquina, el sistema solicitará un *login* de acceso o nombre de usuario. Si el *login* es incorrecto, el sistema no lo notificará para impedirle conocer qué accesos se encuentran dados de alta. Si la *password* coincide con la que tiene asignada el *login* que se emplea, el sistema permitirá el acceso.

Una vez encriptada una password, no se puede desencriptar. Sin embargo, esto no garantiza la seguridad de la *password*, puesto que no significa que la *password* no se pueda averiguar.

El mecanismo que se utiliza para descubrir (no desencriptar) las *passwords* consiste en efectuar encriptaciones de palabras (posibles *passwords*) y comparar estas encriptaciones con el original.

¿De que depende el éxito? El éxito depende de la calidad del diccionario (archivo que contiene un conjunto de posibles *passwords*), del programa que se utilice, de la CPU y, por supuesto, de nuestra paciencia. Los programas buscadores de contraseñas son fácilmente diseñables. Si mediante un *bug* se obtiene el archivo `/etc/passwd`, se puede iniciar un ataque de diccionario contra el mismo obteniéndose, de este modo, las *passwords*.

Otro tipo de ataque es el de fuerza bruta, que consiste simplemente en realizar todas la combinaciones posibles de caracteres hasta hallar la password.

27.1.3. E-mail bombing y spamming

En este apartado, se presentarán algunas de las dificultades que pueden surgir como consecuencia de la utilización de los servicios de *mail*. Se brindarán, por otro lado, algunas respuestas a dichos obstáculos.

El *e-mail bombing* consiste en enviar muchas veces un mensaje idéntico a una misma dirección, saturando el *mailbox* del destinatario. El *spamming*, que es una variante del *e-mail bombing*, se refiere a enviar el *email* a centenares o millares de usuarios e, inclusive, a listas de interés. El *spamming* puede resultar aún más perjudicial si los destinatarios contestan el *mail*, haciendo que todos reciban la respuesta.

Puede, además, ocurrir inocentemente como resultado de enviar un mensaje a la lista y no darse cuenta de que la lista lo distribuye a millares de usuarios, o como resultado de mala configuración de un autorespondedor, por ejemplo el vacation[3].

El *e-mail bombing/spamming* se puede combinar con el *e-mail spoofing*[4], logrando que sea más difícil determinar quién está enviando realmente el *mail*.

Cuando se proveen los servicios de *e-mail* los usuarios son, lógicamente, vulnerables al *e-mail bombing* y *spamming*. En efecto, el *e-mail spamming* es casi imposible de prevenir. Un usuario con una dirección válida de *mail* puede realizar *spam* a cualquier otra dirección de *mail* o *newsgroup*.

Cuando gran cantidad de *mails* son dirigidos a un solo sitio, éste puede sufrir DoS por pérdida de conectividad, caerse el sistema o producirse fallos en el servicio debido a:

- sobrecarga de conexiones de red
- utilización de todos los recursos de sistema disponibles
- llenado del disco como resultado de *postings* múltiples y de entradas en el *syslog*.

Si el sistema aparece repentinamente lento (el *e-mail* está lento o no parece que se envía o recibe), la razón puede ser que el servidor de correo está intentando procesar una excesiva cantidad de mensajes. Esto puede comprobarse a través del log de sistema.

Es importante:

- Identificar la fuente del *e-mail bomb/spam* y configurar el *router* para evitar el acceso de los paquetes entrantes de esa dirección. Puede colocarse una lista de control de acceso[5] en el puerto 25 (`SMTP`) para esa dirección.
- Observar las cabeceras del *e-mail* para determinar su origen verdadero.
- Ponerse en contacto con el sitio que se identifique como origen con el propósito de alertarlo de la actividad del *spammer*.
- Asegurarse de tener la versión más actualizada del demonio de *mail* (por ejemplo Sendmail) y aumentar el grado de *debug* o *log* que posea el proceso, para detectar o alertar sobre estas actividades. Vigilar el tamaño del archivo de log, ya que puede crecer considerablemente, si se está bajo un *e-mail bombing*.

Desafortunadamente, hasta el momento, no hay manera de prevenir el bombardeo de *e-mail* o *spamming* y es imposible predecir el origen del ataque siguiente. Es trivial obtener acceso a listas de interés o acceder a información que contenga grandes volúmenes de direcciones de *e-mail*, las que proporcionan al atacante direcciones de destino para el *spam*.

Pueden desarrollarse herramientas internas, que pueden ayudar a reconocer y a responder al *e-mail bombing/spamming* reduciendo, de esta manera, el impacto de tal actividad. Tales herramientas deben aumentar las capacidades de log y alertar de mensajes que vienen de un mismo

[3] Responde con un mensaje automáticamente diciendo que estamos de vacaciones o no podemos atender el correo por algún tiempo determinado.

[4] Que altera la identidad de la cuenta que envía el mail

[5] Ver el apartado sobre `tcp-wrappers` de las entregas anteriores.

lugar en un corto período de tiempo. Asimismo, deberían ser capaces de rechazar esos mensajes, o descartarlos.

Si un sitio utiliza un número pequeño de servidores de *e-mail*, podría configurarse un *firewall* para asegurarse de que las conexiones de *smtp* fuera del *firewall* puedan hacerse solamente a sus *pasarelas* de *mail* y a ninguno de los otros equipos. Aunque esta operación no prevendrá un ataque, reduce al mínimo el número de las máquinas disponibles para un ataque basado en SMTP. De este modo, se puede controlar el tráfico entrante SMTP y filtrarlo de manera acorde.

Es importante no contestar y/o hacer un *forward (reenvío)* de los *spams*. De este modo evitaremos que el problema se propague.

27.1.4. Seguridad en WWW

A continuación vamos a tratar algunos aspectos relativos a la seguridad de un servidor web. Nos centraremos en Apache, tratado en una entrega anterior. No hay que confundir la seguridad del servidor web con la de los clientes web (Netscape, Internet Explorer, ...). Los problemas que puedan originarse en unos y otros no tienen nada que ver.

Cuando accedemos a un link a través del navegador web, realizamos una conexión TCP/IP al servidor donde residen las páginas. Esta conexión se suele realizar a través del puerto 80 (HTTP). El navegador envía un mensaje denominado (HTTP request) al servidor y éste responde con los datos solicitados. Podemos también efectuar una conexión mediante telnet en lugar de con un navegador web

```
telnet www.midomio.org 80
```

Supongamos que utilizamos la utilidad `curl`[6]:

```
# curl --head http://www.midominio.org
HTTP/1.1 302 Found
Date: Sat, 04 Jun 2005 16:51:07 GMT
Server: Apache/2.0.54 (Debian GNU/Linux) PHP/4.3.10-15 mod_ssl/2.0.54
    OpenSSL/0.9.7e mod_perl/1.999.21 Perl/v5.8.4
Location: http://www.midominio.org/
Content-Type: text/html; charset=iso-8859-1
```

Un intruso podría obtener de esta manera información sobre la versión de Apache que se encuentra corriendo en el servidor. Sólo tendrá que encontrar un posible *exploit*[7] de esa versión o esperar a que aparezca alguno. Una manera de evitar esto, es modificar la información que aparece en la cabecera que muestra el servidor web Apache[8].

La directiva de Apache `ServerTokens` tiene esta misión:

- `ServerTokens Full`: Muestra todos los nombres de módulos y versiones del servidor
- `ServerTokens Min`: Muestra únicamente el nombre de los módulos y del servidor
- `ServerTokens ProductOnly`: Muestra únicamente el nombre del servidor, en nuestro caso Apache

Este tipo de solución se denomina "seguridad por oscuridad". Se trata de evitar proporcionar información a cualquier usuario y en concreto a los que tienen intenciones no muy recomendables, sobre el software y versiones utilizados.

[6] Para poder disponer de ella:

```
#apt-get install curl
```

Para mayor información sobre esta utilidad consultar el manual `man curl`

[7] Utilizado en el argot para nombrar a un programa que aprovecha un fallo y que permite sacar partido de él.

[8] Estudiado en la 3ª entrega del curso.

Otra recomendación de seguridad es estar informado de las posibles vulnerabilidades que aparecen de los servidores web Apache. Si tenemos una versión con una vulnerabilidad que se soluciona en una versión posterior, debemos actualizar la versión lo antes posible.

Es importante también controlar los datos que están disponibles. Por defecto, todos los datos tienen acceso por parte de cualquier persona que acceda al servidor mediante un navegador web. Cualquier tipo de datos que requieran un acceso restringido deberán configurarse para que se cumpla esa condición. Utilizaremos las directivas de Apache que restringen el acceso a determinadas direcciones IP o a determinados usuarios.

Hasta aquí hemos tratado recomendaciones referentes a la configuración de Apache. Sin embargo, no debemos olvidar la seguridad desde el punto de vista del sistema operativo. Si dentro de un directorio del `DocumentRoot` de Apache tenemos un enlace simbólico a `/etc`, cualquier usuario de internet podrá tener acceso al fichero `/etc/password`. Únicamente tendrá que escribir la URL de forma adecuada y obtendrá en su navegador el fichero de password. Es recomendable el uso de las opciones de Apache `FollowSymLinks` y `SymLinkIfOwnerMatch`.

Permitir el índice de los directorios tampoco es una buena idea, ya que el intruso tendrá información de la estructura de directorios del servidor web. Podría encontrar cualquier fichero que hayamos dejado por olvido y que tenga información sensible. Podemos controlar este comportamiento con `Option Indexes`.

En el caso de usar los mecanismos de autenticación proporcionados por Apache, para limitar el acceso a determinadas áreas de los contenidos del servidor web, tendremos que ser igualmente cuidadosos. Normalmente usaremos el fichero `.htaccess`, restringiendo al máximo el acceso al mismo. No podemos permitir que un intruso, por el método de "prueba y error" construya URLs en las que busque obtener el fichero `.htaccess`. Para ello es conveniente utilizar:

```
<Files .htaccess>
     Order allow, deny
     Deny from all
</Files>
```

Otra opción es configurar el acceso restringido en el propio `httpd.conf`, con lo que evitaríamos la circunstancia anterior.

Existe otro punto que debemos controlar y son los *scripts* y *cgi-bin*. Atacar el sistema operativo vía internet implica buscar algún error en un *script cgi* o lograr que el servidor web haga algo para lo que no fue pensado, como por ejemplo dar al intruso acceso al *shell* del servidor, que ese intruso ejecute comandos arbitrarios en él , o consiga información útil para lograr esos objetivos.

Es obvio que los datos proporcionados a cualquier *cgi script* a través de un formulario, deben ser probados para su validez por una razón u otra, y una de esas razones indudablemente es la seguridad. Dependiendo de lo que el *script* vaya a hacer, la entrada aparentemente inocua de información puede tener graves consecuencias.

Por ejemplo, consideremos el siguiente script en `perl` en el cual se realiza un `finger` al usuario que se indicó en el campo de entrada del formulario y devuelve los resultados al navegador web:

```
#!/usr/local/bin/perl
$|=1;
require 'cgi-lib.pl';
&ReadParse;
Script poco seguro
print &PrintHeader;
open(IN, "/usr/bin/finger $in{'user_id'} |");
@stuff=;
foreach(@stuff) { print; }
exit;
```

Si porporcionamos la siguiente entrada:

```
legolas; /bin/cat / etc/passwd
```

En caso de no poseer soporte de *shadow passwords*, tendremos graves problemas. El *script* anterior constituye un ejemplo muy básico de la forma que un *password grab*[9] podría tomar. El origen del problema radica en que la cadena de entrada podría contener cualquier comando arbitrario. El ejemplo anterior no controló si la entrada en el formulario era un usuario o una "bomba atómica".

Otro aspecto a tener en cuenta es que durante los últimos años, en los cuales se ha extendido el uso de documentos dinámicos, otras vulnerabilidades han entrado en escena. El uso de los *Server Side Includes* (SSIs), en algunos casos significó una extensión nueva de archivo, como `shtml`, en otros significó permitir SSIs para cada documento en el servidor o en un árbol dado del documento. En cualquier caso, permitir SSIs permite un `exec`. Un uso legítimo típico de una etiqueta `exec` es:

```
Esta ápgina ha sido visitada <!-- #exec cgi="/cgi-bin/counter " --> veces
```

Pero imaginemos un sistema de mensajería de alguna clase basado en HTML, por ejemplo un libro de visitas que toma la entrada y construye un documento HTML. Alguien entra y deja:

```
Hey! Que ápginas áms bonitas, évolver pronto!
<!-- #exec cmd="/bin/cat /etc/passwd " -->
```

Si no se están analizando los campos que introducimos, nuevamente tenemos un *password grab*. O podría introducirse cualquier cosa que el servidor pudiera ejecutar, lo cual sería fatal para nuestro sistema si el servicio web se ejecuta como `root`.

Las últimas versiones de Apache proporcionan como opción invalidar los SSIs de tal manera que se pueden habilitar sin el `exec`. Muchos de estos problemas se pueden reducir permitiendo el *chrooting*[10] del servidor web, aunque a pesar de los aumentos que se hacen de seguridad, éstos no son de ninguna forma un ejercicio trivial.

27.2. Analizador de vulnerabilidades Nessus

La herramienta Nessus `http://www.nessus.org` es un proyecto que proporciona a la comunidad de internet una herramienta de análisis de vulnerabilidades fácil de usar, gratuita y muy potente. Nessus es un programa de dominio público desarrollado bajo licencia GPL que permite automatizar la comprobación de forma remota de los posibles agujeros de seguridad por parte de los administradores de sistemas, determinando así por qué sitios pueden acceder los intrusos. Se diseñó para ayudar a identificar y resolver los problemas conocidos de los diferentes servicios que corren en un servidor, permitiendo adelantarnos a los intrusos.

A diferencia de otros analizadores de vulnerabilidades, Nessus no supone nada, es decir, no considerará que un servicio está a la escucha en un puerto específico. Si nuestro servidor web está a la escucha en el puerto 2004, Nessus detectará que hay corriendo un servidor web en ese puerto y buscará las posibles vulnerabilidades del servicio. Del mismo modo, tampoco supondrá que existe una vulnerabilidad a partir de la versión del software que proporciona el servicio, sino que intentará explotarla.

Nessus tiene una arquitectura modular cliente/servidor, lo que permite tener una máquina que realiza los chequeos de seguridad (servidores) y la interfaz gráfica en varias estaciones de trabajo (clientes). Los servidores que se van a chequear no es necesario que ejecuten ningún software adicional y pueden chequearse al mismo tiempo tantos como queramos.

Otro de los puntos fuertes de Nessus es su actualización constante de la base de datos de vulnerabilidades. Todos los nuevos chequeos de seguridad en forma de scripts, se pueden encontrar en `http://www.nessus.org/scripts.php`.

[9] Recolector de passwords

[10] Consiste en limitar el `DocumentRoot` a una especie de partición dentro del sistema de archivos en el que se encuentra. Algo así como una jaula de la que no puede salir.

Los informes que proporciona Nessus a la finalización de un escaneo en busca de vulnerabilidades pueden grabarse en disco en distintos formatos (XML, HTML, Texto, ...) y ofrecen detalles sobre las vulnerabilidades encontradas, así como las referencias de información al respecto.

27.2.1. Instalación de Nessus

La instalación de Nessus es bastante simple. Es recomendable, aunque no estrictamente necesario, la instalación[11] de varios programas externos que aumentan la potencia de Nessus. Estos programas son:

- NMAP (escaneador de puertos)
- Hydra[12] (chequeador de password sencillo)
- Nikto (chequeador de cgi/script)

Nos referimos a ellos porque son los mejores en su categoría. Si se instalan en su ruta por defecto, durante el proceso de instalación de Nessus, estarán disponibles de forma automática.

Los requerimientos referentes a paquetes adicionales necesarios para instalar Nessus son:

- Los paquetes `shareutils`, `bison`, `flex`
- El compilador gcc instalado, ya que se realizará la compilación de Nessus
- La librería `gtk` para el cliente gráfico

La instalación de Nessus se puede hacer de tres formas, siempre con el usuario `root`:

A partir de los paquetes: es la más sencilla para Debian, ya que sólo hay que ejecutar:

```
#apt-get install nessus
```

y en su caso, los paquetes `nessus-plugins` y `nessusd`

Compilación manual: Se bajarán los fuentes de las distintas partes que componen Nessus y se compilarán en el orden que se indica en la página web de Nessus.

Compilación automática: Se baja un único paquete que descomprime los paquetes con los fuentes y realiza el proceso de compilación e instalación.

Optaremos por documentar la tercera opción, ya que la primera no presenta mayor problema.

Bajaremos el archivo `nessus-installer.sh` de `http://www.nessus.org/nessus_2_0.html`

```
                    NESSUS INSTALLATION SCRIPT

Welcome to the Nessus Installation Script !
This script will install Nessus 2.0.10a (STABLE) on your system.
Please note that you will need root privileges at some point so that
the installation can complete.
Nessus is released under the version 2 of the GNU General Public License
(see http://www.gnu.org/licences/gpl.html for details).
To get the latest version of Nessus, visit http://www.nessus.org
Press ENTER to continue
x - creating lock directory
```

[11] `#apt-get install nmap nikto`

[12] `http://thc.org/thc-hydra/`

```
x - extracting nessus.tar.gz (binary)
x - now extracting this archive
x - done
-------------------------------------------------------------------------------
Nessus installation : installation location
-------------------------------------------------------------------------------
Where do you want the whole Nessus package to be installed ?
[/usr/local]
-------------------------------------------------------------------------------
Nessus installation : Ready to install
-------------------------------------------------------------------------------
Nessus is now ready to be installed on this host.
The installation process will first compile it then install it
Press ENTER to continue
-------------------------------------------------------------------------------
Nessus installation : Finished
-------------------------------------------------------------------------------
Congratulations ! Nessus is now installed on this host
. Create a nessusd certificate using /usr/local/sbin/nessus-mkcert
. Add a nessusd user use /usr/local/sbin/nessus-adduser
. Start the Nessus daemon (nessusd) use /usr/local/sbin/nessusd -D
. Start the Nessus client (nessus) use /usr/local/bin/nessus
. To uninstall Nessus, use /usr/local/sbin/uninstall-nessus
. Remember to invoke 'nessus-update-plugins' periodically to update your
  list of plugins
. A step by step demo of Nessus is available at :
        http://www.nessus.org/demo/
Press ENTER to quit
```

Una vez instalado, pasaremos a realizar los pasos que se nos indican. Cuando hemos instalado el servicio de Nessus, es necesario llevar a cabo una serie de sencillos pasos. La primera tarea es generar un certificado para encriptar el tráfico entre el cliente y el servidor. El comando `nessus-mkcert` se encarga de realizar esta función.

```
[root@fedora root]# nessus-mkcert
/usr/local/var/nessus/CA created
/usr/local/com/nessus/CA created
-------------------------------------------------------------------------------
                        Creation of the Nessus SSL Certificate
-------------------------------------------------------------------------------
This script will now ask you the relevant information to create the SSL
certificate of Nessus. Note that this information will *NOT* be sent to
anybody (everything stays local), but anyone with the ability to connect to
    your
Nessus daemon will be able to retrieve this information.
CA certificate life time in days [1460]:
Server certificate life time in days [365]:
Your country (two letter code) [FR]: ES
Your state or province name [none]: Sevilla
Your location (e.g. town) [Paris]: Sevilla
Your organization [Nessus Users United]: Mi Organizacion
-------------------------------------------------------------------------------
                        Creation of the Nessus SSL Certificate
-------------------------------------------------------------------------------
Congratulations. Your server certificate was properly created.
/usr/local/etc/nessus/nessusd.conf updated
The following files were created :
. Certification authority :
```

```
    Certificate = /usr/local/com/nessus/CA/cacert.pem
    Private key = /usr/local/var/nessus/CA/cakey.pem
. Nessus Server :
    Certificate = /usr/local/com/nessus/CA/servercert.pem
    Private key = /usr/local/var/nessus/CA/serverkey.pem
Press [ENTER] to exit
```

La siguiente tarea para completar la instalación, es añadir un usuario. Puede añadirse mediante el comando `nessus-adduser`. Este script preguntará qué método de autenticación queremos usar, el recomendado (y más simple) es el de password. La siguiente pregunta que se nos hace es referente a las reglas para restringir la cuenta del usuario. Podemos restringir a un determinado usuario para que sólo pueda realizar escaneos de determinadas IP. El usuario que creemos será un usuario propio de Nessus y no tendrá reflejo en el sistema.

```
[root@fedora root]# nessus-adduser
Using /var/tmp as a temporary file holder
Add a new nessusd user
----------------------

Login : nessus
Authentication (pass/cert) [pass] :
Login password : nessus
User rules
----------

nessusd has a rules system which allows you to restrict the hosts
that nessus has the right to test. For instance, you may want
him to be able to scan his own host only.
Please see the nessus-adduser(8) man page for the rules syntax
Enter the rules for this user, and hit ctrl-D once you are done :
(the user can have an empty rules set)
Login                : nessus
Password             : nessus
DN                   :
Rules                :
Is that ok ? (y/n) [y] y
user added.
```

Una vez creado el usuario de Nessus tenemos la infraestructura preparada para poder arrancar las partes de que se compone Nessus. Arrancaremos el demonio `nessusd` y a continuación el cliente gráfico. Supondremos que tanto el cliente gráfico como el demonio han sido instalados en la misma máquina.

27.2.2. Actualización de plugins

Antes de realizar un chequeo de un servidor, es recomendable actualizar los plugins que tenemos instalados. Los plugins de Nessus son como las firmas de virus para los antivirus. Cada uno está hecho para una vulnerabilidad específica, explotando la vulnerabilidad en cuestión, o simplemente comprobando versiones de software que son vulnerables. Normalmente están escritps en NAS (*Nessus Attack Scripting Language*) que es un lenguaje propio de Nessus, aunque pueden ser escritos en casi cualquier lenguaje de programación. La actualización de estos plugins debe ser hecha frecuentemente al descubrirse nuevas vulnerabilidades prácticamente todos los días.

El script `nessus-update-plugins` buscará los nuevos scripts que detectan nuevas vulnerabilidades. Este script hará uso de las utilidades `lynx`, `tar` y `gzip`.

`nessus-update-plugins [-v] [-r <pluginname>] [-h] [-i <pluginname>]`

Para más detalles, puede verse la entrada en el manual de sistema (`man nessus-update-plugins`).

En caso de que la máquina donde está instalado `nessusd` esté situada detrás de un proxy, será necesario crear el archivo `.nessus-update-pluginsrc` en el directorio `$HOME` del usuario que esté ejectando el script. Incluiremos las siguientes líneas en este fichero:

```
proxy_user= username
proxy_passwd= password
proxy= address_of_your_proxy
```

Un ejemplo de un fichero sería:

```
proxy_user=hugo
proxy_passwd=topsecr3t
proxy=proxy.miordenador.es:3128
```

27.2.3. Arrancando Nessus

Una vez que tenemos Nessus correctamente instalado y configurado y con los últimos plugins instalados, podemos arrancar el demonio. La forma más simple es arrancar con el usuario root el demonio `nessusd`

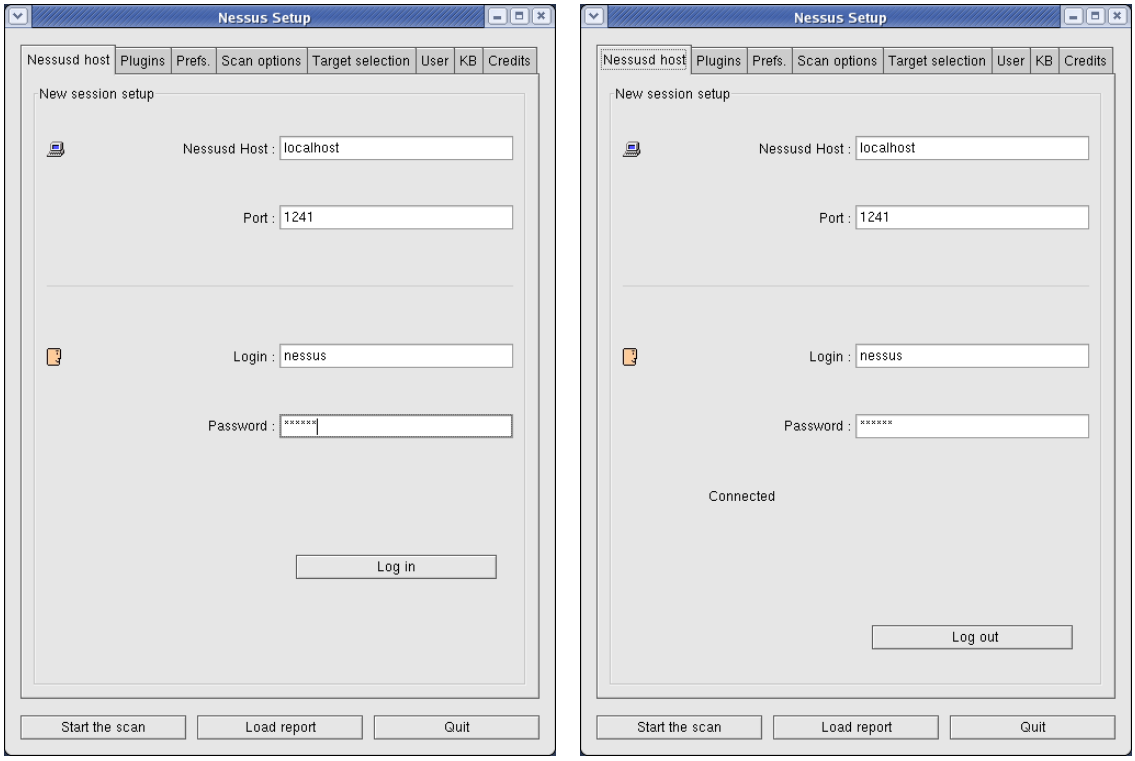
```
nessusd -D
```

A continuación, será necesario utilizar un cliente, instalado en el mismo servidor o en otro ordenador, para conectarnos y comenzar los escaneos. Existe también la posibilidad de prescindir de la interfaz gráfica y utilizar Nessus desde la línea de comandos. El cliente gráfico se arranca:

```
nessus
```

Una vez arrancado, es necesario conectarnos al servidor donde se encuentra arrancado `nessusd`. Es necesario proporcionar la dirección IP del servidor Nessus, así como el usuario y clave con el que nos conectaremos.

Figura 27.1: Acceso de usuario en Nessus

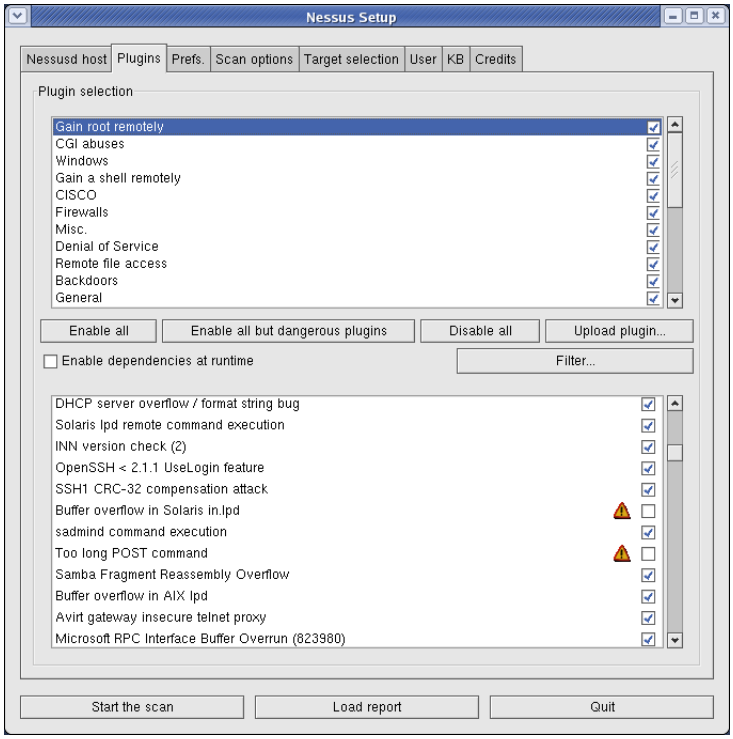


El cliente se conecta al servidor a través de una conexión SSL, cargando la lista de los plugins que tiene instalados. La primera vez que lo ejecutemos, el certificado pedirá una confirmación del mismo antes de ser descargado. Esto asegura que en el futuro las comunicaciones se realizarán con este servidor.

27.2.4. Usando Nessus

Como ya hemos indicado, uno de los aspectos que destacan a Nessus es la cantidad de plugins. Dependiendo de los plugins que seleccionemos, obtendremos un informe más o menos útil para nuestros propósitos. Hay que tener en cuenta que algunos plugins pueden darnos información de vulnerabilidades existentes cuando esto no es así, provocando lo que se denomina falsos positivos. No debemos sorprendernos si realizamos el escaneo de nuestro sistema Linux y detectamos algún tipo de vulnerabilidad relacionada con Windows. Este tipo de problemas no hay que interpretarlos como inestabilidad o falta de confianza sobre Nessus. Por el contrario, el origen es la mala utilización de algunos de los plugins (uso de plugins que buscan vulnerabilidades de Windows en sistemas Unix). Estos comportamientos erróneos también se producen en el software comercial destinado a este propósito.

Figura 27.2: Plugins de Nessus



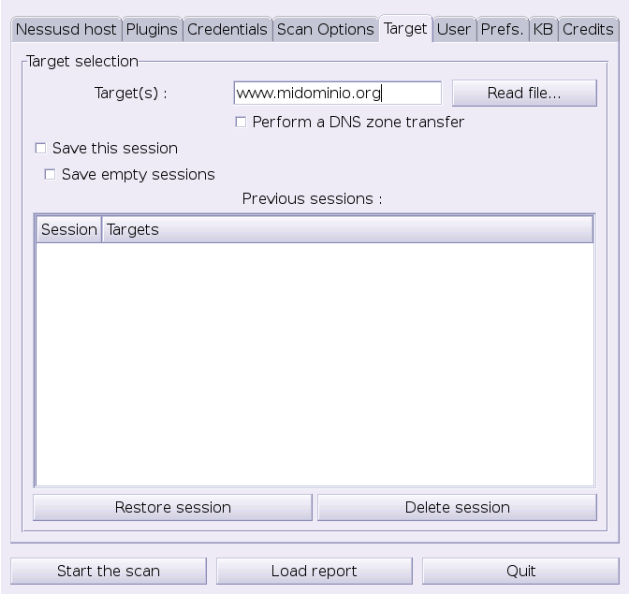
Los plugins están categorizados de varias formas, lo que en ocasiones puede llevarnos a confusión. Uno de los métodos de agrupación de los plugins, es por categoría. Más importante aún es la distinción de los plugins basándonos en su peligrosidad o habilidad para provocar ataques DoS. Estos plugins pueden llegar a bloquear un sistema que sea vulnerable al ataque, por lo que es recomendable utilizarlos con cuidado. No causarán pérdidas irreparables, pero será necesario reiniciar el sistema[13]. Existe un botón donde podemos seleccionar todos los plugins o únicamente aquellos que no sean peligrosos. Otro método de categorización de los plugins es mediante familias:Windows, FTP SNMP, SMB, Cisco, etc.

Otra de las partes críticas del proceso de escaneo, es el escaneo de puertos. Es un proceso en el que se identifican los puertos activos para una dirección IP. Cada puerto está unido a una aplicación específica. Nessus tiene la inteligencia suficiente para realizar un chequeo únicamente si el programa para ese chequeo está disponible. Por ejemplo, sólo se ejecutarán los plugins para servidores web si se encuentra uno. Debido a que muchos servicios no corren en los puertos por defecto, Nessus tiene un plugin que se encarga de reconocer qué aplicaciones están asociadas a cada puerto que se encuentra a la escucha.

Existen más pestañas que nos permiten configurar otras opciones de Nessus, pero no entraremos en ellas. Lo que pretendemos es tener una primera toma de contacto con esta utilidad, lo suficiente para permitirnos lanzar escaneos contra servidores para identificar sus vulnerabilidades. Así, pasaremos directamente a la pestaña **Target Selection**, donde debemos identificar nuestros objetivos. Pueden especificarse mediante una dirección IP, como una subred o como un rango de direcciones IP. Una vez configurado este último punto, podemos comenzar la ejecución del escaneo.

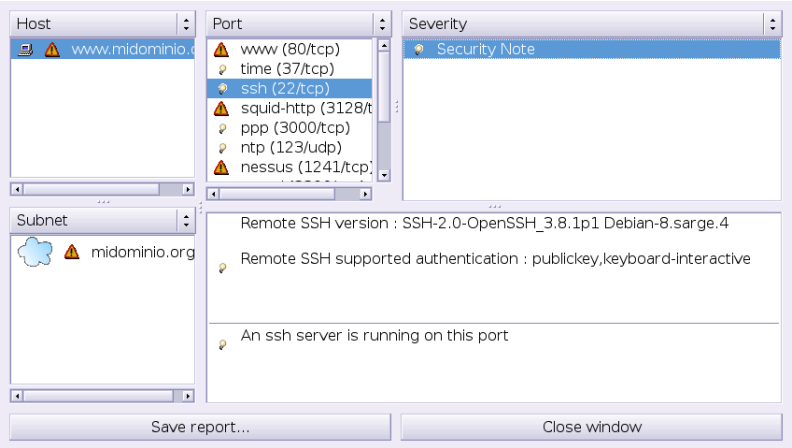
[13]Están identificados en la pestaña de plugins por una señal de peligro.

Figura 27.3: Selección de objetivos



Como ejemplo, utilizaremos el mismo servidor donde hemos instalado el servidor y cliente de Nessus.

Figura 27.4: Informe de vulnerabilidades presentado por Nessus



El informe obtenido puede guardarse en varios formatos (entre los que se encuentra HTML), para su posterior estudio.

27.3. Crackeadores de password: John the Ripper

Uno de los crackeadores de claves más famosos es John the Riper. Está disponible para Unix (hasta 11 tipos independientes de la arquitectura), DOS, Win32, BeOS y OpenVMS. A pesar de que muchas veces su uso tiene oscuras intenciones, se creó como una herramienta capaz de detectar las claves débiles de un sistema Unix.

En los comienzos de Unix, incluso en la actualidad, los administradores de sistemas no le daban importancia a que cualquier usuario del sistema pueda leer el fichero `/etc/passwd`. Confiaban ciegamente en la función `crypt()` que se encargaba de almacenar en este fichero la clave encriptada. Incluso el sistema `shadow` no libra a un sistema de obtener un fichero `/etc/passwd` con las claves encriptadas, basta con conseguir `/etc/passwd` y `/etc/shadow` y migrarlos a otro sistema donde se tenga acceso como `root`. A continuación se verá un ejemplo que representa esta problemática.

27.3.1. Instalacion

Se va a construir la herramienta *John the Ripper* a partir de los fuentes[14], obteniendo de esta manera un binario para el sistema. Lo primero es descargar la última versión de `http://www.openwall.com/john/` y descomprimirlo en el directorio que se desea.

Una vez desempaquetado se procede a compilar el código fuente:

```
cd scr
make
```

Con esto se obtiene una lista de los sistemas soportados. Puede elegirse uno de los que aparece en la lista o simplemente `Generic`:

```
make SISTEMA
```

ó

```
make generic
```

En el caso que la compilación se lleve a cabo con éxito, el binario de la herramienta es `run/john`. Es recomendable que el directorio donde se encuentre instalada esta herramienta tenga un acceso restringido para evitar que cualquier usuario puede hacer un uso fraudulento de sus funcionalidades.

27.3.2. Crackeando el fichero /etc/passwd

La instalación de esta herramienta es extremadamente fácil tal como acaba de verse, sin embargo, su potencia es muy grande, siendo una herramienta de uso obligado para todos los administradores de sistemas. Con un uso regular de John the Riper se obtiene la certeza de una buena elección de las claves de nuestros usuarios.

Una de las principales funcionalidades es el descubrimiento de las claves mediante ataques de diccionario. Este tipo de ataque consiste en utilizar una base de datos de posibles claves cuyo origen es un diccionario. La Real Academia de la Lengua Española proporciona una gran base de datos con esta información.

Para el ejemplo que nos ocupa, en caso de tener soporte `shadow` para las claves será necesario ejecutar la utilidad `pwunconv`, que repone las claves encriptadas al fichero `/etc/passwd`. Se realizará una copia de este fichero para trabajar con él y a continuación se ejecuta `pwconv` para volver a la situación inicial y devolver al sistema el soporte `shadow`. Para realizar esto también puede utilizarse la utilidad `unshadow` que se incluye con John the Riper.

Como primera opción se utilizará el método single, que busca posibles claves en el propio fichero (muchas personas utilizan como clave su nombre o apellidos):

```
root@guadalinex:/usr/local/src/john-1.6# ./run/john -single passwd.john
Loaded 1 password (Standard DES [24/32 4K])
guesses: 0  time: 0:00:00:00 100%  c/s: 738  trying: hugo1934 - hugo1969
```

[14] Se puede optar por:

```
#apt-get install john wenglish wspanish
```

En este caso no ha detectado ninguna clave al ser el fichero elegido el de un sistema con pocos usuarios.

El siguiente paso es realizar la búsqueda de claves comparando contra un listado (lo mejor es tener un diccionario de la lengua de origen de los usuarios del sistema) de posibles claves. Sería el modo *wordlist*. Para indicar el fichero que se utilizará para la comparación se utiliza la opción `-wordfile`.

Para utilizar modificaciones del listado proporcionado se utiliza la opción `-rules`. Las reglas que se utilizan para generar estas variaciones se encuentra en el fichero de configuración `john.ini`, recomendando su visualización para comprender las variaciones que se van a realizar.

A continuación se ve la ejecución de la utilidad indicando que la lista de posibles claves está en `pasword.lst` y utilizando reglas para realizar variaciones de las mismas, estando el fichero de claves en `password.john`.

```
root@guadalinex:/usr/local/src/john-1.6# ./run/john -wordfile:run/password.
    lst -rules passwd.john
Loaded 1 password (Standard DES [24/32 4K])
guesses: 0  time: 0:00:00:01 100%  c/s: 101207  trying: Raptorin - Zenithin
root@guadalinex:/usr/local/src/john-1.6# ./run/john -show passwd.john root:
    administ:0:0:root:/root:/bin/bash
1 password cracked, 2 left
```

En este caso hemos encontrado la clave de `root` (no haría falta ninguna más para poner en apuros al administrador del sistema).

Es recomendable editar el fichero `password.lst` para que contenga posibles claves en español, ya que los valores que contiene se refieren a claves en inglés. Como ya se indicó lo mejor sería conseguir un listado de las palabras de un diccionario, seguro que algún usuario ha utilizado como contraseña algo como "ornitorrinco" pensando que a nadie se le va a ocurrir probar este clave (a John si se le ocurrirá).

Otra forma de realizar de generar claves para chequear es utilizando combinaciones de letras y número. Este sería el modo incremental de verificación de John the Riper. Las opciones disponibles son:

`alpha` Genera palabras con letras solamente, es decir 26 letras

`digits` Genera palabras con numeros solamente, desde el 0 hasta el 9

`all` Genera palabras con letras, numeros y caracteres especiales, en total son 90 caracteres

Cuantas más combinaciones más tiempo se tardará en chequear todo el fichero de claves. Sin embargo, hay que tener en cuenta que un posible intruso suele ser un individuo al cual le sobran recursos y utiliza su tiempo libre para buscar vulnerabilidades.

Son muchas las opciones que presenta John the Riper y se recomienda la lectura de
`http://www.decowar.com/manual_john_the_ripper.htm`
donde se explica de forma detallada cómo crear nuevas reglas. Para un uso ético de esta herramienta bastaría con las opciones que se han detallado anteriormente, la referencia anterior sería para usos más oscuros[15].

27.4. Detección de intrusiones

27.4.1. Razones para la detección de intrusiones

Hasta ahora hemos cubierto algunos aspectos sobre la seguridad del sistema en el ámbito preventivo. Cuando un atacante decide probar su suerte contra nuestro sistema, lo primero que hace

[15]Como se está viendo todas las herramientas de detección de vulnerabilidades tienen su "reverso tenebroso", dependiendo de si la usa Obi-Wan Kenobi o Darth Vader el objetivo que se busca es uno u otro.

es recopilar cuanta información le sea posible acerca del mismo. Toda información que consiga averiguar puede serle útil: sistema operativo, versión, servicios que ofrecemos, versión de los programas que tenemos... Cualquiera de estos datos puede ser suficiente para que su ataque tenga éxito. Basta con que el atacante vea, por ejemplo, que tenemos una versión vieja de un programa, aunque no tenga éste ninguna vulnerabilidad importante, para que se dé cuenta de que no somos administradores muy cuidadosos y probablemente tengamos otros servicios descuidados.

Esto nos exige no sólo que cuidemos las versiones y posibles parches de seguridad a aplicar en nuestro sistema, sino también el vigilar los intentos de acceso.

La técnica más utilizada por los potenciales intrusos de nuestro sistema para obtener información acerca de nosotros, es el barrido de puertos. Esta técnica se basa en intentar conectarse a cada uno de los puertos que tiene abiertos nuestro servidor, anotando qué es lo que tiene activo y analizando dicha información. Una de las herramientas más comunes para realizar barridos de puertos es `nmap`.

No entraremos en el uso de `nmap`, ya que se trató en una entrega anterior. Recordemos que tras una ejecución de esta herramienta, obtendremos un listado de los puertos abiertos, pudiendo el atacante revisar si encuentra alguna versión vieja o vulnerable de software.

```
[root@fedora root]# nmap -sT fedora
Starting nmap 3.48 ( http://www.insecure.org/nmap/ ) at 2004-05-18 13:38
   CEST
Interesting ports on localhost (172.26.0.40):
(The 1648 ports scanned but not shown below are in state: closed)
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
110/tcp   open  pop-3
111/tcp   open  rpcbind
443/tcp   open  https
514/tcp   open  shell
10000/tcp open  snet-sensor-mgmt
10082/tcp open  amandaidx
10083/tcp open  amidxtape
32770/tcp open  sometimes-rpc3
Nmap run completed -- 1 IP address (1 host up) scanned in 3.660 seconds
```

Ahora podemos intentar averiguar más información sobre algunos de los servicios:

```
[root@fedora root]# telnet 172.26.0.40
Trying 172.26.0.40...
Connected to 172.26.0.40.
Escape character is '^]'.
Fedora Core release 1 (Yarrow)
Kernel 2.4.22-1.2115.nptl on an i686
login:
[root@fedora root]# telnet 172.26.0.40 22
Trying 172.26.0.40...
Connected to 172.26.0.40.
Escape character is '^]'.
SSH-1.99-OpenSSH_3.6.1p2
[root@fedora root]# telnet 172.26.0.40 25
Trying 172.26.0.40...
Connected to 172.26.0.40.
Escape character is '^]'.
220 fedora.elpiso.es ESMTP Sendmail 8.12.10/8.12.10; Tue, 18 May 2004
    13:50:11 +0200
```

A partir de la información obtenida con `nmap`, hemos logrado averiguar los siguientes datos acerca de las versiones del software instalado:

- SSH-1.99-OpenSSH_3.6.1p2,
- 8.12.10 de sendmail y
- kernel 2.4.22-1.2115.nptl.

Bastaría con buscar información referente a vulnerabilidades sobre este software para explotar agujeros que nos permitirían acceso como root al servidor.

Cabe mencionar que nmap no es una herramienta utilizada exclusivamente por atacantes. Puede utilizarse para barrer nuestras máquinas buscando servicios que hayamos dejado abiertos por error.

27.4.2. Intentos de intrusión en el sistema (Portsentry)

Vemos claramente la gran cantidad de información que en menos de un minuto puede ser expuesta por nuestro mal administrado servidor. Incluso aunque el servidor se revise periodicamente, ¿por qué permitir que un desconocido sepa qué versiones de software tenemos instaladas? Podría guardar estos datos a la espera de la aparición de nuevas vulnerabilidades, adelantándose a nuestras actuaciones.

Como acabamos de ver, es fundamental la detección de estos barridos. Detectar un barrido de puertos es muy fácil; se producirán muchas conexiones casi simultáneas a una gran cantidad de puertos originadas desde la misma máquina. Aunque los programas que se dedican a realizar estos barridos se han vuelto muy sofisticados y cada vez es más difícil detectarlos por las diferentes estrategias que emplean[16], el principio básico es el mismo. Hay un excelente programa dedicado precisamente a encontrar este patrón y tomar la acción que le indique el administrador del sistema: Portsentry, de Psionic.

Portsentry es un programa muy sencillo. Su misión es escuchar a los puertos que le indiquemos que deben permanecer siempre inactivos. En caso de llegar una conexión a uno de ellos, puede marcarlo en la bitácora del sistema, bloquear toda la comunicación con la dirección identificada como agresora, o correr un comando externo. Podemos bajar la última versión de `http://sourceforge.net/projects/sentrytools/`.

La compilación[17] de Portsentry es muy sencilla. Lo primero que haremos es extraer los ficheros que componen el paquete:

```
[root@fedora src]# tar -zxvf portsentry-1.2.tar.gz
portsentry_beta/
portsentry_beta/portsentry.c
portsentry_beta/portsentry.h
portsentry_beta/portsentry_io.c
portsentry_beta/portsentry_io.h
portsentry_beta/portsentry_util.c
portsentry_beta/portsentry_util.h
portsentry_beta/portsentry_config.h
portsentry_beta/portsentry_tcpip.h
portsentry_beta/portsentry.ignore
portsentry_beta/portsentry.conf
portsentry_beta/Makefile
portsentry_beta/README.COMPAT
```

[16] Nmap sabe hacer desde una sencilla conexión TCP hasta un barrido silencioso con SYN, FIN, Xmas, Null, UDP, paquetes fragmentados, barridos paralelos de diferentes tipos.

[17] Si deseamos instalar el paquete usando el comando `apt-get`, escribiremos

```
#apt-get install portsentry
```

```
portsentry_beta/README.install
portsentry_beta/README.methods
portsentry_beta/README.stealth
portsentry_beta/CHANGES
portsentry_beta/CREDITS
portsentry_beta/LICENSE
portsentry_beta/ignore.csh
```

Es conveniente leer los distintos ficheros README que se han obtenido del paquete, para conocer con más detalle la operación del programa y los pasos a seguir. Una vez descomprimido el paquete, es necesario compilar. Esto se hace con un simple `make <sistema>`, sustituyendo `<sistema>` por nuestro tipo de sistema operativo. En nuestro caso es un sistema Linux:

```
[root@fedora portsentry_beta]# make linux
SYSTYPE=linux
Making
cc -O -Wall -DLINUX -DSUPPORT_STEALTH -o ./portsentry ./portsentry.c \
./portsentry_io.c ./portsentry_util.c
```

Por último, pasamos a instalar portsentry:

```
[root@fedora portsentry_beta]# make install
Creating psionic directory /usr/local/psionic
Setting directory permissions
Creating portsentry directory /usr/local/psionic/portsentry
Setting directory permissions
chmod 700 /usr/local/psionic/portsentry
Copying files
cp ./portsentry.conf /usr/local/psionic/portsentry
cp ./portsentry.ignore /usr/local/psionic/portsentry
cp ./portsentry /usr/local/psionic/portsentry
Setting permissions
chmod 600 /usr/local/psionic/portsentry/portsentry.ignore
chmod 600 /usr/local/psionic/portsentry/portsentry.conf
chmod 700 /usr/local/psionic/portsentry/portsentry
Edit /usr/local/psionic/portsentry/portsentry.conf and change
your settings if you haven't already. (route, etc)
WARNING: This version and above now use a new
directory structure for storing the program
and config files (/usr/local/psionic/portsentry).
Please make sure you delete the old files when
the testing of this install is complete.
```

Con esto, Portsentry quedará instalado en el directorio `/usr/local/psionic/portsentry` listo para ser configurado.

Configuración de portsentry

La configuración de Portsentry se hace en el archivo[18] `/usr/local/psionic/portsentry/portsentry.conf`. A primera vista el archivo parece muy complicado, con muchas opciones. Sin embargo, configurarlo es más fácil de lo que parece.

Portsentry tiene varios modos de operación. El más común y sencillo es el modo clásico, y el más potente (aunque no disponible en todos los sistemas) es el modo avanzado.

Modo clásico. En este modo, le especificamos a Portsentry que escuche determinados puertos TCP y UDP, especificados con las opciones UDP_PORTS y TCP_PORTS por los cuales

[18] En `/etc/portsentry/portsentry.conf` si hemos usado el `.deb`

no estamos dando ningún servicio y son poco solicitados. Por ejemplo, puede que nuestro servidor no esté dando servicio de red SMB (Samba, red tipo Microsoft). Sin embargo, es común que las computadoras windows manden mensajes broadcast buscando a un sistema en específico, con lo que podríamos recibir una gran cantidad de alertas en falso. Vienen varios puertos predefinidos en el archivo de configuración, y son buenos como recomendación inicial.

Modo *stealth*. En este modo Portsentry abre sockets crudos (raw), lo que le permite detectar una mayor cantidad de barridos y ataques tales como ataques de conexión normal, SYN/half-open, FIN, NULL y XMAS. Este modo es un tanto experimental, por lo cual no funcionará en todos los sistemas.

Modo avanzado. Este modo es también considerado, hasta cierto punto, perteneciente a la categoría stealth. En este modo, Portsentry no abre ningún puerto, sino que le pide al kernel que le notifique si llega alguna petición a algún puerto menor al especificado en las opciones ADVANCED_PORTS_TCP y ADVANCED_PORTS_UDP. Será necesario excluir algunos puertos que puedan generar falsas alarmas, como el ejemplo que comentábamos sobre SMB. Para excluir estos puertos tenemos las opciones ADVANCED_EXCLUDE_TCP y ADVANCED_EXCLUDE_UDP.

Como ventaja adicional del modo avanzado, está que al ejecutar `netstat -na` no nos reportará los puertos que está escuchando, dado que no están realmente abiertos. Ésto puede simplificar un tanto nuestro trabajo como administradores. El modo avanzado es mucho más sensible que el modo clásico, dado que escucha a muchos más puertos, por lo que puede efectivamente causar una negación de servicio si no es configurado con cuidado.

Tras haber especificado los puertos que deseamos escuchar, hay algunos parámetros adicionales que debemos especificar:

IGNORE_FILE es el nombre del archivo que incluye la lista de direcciones en las que confiamos, y por tanto no queremos bloquear si intentan acceder a un puerto bloqueado.

HISTORY_FILE contiene la lista de direcciones que Portsentry ha detectado intentando acceder a puertos monitorizados.

BLOCKED_FILE es equivalente a HISTORY_FILE, pero relevante únicamente a la sesión actual de Portsentry.

BLOCK_TCP especifica qué hacer cuando un barrido de puertos TCP es detectado. Tiene tres posibles valores:

0	sólo registrar el intento,
1	bloquear la dirección que intentó averiguar acerca de nosotros, y
2	correr un comando externo especificado en KILL_RUN_CMD.

BLOCK_UDP es equivalente a BLOCK_TCP para barridos de puertos UDP.

KILL_ROUTE guarda el comando utilizado para descartar toda la comunicación con una dirección, normalmente mediante reglas de filtrado de paquetes.

KILL_HOSTS_DENY tiene la línea que deberá ser agregada a `/etc/hosts.deny` para que la dirección atacante sea bloqueada por TCPwrappers. Es conveniente activarlo, pues a diferencia de las reglas manejadas por KILL_ROUTE este archivo permanecerá aunque el sistema se rearranque.

KILL_RUN_CMD puede guardar un comando para ser ejecutado en caso de ser detectada una intrusión. No recomendamos utilizar esta opción, ya que puede fácilmente llevar a una negación de servicio. Hay administradores que sugieren utilizar esta opción para lanzar un contraataque contra el atacante. La mejor defensa es tener nuestro sistema seguro, no atacar al enemigo. Al atacar al enemigo, lo más probable es que centre más su atención en nosotros[19] y, con el paso del tiempo, logre penetrar nuestra seguridad. Es mucho mejor aparentar que no ocurrió nada o simplemente tirar la conexión que atacar de vuelta.

SCAN_TRIGGER indica con qué retardo se marcará un intento de conexión fallido como un ataque. Probablemente, si a la primera bloqueamos toda comunicación con el presunto atacante, dejaremos fuera a muchos usuarios legítimos que por casualidad hicieron la conexión equivocada. Sin embargo, si ponemos un número muy alto nos exponemos a dar más información de la que hubiéramos querido. Un valor de 1 ó 2 es recomendado, aunque los muy paranoicos querrán mantenerlo en 0.

Arranque automático de portsentry

La manera más genérica[20] de iniciar portsentry es incluirlo en el último archivo que se ejecuta al iniciar el sistema `/etc/rc.local`. Basta con agregar al final de éste las líneas necesarias para levantar a portsentry con la configuración que deseemos. Las opciones de arranque de portsentry son:

-tcp Iniciar en modo clásico, escuchar TCP.

-udp Iniciar en modo clásico, escuchar UDP.

-stcp Iniciar en modo stealth, escuchar TCP

-sudp Iniciar en modo stealth, escuchar UDP

-atcp Iniciar en modo avanzado, escuchar TCP

-audp Iniciar en modo avanzado, escuchar UDP

Normalmente levantaremos dos copias del programa en el mismo modo general, una escuchando UDP y la otra TCP.

Respuesta de portsentry ante un ataque

El simple hecho de que Portsentry evite ciertos ataques al sistema es de por sí muy bueno y deseable. Sin embargo, para que ésto nos sea realmente útil, tenemos que analizar nuestras bitácoras y llevar registro de quién y cuándo intentó barrer nuestros puertos. Además, sólo leyendo las bitácoras sabremos si estamos limitando de más, bloqueando el acceso de máquinas legítimas.

Afortunadamente, Portsentry utiliza syslog para informar de toda la información que el administrador debe saber, por lo cual todo lo que necesitamos estará típicamente en el archivo `/var/log/messages`, o donde se lo hayamos especificado en el `syslogd.conf`. La detección de un barrido hecho por nmap en un sistema Linux se ve así:

```
[root@fedora root]# nmap -sT fedora
May 18 14:17:44 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 79
May 18 14:17:44 fedora portsentry[2642]: attackalert: Host 172.26.0.40 has
    been blocked via wrappers with string: "ALL: 172.26.0.40"
May 18 14:17:44 fedora xinetd[2648]: libwrap refused connection to shell (
    libwrap=in.rshd) from 172.26.0.40
```

[19] Puede considerarlo un reto al enfrentarse a un administrador de un gran nivel.
[20] Si lo hemos instalado a partir de las fuentes.

```
May 18 14:17:44 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 635
May 18 14:17:44 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:44 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 1524
May 18 14:17:44 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:44 fedora xinetd[2650]: libwrap refused connection to amidxtape
     (libwrap=amidxtaped) from 172.26.0.40
May 18 14:17:46 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 119
May 18 14:17:46 fedora xinetd[2652]: libwrap refused connection to amandaidx
     (libwrap=amindexd) from 172.26.0.40
May 18 14:17:46 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:48 fedora xinetd[2654]: warning: can't get client address:
    Connection reset by peer
May 18 14:17:48 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 1
May 18 14:17:48 fedora xinetd[2655]: warning: can't get client address:
    Connection reset by peer
May 18 14:17:48 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:48 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 11
May 18 14:17:49 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:49 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 15
May 18 14:17:49 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:49 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 111
May 18 14:17:49 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:49 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 143
May 18 14:17:49 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:49 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 540
May 18 14:17:49 fedora portsentry[2642]: attackalert: Host: 172.26.0.40 is
    already blocked. Ignoring
May 18 14:17:49 fedora portsentry[2642]: attackalert: Connect from host:
    localhost/172.26.0.40 to TCP port: 1080
(...)
```

Ésta sería la respuesta con el archivo de configuración por defecto. Sin embargo, hemos visto que existen varias acciones a tomar dependiendo de lo que consideremos oportuno como respuesta. Estas acciones quedarán reflejadas también en el fichero de log.

Cuando configuremos portsentry es muy común que marquemos direcciones como atacantes por error. Borrarlas de la lista de bloqueo es muy sencillo. Nuestro primer paso será editar el archivo `/etc/hosts.deny` y buscar la dirección que queremos eliminar. En caso de que hayamos establecido algún tipo de filtrado de paquetes (por ejemplo con iptables) será necesario permitir de nuevo la entrada de los paquetes con origen en la dirección que hemos vetado por error.

Las posibilidades de portsentry son tantas como nosotros configuremos, aunque debemos tener

cuidado y afinar bien la configuración para evitar negar el acceso a direcciones por error.

27.4.3. Integridad del sistema (Tripwire)

Existe un **bug** en la **versión actual de Tripwire** que provoca un error en la instalación **sobre Fedora**. Aún así, hemos preferido dar información sobre esta herramienta a la espera de obtener una versión de Tripwire que solucione este error. Puede encontrarse información más detallada en `http://www.redhat.com/archives/fedora-list/2003-November/msg05954.html`.

No existen sistemas perfectos e invulnerables a los ataques de usuarios maliciosos. Además de las posibles medidas preventivas que tomemos (cortafuegos, parches, políticas, ...) siempre cabe la posibilidad de que un intruso logre entrar en nuestro sistema. Normalmente los ataques que se producen, conllevan la modificación parcial del sistema (archivos de configuración, páginas web, ...). Estas modificaciones tienen como objetivo dejar una marca o firma en el servidor como prueba de la violación del sistema o posibles puertas traseras para accesos posteriores.

Tripwire asumirá que los controles preventivos de seguridad han fallado y que nuestro sistema ha sido alterado. El atacante intentará por todos los medios que el administrador no sepa los ficheros que han sido modificados. Es aquí donde interviene Tripwire, alertando al administrador de los cambios que se produzcan en el sistema.

Tripwire realiza una monitorización de la integridad de los archivos que le indiquemos, detectando cualquier cambio en los ficheros indicados. El cambio puede ser tanto de contenido como de permisos.

Instalación de Tripwire

Para bajar la última[21] versión disponible de Tripwire accedemos a
`http://www.tripwire.org/download/index.php`
y seleccionaremos para bajar el fichero `tripwire-2.3-47.bin.tar.gz`.
Una vez bajado en nuestro sistema:

```
# cd /usr/local
# tar -zxvf /home/hugo/Tripwire/tripwire-2.3-47.bin.tar.gz
```

Tripwire utiliza dos claves, que pueden ser palabras u oraciones, para almacenar su información de forma encriptada. La denominada "site key" se emplea para encriptar los archivos de configuración y las políticas. La denominada "local key" se utiliza para encriptar la información referida al estado de los ficheros que se monitorizan. Para finalizar la instalación de Tripwire es necesario ejecutar el script `install.sh`, el cual creará las claves que acabamos de mencionar.

```
[root@fedora tripwire-2.3]# ./install.sh
Installer program for:
Tripwire(R) 2.3 Open Source for LINUX
Copyright (C) 1998-2000 Tripwire (R) Security Systems, Inc. Tripwire (R)
is a registered trademark of the Purdue Research Foundation and is
licensed exclusively to Tripwire (R) Security Systems, Inc.
LICENSE AGREEMENT for Tripwire(R) 2.3 Open Source for LINUX
Please read the following license agreement. You must accept the
agreement to continue installing Tripwire.
Press ENTER to view the License Agreement.
                    GNU GENERAL PUBLIC LICENSE
                       Version 2, June 1991
```

[21] U optar por

```
#apt-get install tripwire
```

```
 Copyright (C) 1989, 1991 Free Software Foundation, Inc.
                      59 Temple Place, Suite 330, Boston, MA  02111-1307
                         USA
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.
                            Preamble
  The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public
(...)
Please type "accept" to indicate your acceptance of this
license agreement. [do not accept] accept
Using configuration file install.cfg
Checking for programs specified in install configuration file....
/usr/lib/sendmail exists.  Continuing installation.
/bin/vi exists.  Continuing installation.
-------------------------------------------------------
Verifying existence of binaries...
./bin/i686-pc-linux_r/siggen found
./bin/i686-pc-linux_r/tripwire found
./bin/i686-pc-linux_r/twprint found
./bin/i686-pc-linux_r/twadmin found
This program will copy Tripwire files to the following directories:
        TWBIN: /usr/sbin
        TWMAN: /usr/man
     TWPOLICY: /etc/tripwire
     TWREPORT: /var/lib/tripwire/report
         TWDB: /var/lib/tripwire
 TWSITEKEYDIR: /etc/tripwire
TWLOCALKEYDIR: /etc/tripwire
CLOBBER is false.
Continue with installation? [y/n]
(...)
-------------------------------------------------------
The Tripwire site and local passphrases are used to
sign a variety of files, such as the configuration,
policy, and database files.
Passphrases should be at least 8 characters in length
and contain both letters and numbers.
See the Tripwire manual for more information.
-------------------------------------------------------
Creating key files...
(When selecting a passphrase, keep in mind that good passphrases typically
have upper and lower case letters, digits and punctuation marks, and are
at least 8 characters in length.)
Enter the site keyfile passphrase: sitekey
Verify the site keyfile passphrase: sitekey
Generating key (this may take several minutes)...
Generating key (this may take several minutes)...Key generation complete.
(When selecting a passphrase, keep in mind that good passphrases typically
have upper and lower case letters, digits and punctuation marks, and are
at least 8 characters in length.)
Enter the local keyfile passphrase: localkey
Verify the local keyfile passphrase: localkey
Generating key (this may take several minutes)...Key generation complete.
-------------------------------------------------------
Generating Tripwire configuration file...
-------------------------------------------------------
Creating signed configuration file...
```

```
Please enter your site passphrase:
Wrote configuration file: /etc/tripwire/tw.cfg
A clear-text version of the Tripwire configuration file
/etc/tripwire/twcfg.txt
has been preserved for your inspection.  It is recommended
that you delete this file manually after you have examined it.
---------------------------------------------------------------
Customizing default policy file...
---------------------------------------------------------------
Creating signed policy file...
Please enter your site passphrase:
Wrote policy file: /etc/tripwire/tw.pol
A clear-text version of the Tripwire policy file
/etc/tripwire/twpol.txt
has been preserved for your inspection.  This implements
a minimal policy, intended only to test essential
Tripwire functionality.  You should edit the policy file
to describe your system, and then use twadmin to generate
a new signed copy of the Tripwire policy.
---------------------------------------------------------------
The installation succeeded.
Please refer to /usr/doc/tripwire/Release_Notes
for release information and to the printed user documentation
for further instructions on using Tripwire 2.3 Open Source for LINUX.
```

Durante la propia instalación ha sido necesario utilizar site key para encriptar el fichero con las políticas que utilizará Tripwire durante su ejecución.

Configuración de Tripwire

La configuración de los archivos que se van a monitorizar se mantiene en el archivo de políticas (*policy file*). Su manipulación es algo tediosa debido a su extensión. A modo de ayuda, Tripwire proporciona un archivo de ejemplo que sirve de plantilla para definir nuestras políticas. Este archivo es `/etc/tripwire/twpol.txt`.

Podemos empezar a trabajar sobre este archivo, aunque es recomendable hacer una copia sin modificar del mismo. Hay que hacer una consideración sobre este archivo. Está creado con vistas a comprobar la integridad de todo el sistema, lo que implica que tardará varios minutos (dependiendo de la potencia de nuestro equipo).

Una vez tengamos claro el archivo de políticas que vamos a utilizar, ya sea el que viene como ejemplo o uno basado en modificaciones sobre el mismo, es necesario instalarlo. Tripwire utilizará una versión compilada y encriptada de este archivo, que se almacenará en `/etc/tripwire/tw.pol`. Para generarlo utilizaremos la utilidad `twadmin`:

```
[root@fedora tripwire]# twadmin -m P /etc/tripwire/twpol.txt
Please enter your site passphrase:
Wrote policy file: /etc/tripwire/tw.pol
```

Una vez configurado e instalado el archivo de políticas, Tripwire necesita recolectar la información actual de los archivos a los cuales comprobará su integridad. Dicha información se almacena en una base de datos, que se genera mediante la utilidad `tripwire`:

```
[root@fedora tripwire]# tripwire -m i
```

En caso de que se generen errores, será necesario corregirlos sobre el fichero `twpol.txt`, siendo necesario instalarlo de nuevo con `twadmin`.

Una vez que tengamos Tripwire correctamente configurado con su base de datos, es el momento de verificar la integridad del sistema. Para ello ejecutaremos de nuevo la utilidad `tripwire`:

```
[root@fedora tripwire]# tripwire -m c
```

Utilizaremos la llamada a la utilidad `tripwire` de esta manera cada vez que deseemos saber el estado en que se encuentra nuestro sistema respecto a la integridad del mismo.

Si por algún motivo, alguno de los archivos que estamos comprobando ha sido modificado, será necesario reconstruir la base de datos de Tripwire. Previamente es necesario haber comprobado que la modificación ha sido originada por un proceso controlado y no es consecuencia de una intrusión al sistema.

Una vez comprobado que el archivo `twpol.txt` se adapta a los requerimientos de control de la integridad de nuestro sistema y que hemos alcanzado un estado estable del mismo, es recomendable programar la ejecución de los chequeos de forma automática. Para ello podemos hacer uso del `cron` con una entrada similar a la siguiente:

```
/usr/sbin/tripwire -m c | mail root@localhost
```

Así, la salida del chequeo será enviada por correo a `root`. Esta funcionalidad de envío por correo de los informes puede conseguirse con la configuración de Tripwire. La directiva `email-to=user@host.domain` proporciona esta funcionalidad. Deberá insertarse en la configuración de cada grupo de archivos que vamos a comprobar. Cuando alguno de estos archivos se modifique, Tripwire notificará al destinatario especificado en esta directiva, utilizando para ello Sendmail o Postfix.

Capítulo 28

Análisis Forense

> La exigencia de la seguridad es la parte más atractiva de la administración de sistemas. Hollywood no describe el drama de trasladar una red operativa de un sitio a otro, la instalación del último hardware, la comedia de las preguntas de los usuarios o la tragedia de operaciones de restauración fallidas. Sin embargo, la intriga de la seguridad en las computadoras ha sido objeto de numerosas películas. (*Administración de Sistemas Linux*, M Carling y otros)

Una de las afirmaciones más conocidas en seguridad es la que dice algo así como que el servidor más seguro es aquél que se encuentra desconectado de la red y guardado en una caja de seguridad. Este servidor es también el más inútil ya que nadie tiene acceso a los posibles servicios que ofrece. De aquí se deduce que la seguridad 100 % no existe y siempre, a pesar de nuestro esfuerzo, es posible que un atacante tenga éxito en sus intentos por acceder de forma clandestina a nuestro sistema.

El Análisis Forense de Sistemas (*Computer Forensics*) es el proceso de extracción, conservación, identificación, documentación, interpretación y presentación de las evidencias de un ataque informático. La víctima de un ataque informático es un servidor y es preciso que se compruebe el alcance de la intrusión. No basta con borrar los contenidos y reinstalar de nuevo. En la reinstalación haremos uso de las copias de seguridad existentes en lo referente a ficheros de configuración y puede que éstos hayan sido modificados por el atacante. Es necesario averiguar cuándo se realizó el ataque, en qué consistió el mismo y cual fue la puerta de acceso al sistema. Sin un análisis forense de sistemas se caerá en los mismos errores que provocaron el acceso al sistema.

La metodología básica de un análisis forense consiste en:

1. Adquirir evidencias sin alterar el sistema original. Un primer paso a realizar es aislar el sistema sospechoso, si es posible deteniendo los servicios proporcionados y desconectándolo de la red.

2. Comprobar las evidencias recogidas. Cualquier evidencia que se encuentre debe ser comprobada mediante técnicas de hashing. De esta forma se obtendrá una huella con la que comparar los datos recogidos.

3. Analizar los datos sin modificarlos. Lo ideal es acceder a los discos del servidor desde otro sistema en modo de solo lectura, evitando la posible modificación accidental de algún dato, y realizar una copia de los mismos.

Los pasos para empezar la investigación de un incidente son diferentes en cada caso. El investigador debe tomar decisiones basándose en su experiencia y el "sexto sentido" para llegar al fondo del asunto. No es necesario seguir pasos determinados, ni su orden es importante a veces.

28.1. Recopilando evidencias

Ya se ha visto que el primer paso a la hora de realizar un análisis del sistema comprometido es aislar los datos. Es importante recopilar la información antes de apagar el sistema comprometido. Una vez apagado el ordenador se borrará cualquier evidencia posible de la intrusión (procesos escuchando en determinados puertos o procesos realizando tareas en segundo plano).

Hay que resaltar que no es factible el análisis de un sistema comprometido utilizando las herramientas del mismo, éstas podrían estar infectadas, haber sido modificadas para borrar evidencias. Esto puede solucionarse si se dispone de un dispositivo de almacenamiento extraible (CDROM, disco USB, ...) con el kit de herramientas necesarias para recopilar los datos[1].

Para obtener datos del sistema de archivos, una opción es realizar una copia de los distintos sistemas de archivos para su estudio en otro servidor seguro, en el caso que no sea posible apagar el sistema comprometido. Será necesario disponer de almacenamiento adicional para el volcado de estas imágenes.

Mediante el comando `mount` se obtiene un listado de los sistemas de archivos montados en el servidor y mediante `fdisk` el listado de las particiones existentes en cada unidad de disco, independientemente que estén montadas o no.

Con los datos proporcionados por `mount` y `fdisk` se puede utilizar el comando `dd` para crear una imagen del disco. Este comando realiza imágenes copiando bit a bit los sistemas de archivo. Es recomendable crear una imagen por cada partición del sistema.

```
dd if=/dev/hda1 of=/tmp/hda1.img
```

Una vez generadas las imágenes es conveniente garantizar la autenticidad de las mismas para su posterior verificación. El comando `md5sum` proporcionará una huella de las imágenes para su posterior comprobación.

Para enviar estos datos a otro servidor, en el cual se realizará la investigación, puede utilizarse el comando `netcat`[2]:

```
(host remoto) # nc -l -p 8888 > fichero
(host comprometido) # route -cn | nc guadalinex 8888
```

Con las líneas anteriores el host remoto tiene a `netcat` escuchando por el puerto `8888`, redireccionando todo lo que reciba a `fichero`. Desde el host comprometido se envían las salidas de los comandos necesarios, mediante un pipe, conectando con el host remoto `guadalinex` por el puerto `8888`. En el caso de querer enviar por la red la imagen del sistema de archivos obtenida con `dd`:

```
(host remoto) # nc -l -p 8888 > hda.dd
(host comprometido) # dd if=/dev/hda | nc guadalinex 8888
```

El acceso a las imágenes para analizar su contenido se realizará montando el dispositivo virtual `loop` existente en Linux. Este dispositivo representa una abstracción que permite acceder a imágenes de sistemas de archivos.

```
mkdir /mnt/hda1
mount -t ext2 -o loop -r hda1.img /tmp/hda1
```

Una vez montado se accede al mismo al igual que otro sistema de archivos convencional.

Es importante también obtener datos del estado de la memoria. El que Linux trate como a un fichero la memoria hace esta labor más fácil. La memoria principal se encuentra alojada en `/dev/men` y la memoria de `swap` en la partición correspondiente.

Mediante comandos como `strings` o `grep` puede recorrerse el contenido de la memoria.

```
strings /dev/mem | more
```

[1] Será necesario compilar las herramientas necesarias de forma estática.

[2] Como curiosidad ver la página del manual de netcat (`man nc`) para ver cuál es la definición de esta herramienta.

Visualizando este fichero pueden descubrirse las librerías que hay cargadas en memoria. Al ser la memoria un dispositivo volatil es imposible verificar, como se hizo con los sistemas de archivos, que los datos capturados se corresponden con los originales.

Otro punto interesante es la red. Es necesario comprobar las conexiones de red existentes y los servicios que se encuentran levantados, así como los procesos que los soportan. El comando `netstat` nos ayudará en este aspecto, siendo posible obtener información sobre los procesos asociados con cada conexión específica:

```
netstat -pan | more
```

Dentro del proceso de recopilación de evidencias es muy importante llevar un completo registro (incluyendo fecha y hora) de las evidencias que se vayan descubriendo y los pasos que se están siguiendo. Un método muy sencillo de hacer esto es con el comando `script`, el cual irá almacenando en un fichero todo lo que se teclee en la consola.

```
script -a fichero
```

Para finalizar la captura de datos será necesario teclear `exit`.

28.2. Analizando datos

Hasta ahora se ha conseguido obtener una foto detallada del sistema en el estado en el que se sospecha está comprometido. Es necesario seguir investigando para obtener más datos que ayuden a descubrir la forma en que se produjo la intrusión.

Se empezará comprobando el fichero `/etc/passwd`. Se comprobará detenidamente en busca de usuarios con permisos de root (UID y GID con valor 0) que no debieran tenerlos. Debe buscarse también directorios home de usuarios en localizaciones no habituales (`/tmp` o `/` son muy habituales en estos casos).

Los ficheros dejados por los intrusos como troyanos y similares también suelen localizarse en el directorio `/dev`. Esto hace que sea otro punto en el que deba investigarse, comprobando los ficheros de dispositivos que se hayan creado más recientemente.

Los troyanos que normalmente deja un intruso tienen como objetivo ocultar información sobre el sistema que haga sospechar al administrador que algo va mal, así como puertas traseras que permiten al intruso acceder al sistema cuando quiera.

28.3. Una ayuda al forense: Sleuthkit

Existen también herramientas que nos permiten realizar este análisis de una forma más eficiente. La herramienta Sleuthkit junto con su interfaz web Autopsy (`http://www.sleuthkit.org`) es un buen ejemplo. Sleuthkit es una colección de herramientas de análisis forense, complementarias a las herramientas que proporciona Linux. Permiten examinar el sistema de ficheros sospechoso de una forma no intrusiva. También permiten analizar el medio, soportando el análisis de particiones DOS, BSD, Mac y Sun entre otros.

Cuando se realiza un análisis completo del sistema de ficheros, es de gran ayuda la interfaz gráfica a estas herramientas. El *Autopsy Forensic Browser*, conocido familiarmente como Autopsy, es la interfaz gráfica a las herramientas de Sleuthkit, permitiendo la organización de la información en casos, comprobando la integridad de las imágenes del sistema, búsquedas, etc.

Como características de los datos de entrada que analiza Sleuthkit cabe destacar:

- Sistemas de ficheros raw e imágenes de discos.
- Soporta sistema de ficheros NTFS, FAT, FFS, EXT2FS y EXT3FS.

Dentro de las búsquedas que permite:

- Nombres de ficheros borrados.
- Muestra los contenidos de todos los atributos NTFS.
- Muestra sistemas de ficheros y detalles de la estructura de meta-datos.
- Crea una línea de tiempo de la actividad de los ficheros, con posibilidades de importación y creación de informes.
- Localiza las huellas (hashes) de los ficheros.
- Organiza los ficheros basámdose en su tipo.

Las herramientas que proporciona Sleuthkit pueden dividirse en cuatro categorías:

1. Información de sistemas de archivos completos: fsstat
2. Acceso a datos almacenados en archivos: dcalc, dcat, dls y dstat
3. Información Meta almacenada en inodos: icat, ifind, ils y istat
4. Tareas de nivel archivos: mactime, file y sorter

Para empezar a trabajar con Sleuthkit[3] es necesario tener las imágenes del sistema (obtenidas anteriormente con `dd`). Se montarán estas imágenes como solo lectura, utilizando el dispositivo virtual `loop`.

El primer punto en el que hay que detenerse es el relativo a los archivos borrados, normalmente serán huellas que el intruso ha querido ocultar. El comando `ils` muestra la información del inodo de un archivo del sistema. Si queremos obtener información de los archivos borrados de la partición almacenada en `hda1.dd` de tipo `ext3`:

```
ils -f linux-ext3 -r hda1.dd
```

La salida de la ejecución de `ils` muestra en un formato tabular la información. La línea 1 contendrá la cabecera y la línea 2 los datos correspondientes. Otro comando interesante es `fls` que permite recoger información de cualquier archivo que aún existe.

El conjunto de herramientas proporcionado por Sleuthkit es bastante completo y lo anterior es solo una pequeña introducción. Pueden ser difíciles de usar y el formato en que ofrecen los resultados no ayuda, siendo necesario la ayuda de scripts auxiliares que hagan funciones de filtrado para discriminar información que no interese.

En estas circunstancias el uso de una interfaz gráfica mejora de forma considerable las condiciones de trabajo. Autopsy proporcionará una capa abstracta para los comandos, presentando únicamente los resultados, que realmente es lo que interesa. Proporciona también una herramienta con la que documentar y comentar los datos y resultados.

Autopsy permite abrir tantos casos como deseemos, permitiéndonos así almacenar información de varios incidentes. Se creará un directorio por cada caso, que servirá como almacén de los resultados y datos que vayan extrayéndose de las imágenes.

Desde Autopsy se puede realizar también la gestión de las imágenes que dispongamos, almacenándolas en el directorio indicado anteriormente para su estudio o creando un enlace simbólico

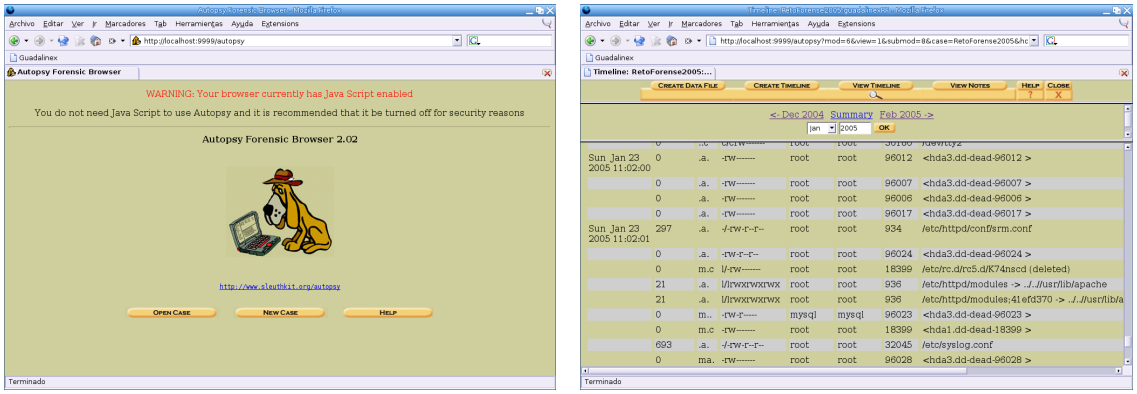
[3]Existen paquetes para la versión inestable de Debian. Instalar paquetes de esa versión puede hacer que nuestro sistema se vuelva inestable y debemos hacerlo con sumo cuidado y bajo nuestra responsabilidad. Si aún así optamos por instalarlos usando el comando `apt-get`, antes hemos de modificar nuestro fichero `/etc/apt/sources.list` añadiendo la línea

```
deb http://ftp.fi.debian.org/debian sid main contrib non-free
```

y después

```
#apt-get update
#apt-get install sleuthkit autopsy
```

Figura 28.1: Pantalla de inicio de Autopsy y análisis de datos



a las imágenes, también con origen en el directorio del caso. El procedimiento a seguir será añadir todas las imágenes indicando el punto de montaje correspondiente, el tipo de sistema de ficheros que utiliza y si se desea que se calcule el MD5 de cada imagen. Esto último nos permite comprobar la integridad de las imágenes.

Una vez acabado este proceso de creación del caso puede comenzar la investigación. Inicialmente puede hacerse una recogida de datos de la línea de tiempo de los cambios producidos en el sistema.

Sleuthkit es una herramienta muy potente y es muy utilizada en los análisis forenses realizados tanto sobre sistemas de archivos Linux como Windows. Sin embargo, estas herramientas son sólo ayudas para la persona que realiza el análisis. Por sí solas no van a decirnos qué ha pasado en el sistema. Será el encargado de realizar el análisis forense el que con su experiencia y conocimiento del sistema deduzca qué es lo que ha pasado.

Existen concursos en los que se publican las imágenes de un sistema comprometido y se propone su estudio y análisis[4]. Los concursantes presentarán sus conclusiones así como los pasos y herramientas utilizadas en el proceso. Si se tiene interés en profundizar sobre este tema es interesante empezar con una de estas imágenes publicadas e intentar relizar el análisis. Una vez finalizado es posible compararlo con las conclusiones obtenidas por los concursantas, descubriendo así los puntos que se hayan pasado por alto y las técnicas que siguen distintas personas.

[4]Pueden encontrarse ficheros de desafío forense en `http://project.honeynet.org/challenge/images.html`

Capítulo 29

Detección de virus

> Un virus es una secuencia de código que se inserta en un fichero ejecutable denominado host, de forma que al ejecutar el programa también se ejecuta el virus; generalmente esta ejecución implica la copia del código viral, o una modificación del mismo, en otros programas
>
> *Seguridad en Unix y Redes.* Antonio Villalón Huerta
>
> virus.(Del lat. virus).
>
> 1. m. Biol. Organismo de estructura muy sencilla, compuesto de proteínas y ácidos nucleicos, y capaz de reproducirse solo en el seno de células vivas específicas, utilizando su metabolismo.
>
> 2. m. Inform. Programa introducido subrepticiamente en la memoria de un ordenador que, al activarse, destruye total o parcialmente la información almacenada.
>
> *DICCIONARIO DE LA LENGUA ESPAÑOLA (Vigésima segunda edición) Real Academia Española*

29.1. Virus y Troyanos en UNIX

Podemos cometer errores o descuidos a la hora de configurar nuestro sistema o de programar una aplicación. Esta es la fuente principal de los problemas de seguridad. Sin embargo, existen otras fuentes, los denominados *malware* o software malicioso. Son programas o `scripts` creados con el objetivo concreto de dañar un sistema o lograr información privilegiada del mismo.

Es importante que nos concienciemos de ejecutar en nuestro sistema únicamente software del cual tengamos la certeza que proviene de una fuente fiable. Ésta consideración es especialmente importante cuando hablamos de la administración del sistema. Muchas labores de administración se realizan desde el usuario `root`, que posee privilegios para realizar cualquier acción en el sistema[1].

Supongamos que encontramos en internet un programa que dice cambiar la configuración gráfica de nuestro sistema de X-Window. En la página donde nos podemos descargar este software nos habla de las excelencias del programa, lo fácil de utilizar que es y nos muestra varias copias de pantalla. Nos descargamos el paquete y en las instrucciones nos indica que ejecutemos el *script* `install.sh` con el usuario `root`. Confiamos ciegamente en él y lo ejecutamos. Cual es nuestra sorpresa cuando vemos cómo se empiezan a borrar todos los archivos de nuestro sistema porque hay una línea en este *script* que ejecuta `cd /; rm -rf *`. Este es un caso extremo, que no tiene porqué pasar ¿verdad?, pero nos advierte de comprobar siempre las fuentes de nuestros programas, especialmente si los bajamos de páginas *underground*[2].

Si un usuario, que no sea `root`, ejecuta un programa que contiene un virus o un troyano, únicamente afectará a los archivos a los que el usuario tiene permiso de escritura o modificación. Como acabamos de comprobar, esto se debe aplicar con más rigor en el caso del administrador del

[1] Aunque esto cambiará con mecanismos como SELinux, por ejemplo.
[2] Los bajos fondos de Internet.

sistema. Si es `root` el que ejecuta el programa contaminado, cualquier archivo del sistema puede contagiarse.

Además de descargar el software de fuentes fiables, es recomendable utilizar las "huellas" de todos los programas, generalmente en forma de MD5[3] para verificar que hemos bajado el archivo legítimo. Igualmente, también tenemos la opción, aunque más ardua, de descargar el código fuente y compilar nosotros mismos los programas. Lograremos así la posibilidad de revisar el código fuente en busca de potenciales problemas de seguridad.

29.1.1. El problema de los virus

De forma similar a los virus que atacan nuestro cuerpo y nos provocan enfermedades, los virus informáticos que atacan nuestros ordenadores, nos pueden producir "dolores de cabeza" y graves problemas[4]. Los virus informáticos poseen múltiples vías de transmisión (como la picadura de los mosquitos o los estornudos en los virus biológicos), aprovechando cualquier descuido o resquicio de seguridad tanto del sistema operativo, los programas de aplicación o el propio usuario. Hay páginas web que incluyen virus, programas que descargamos, incluso si no aplicamos los parches de seguridad para mantener nuestro sistema actualizado, algunos intrusos como el Sasser pueden introducirse en él. Pero, sin duda, la difusión de virus a través del correo electrónico debe llevarse todos los honores como principal vía de contagio.

La mejor forma de proteger nuestra red frente a estos correos con virus, es que ni siquiera lleguen a entrar. El que nuestro servidor de correo disponga un mecanismo de detección y eliminación de virus nos protegerá de infinidad de peligros.

Para ello utilizaremos ClamAV, que es un detector de virus con licencia GPL y que integraremos con nuestro agente de transporte de correo para rechazar los mensajes con virus.

El propósito principal de este software es tanto la integración con los servidores de correo (escaneo de datos adjuntos) como el escaneo de sistemas de ficheros que puedan contener virus (p. ej. un servidor samba para clientes windows).

En un sistema antivirus es muy importante la actualización de los ficheros de firmas[5] y de los motores antivirus para adaptarse a las mutaciones y apariciones de nuevos virus. ClamAV dispone de una herramienta para actualizarse automáticamente desde Internet.

El sitio principal de este antivirus es `http://www.clamav.net`, donde podremos ampliar información.

29.2. Antivirus ClamAV

29.2.1. Instalación

Debian

Para instalarlo, usaremos `apt-get`, ejecutándolo como `root`. En esta ocasión suponemos que se ha instalado el servidor de correo Postfix junto con amavis, como se explicó en la tercera entrega de este curso.

En primer lugar necesitamos instalar, en nuestro caso casi todos estarán y, a lo sumo se actualizarán, los paquetes relacionados con los ficheros adjuntos, a fin de poder inspeccionarlos:

```
root@guadalinex:/home/mowgli# apt-get install unrar lha arj unzoo zip unzip
    bzip2 gzip cpio file lzop
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
unrar ya áest en su óversin áms reciente.
```

[3] Fácilmente con $md5sum fichero, podemos obtener la huella y comprobarla con la auténtica, para comprobar que no ha sido modificado por los malos.

[4] Los efectos de los virus informáticos pueden ir desde la simple broma o reivindicación, a la filtración de datos confidenciales o la destrucción de datos en el ordenador.

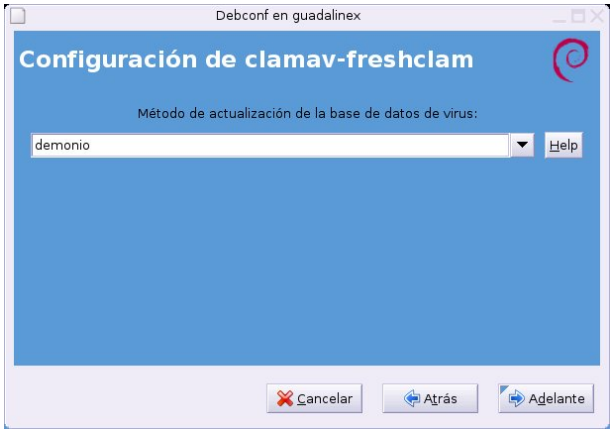
[5] La huella para detectar al virus.

```
unzip ya áest en su óversin áms reciente.
bzip2 ya áest en su óversin áms reciente.
gzip ya áest en su óversin áms reciente.
cpio ya áest en su óversin áms reciente.
Se áinstalarn los siguientes paquetes extras:
libmagic1
Se áinstalarn los siguientes paquetes NUEVOS:
arj lha lzop unzoo
Se áactualizarn los siguientes paquetes:
file libmagic1 zip
3 actualizados, 4 se áinstalarn, 0 para eliminar y 563 no actualizados.
Necesito descargar 680kB de archivos.
Se áutilizarn 984kB de espacio de disco adicional édespus de desempaquetar.¿
Desea continuar? [S/n] S
```

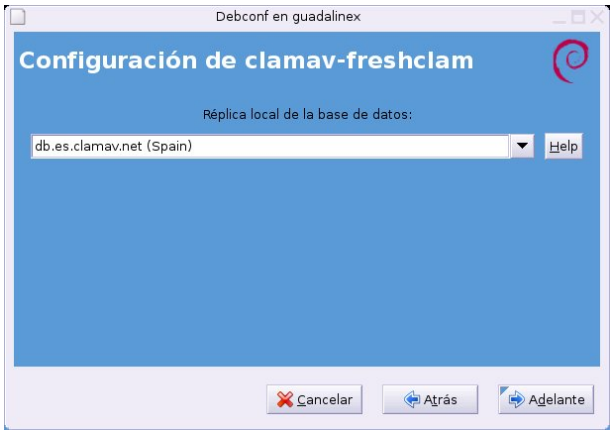
A continuación, instalaremos los paquetes propios de Clam Antivirus:

```
root@guadalinex:/home/mowgli# apt-get install clamav clamav-base clamav-
    daemon clamav-freshclam libclamav1
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se áinstalarn los siguientes paquetes extras:
libcurl3 libgmp3
Paquetes sugeridos:
daemon libcurl3-gssapi ca-certificates
Se áinstalarn los siguientes paquetes NUEVOS:
clamav clamav-base clamav-daemon clamav-freshclam libclamav1 libgmp3
Se áactualizarn los siguientes paquetes:
libcurl3
1 actualizados, 6 se áinstalarn, 0 para eliminar y 562 no actualizados.
Necesito descargar 3069kB de archivos.
Se áutilizarn 4592kB de espacio de disco adicional édespus de desempaquetar.¿
Desea continuar? [S/n] S
```

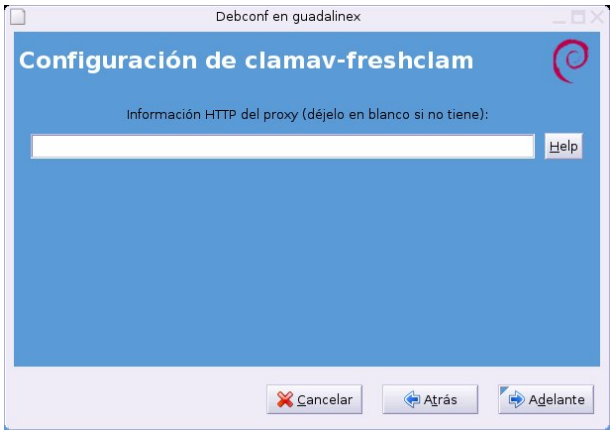
En este momento se inicia de manera automática el asistente de configuración de Clam Antivirus. Seleccionaremos como método de actualización de la base de datos de virus el método de "demonio" (*daemon*) y pulsaremos Adelante:



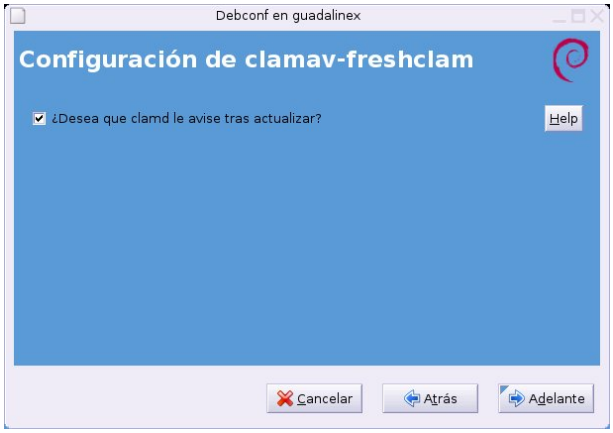
Como *mirror* para la descarga seleccionamos el más cercano a la localización geográfica de nuestro servidor, y pulsamos Adelante:



Si tenemos que usar un proxy para conectarnos a Internet deberemos configurarlo en esta pantalla, o nada con conexión directa a Internet y pulsamos Adelante:



Activamos la opción para que se nos avise tras la actualización y pulsamos **Adelante**, la instalación de los paquetes continuará por el siguiente punto:



Se inicia la instalación y configuración de los paquetes y se añade el usuario y grupo clamav. Por último, se inicia el demonio `clamd`.

Una vez concluida la instalación[6], no es preciso modificar ningún fichero de configuración, pues la instalación ya nos deja todo lo que necesitamos funcionando adecuadamente. Podemos mirar en el fichero `/etc/amavis/amavisd.conf`.

[6]Si todavía los paquetes Debian no se han actualizado a la última versión, puede salir un mensaje de que no es la versión más actual. Esperaremos a que se generen los paquetes y actualizamos con `apt-get`.

El modo de funcionamiento será el siguiente: primero Postfix, nuestro agente de transporte, recibirá los correos y a continuación se los pasará a Amavis para que detecte si contiene virus (dándole el trabajo a Clamav) o es un spam (en este caso el currito será Spamassassin).

Si observamos el siguiente error en el `/var/log/mail.log`:

```
linux amavis[13147]: (13147-01) Clam Antivirus-clamd FAILED - unknown status
    : /var/lib/amavis/amavis-20040818T163812-13147/parts: Access denied.
    ERROR\n
linux amavis[13147]: (13147-01) WARN: all primary virus scanners failed,
    considering backups
```

Entonces, deberemos añadir el usuario `clamav` al grupo `amavis`, una vez que se haya creado, de la siguiente forma:

```
#adduser clamav amavis
```

Y comprobamos que la directiva `AllowSupplementaryGroups` se encuentra en el fichero `/etc/-clamav/clamd.conf`.

Reiniciamos el servidor de correo `postfix`, `amavis` y `clamav`, o el sistema completo.

29.2.2. Probemos la medicina

Una vez instalados los tres paquetes, comprobemos que funciona correctamente.

Primero, con el comando `clamscan`, comprobamos que los ficheros o directorios que le especificamos, están libres de virus o puede que encontremos alguna sorpresa.

```
root@guadalinex:~/# clamscan -r -i -l vir.txt /root/
LibClamAV Warning: ***********************************************************
LibClamAV Warning: *** This version of the ClamAV engine is outdated. ***
LibClamAV Warning: *** DON'T PANIC! Read http://www.clamav.net/faq.html ***
LibClamAV Warning: ***********************************************************
/root/.mozilla/default/0jiyr4ub.slt/Mail/www.midominio.org/Sent: Worm.Gibe.F
    FOUND
/root/Desktop/Descargas/hsgejoq(1).exe: Worm.Gibe.F FOUND
/root/Desktop/Descargas/hsgejoq.exe: Worm.Gibe.F FOUND
/root/hsgejoq.exe: Worm.Gibe.F FOUND
---------- SCAN SUMMARY -----------
Known viruses: 34867
Engine version: 0.84
Scanned directories: 257
Scanned files: 1257
Infected files: 4
Data scanned: 57.43 MB
Time: 70.929 sec (1 m 10 s)
```

El comando anterior (`clamscan`) busca recursivamente (`-r`) a partir del directorio o fichero que le especificamos en último lugar (en este caso desde el home de `root` `/root`).

La opción `-i` indica que solamente nos muestra los ficheros infectados y la opción `-l` nos indica que guardará un informe en el fichero `resultado-virus.txt`. Puede que detectéis algún intruso en vuestro sistema, especialmente si compartís con sistemas windows vía Samba algún directorio.

Otro ejemplo de ejecución es el siguiente:

```
[root@linux tmp]# clamscan -i
/root/tmp/clamav-0.71.tar.gz: ClamAV-Test-Signature FOUND
/root/tmp/msg27986.zip: Worm.SomeFool.Q FOUND
---------- SCAN SUMMARY -----------
Known viruses: 21635
```

```
Scanned directories: 1
Scanned files: 19
Infected files: 2
Data scanned: 3.89 MB
I/O buffer size: 131072 bytes
Time: 6.577 sec (0 m 6 s)
```

En este caso ha detectado dos ficheros con virus y nos presenta el informe final.

También podemos utilizar el comando `clamdscan`.

```
root@guadalinex:~/curso-linux/entrega6/images# clamdscan /root/hsgejoq.exe
/root/hsgejoq.exe: Worm.Gibe.F FOUND
----------- SCAN SUMMARY -----------
Infected files: 1
Time: 0.152 sec (0 m 0 s)
```

29.2.3. Freshclam

El proceso `freshclam` se encarga de mantener actualizado el fichero de firma de virus. Podemos ejecutarlo directamente desde la línea de comandos, aunque lo mejor es incluirlo como tarea periódica (en el sistema `cron`) para que se actualice automáticamente.

El siguiente comando comprueba los ficheros de firmas de virus, y nos da como resultado que estamos actualizados.

```
[root@linux tmp]# freshclam
ClamAV update process started at Sat May 22 14:54:04 2004
Reading CVD header (main.cvd): OK
main.cvd is up to date (version: 23, sigs: 21096, f-level: 2, builder: ddm)
Reading CVD header (daily.cvd): OK
daily.cvd is up to date (version: 325, sigs: 539, f-level: 2, builder:
    ccordes)
```

En la dirección `http://news.gmane.org/gmane.comp.security.virus.clamav.virusdb`, podemos ver las versiones de los ficheros de firma de virus y si son los que nosotros tenemos.

En esta ejecución se produce una actualización del fichero `daily.cvd` a la versión 326.

```
ClamAV update process started at Sun May 23 11:02:31 2004
Reading CVD header (main.cvd): OK
main.cvd is up to date (version: 23, sigs: 21096, f-level: 2, builder: ddm)
Reading CVD header (daily.cvd): OK
Downloading daily.cvd [*]
daily.cvd updated (version: 326, sigs: 554, f-level: 2, builder: ccordes)
Database updated (21650 signatures) from database.clamav.net (80.69.67.3).
Clamd successfully notified about the update.
```

29.2.4. Funcionamiento

Para comprobar que está funcionando, mandemos un correo de prueba al usuario `alumno1@midominio.org`[7] con nuestro cliente de correo preferido. Éste es el resultado del correo una vez entregado, mostrando las cabeceras.

```
From alumno1@midominio.org Tue May 31 01:30:20 2005
Return-Path: <alumno1@midominio.org>
X-Original-To: alumno1@midominio.org
```

[7] O el dominio que tengamos configurado en nuestro sistema.

```
Delivered-To: alumno1@midominio.org
Received: from localhost (guadalinex [127.0.0.1])
by guadalinex (Postfix) with ESMTP id 8AA77A781
for <alumno1@midominio.org>; Tue, 31 May 2005 01:30:07 +0200 (CEST)
Received: from guadalinex ([127.0.0.1])
by localhost (www.midominio.org [127.0.0.1]) (amavisd-new, port 10024)
with LMTP id 08847-01 for <alumno1@midominio.org>;
Tue, 31 May 2005 01:28:59 +0200 (CEST)
Received: from [192.168.200.4] (unknown [192.168.200.4])
by guadalinex (Postfix) with ESMTP id 672DE97A8
for <alumno1@midominio.org>; Tue, 31 May 2005 00:35:39 +0200 (CEST)
Message-ID: <429B953A.7070901@midominio.org>
Date: Tue, 31 May 2005 00:35:38 +0200
From: alumno1 <alumno1@midominio.org>
User-Agent: Mozilla/5.0 (X11; U; Linux i686; es-ES; rv:1.7.6) Gecko/20050324
    Debian/1.7.6-1
X-Accept-Language: en
MIME-Version: 1.0
To: alumno1 <alumno1@midominio.org>
Subject: Re: df
References: <429B9356.6040009@midominio.org>
In-Reply-To: <429B9356.6040009@midominio.org>
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 8bit
X-Virus-Scanned: by amavisd-new-20030616-p10 (Debian) at midominio.org
Status: O
X-UID: 19
Content-Length: 27
X-Keywords:
alumno1 óescribi:
> Hola
```

Comentemos las líneas más interesantes. Primero el camino recorrido por el correo:

```
Received: from [192.168.200.4] (unknown [192.168.200.4])
by guadalinex (Postfix) with ESMTP id 672DE97A8
```

Estas líneas indican que la primera escala ha sido la recogida por parte de Postfix como agente de transporte de correo (puerto `25` de la máquina `192.168.200.4`)

```
Received: from guadalinex ([127.0.0.1])
by localhost (www.midominio.org [127.0.0.1]) (amavisd-new, port 10024)
with LMTP id 08847-01 for <alumno1@midominio.org>;
```

Postfix de nuestra máquina local (`127.0.0.1`) lo ha pasado al puerto `10024` que es donde está escuchando `amavisd-new`. Aquí se aplican las reglas especificadas en el fichero de configuración `/etc/amavis/amavisd.conf`.

```
Received: from localhost (guadalinex [127.0.0.1])
by guadalinex (Postfix) with ESMTP id 8AA77A781
for <alumno1@midominio.org>; Tue, 31 May 2005 01:30:07 +0200 (CEST)
```

Si nuestro valeroso correo pasa todas las pruebas de virus, spam y banned, llegará exitoso a los brazos de la persona destinataria[8].

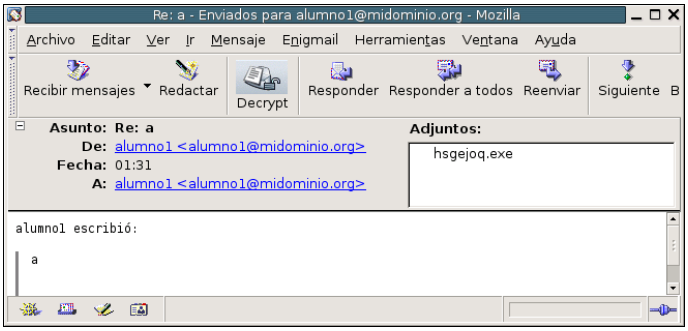
```
X-Virus-Scanned: by amavisd-new-20030616-p10 (Debian) at midominio.org
```

[8] Quién sabe si es que nos ha llegado la devolución de Hacienda. Para despidos o marrones, podemos aplicar algún filtrillo en Amavis ;-)

Nos dice que el correo ha sido escaneado y comprobado por `amavisd-new`.

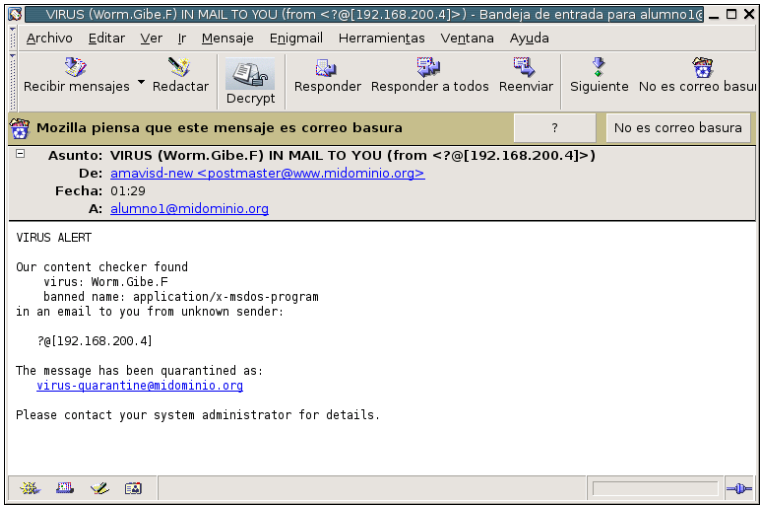
29.2.5. Ejemplo

Comprobemos cómo funciona enviando un virus de verdad. No os debe ser muy difícil capturar alguno, ¿verdad?. De un correo cogemos el fichero `hsgjoq.exe`, que contiene un virus. Lo guardamos y vamos a incluirlo como adjunto en un correo a nuestro sistema.



Lo enviamos y al usuario `postmaster` que esté definido en nuestro sistema le llegará un mensaje de aviso. Podemos personalizar qué hacer cuando se detecta un virus. Lo recomendable es que sólo se avise al `postmaster`, porque ayudará a ver de dónde procede el virus y desinfectar la máquina responsable. Al destinatario normalmente no le interesa saber que un virus iba dirigido a él (hay tantos). El supuesto remitente normalmente es engañoso, el virus se encarga de poner el remitente que le interese para no despertar sospechas y lo que conseguimos es que una persona que no tiene nada que ver, reciba un mensaje diciendo que ha enviado un virus, siendo falso en la mayoría de los casos.

En caso de que configuremos para que nos avise si un virus iba dirigido a nosotros, recibiremos un correo como éste.



Prácticas

Tipo I

E6-I-1

Realizar correctamente todas las preguntas del cuestionario que para esta entrega se ha habilitado en Moodle.

1. Para mantener un sistema seguro y estable es necesario garantizar 3 aspectos:

   a) Seguridad integrada y alta disponibilidad

   b) Integridad, severidad, seguridad

   c) Confidencialidad, integridad y disponibilidad

   d) Confidencialidad, seguridad e independencia

2. Una de las cuestiones más importantes a tener en cuenta desde el punto de vista de la seguridad es el cierre de servicios innecesarios. Para ello:

   a) Debemos revisar la configuración de inetd y cerrar aquellos servicios no utilizados.

   b) Debemos revisar la configuración de todos los demonios que corren en el sistema y detener los que no se utilicen.

   c) Debemos revisar los scripts de arranque existentes en /etc/rc.d y eliminar los que correspondan a servicios innecesarios.

   d) Debemos revisar la configuración de xinetd y cerrar aquellos servicios no utilizados.

3. En el caso de tener que escribir la contraseña en un papel, debido a su dificultad para recordarla, es imprescindible:

   a) No identificarla como contraseña ni hacer referencia al usuario o servidor al que se accede.

   b) Escribirla al revés, detallando el usuario y el servidor al que se accede.

   c) Escribirla en otro idioma para hacer necesaria su traducción.

   d) Pegarla en un postit en el monitor, nadie se fijará en ella.

4. ¿En qué fichero de configuración del sistema se definen los parámetros referentes a políticas de caducidad y cambio de claves?

   a) /etc/password.conf

   b) /etc/login.conf y /etc/default/useradd

   c) /etc/login.defs y /etc/default/useradd

   d) /etc/password y /etc/shadow

5. Un ataque de tipo DoS

   a) Es un tipo de ataque cuya meta fundamental es la de negar el acceso del atacado a un recurso determinado o a sus propios recursos.

   b) Es un tipo de ataque se se realiza desde ordenadores con el sistema operativo MSDOS instalado.

   c) Es un tipo de ataque cuya meta fundamental es dejar fuera de servicio los servidores con sistemas de archivos FAT16.

   d) Es un tipo de ataque cuya meta fundamental es dejar fuera de servicio los servidores de correo.

6. Para ocultar la información sobre la versión de apache así como los módulos que hay instalados usaremos las directivas:

   a) ServerShadow y ServerTokens

   b) Es imposible ocultar esta información

   c) DocumentRoot y ServerShadow

   d) ServerTokens

7. Nessus se compone de los siguientes elementos:

   a) Un único programa que efectúa los chequeos del sistema donde se encuentra instalado.

   b) Un cliente y un servidor que pueden ser instalados en cualquier sistema.

   c) Un cliente para los sistemas *nix y un servidor que puede ser instalado en cualquier sistema.

   d) Un cliente que puede ser instalado en cualquier sistema y un servidor para los sistemas *nix.

8. Uno de los puntos que hay que detectar para evitar intrusiones en un sistema son los barridos de puertos. Un barrido de puertos se detecta porque:

   a) Se producirán muchas conexiones casi simultáneas a una gran cantidad de puertos originadas desde la misma máquina.

   b) Se producirán muchas instancias del proceso port, saturando los recursos del sistema.

   c) Se producirán muchas conexiones casi simultáneas desde nuestro sistema a un sistema remoto.

   d) Se producirán muchas conexiones casi simultáneas al puerto 80.

9. El análisis forense de sistemas consiste en:

   a) Analiza el sistema después de haberse producido un fallo hardware.

   b) Analiza el sistema después de haberse producido un fallo software.

   c) Analiza el sistema después de haberse producido una intrusión.

   d) Analiza el sistema después de haberse producido un fallo de la corriente.

10. Para actualizar el fichero de firma de virus utilizamos la utilidad:

    a) freshclam

    b) clamscan -i

    c) clamscan -update

    d) clamscan -u

E6-I-2 Chequeo del sistema con Nessus

Utilizar la herramienta Nessus para hacer un chequeo de nuestro servidor. Para ello debemos instalar Nessus tal como se explica en los apuntes, instalando tanto el demonio `nessusd` como el cliente.

Para simplificar la instalación, no es necesario que realicemos una actualización de los plugins.

Una vez instalado arrancamos el demonio `nessusd` en segundo plano en una ventana de terminal:

```
nessusd -D &
```

A continuación arrancar el cliente:

```
nessus
```

Logarse con el usuario nessus que hemos creado en la instalación y realizar un chequeo de la misma máquina.

Como resultado de esta práctica será necesario enviar una copia de la pantalla con el resultado del chequeo. La captura gráfica, así como los prosibles comentarios se subirán en un fichero en formato OpenOffice, de nombre `e6-i-2.sxw`

Tipo II

E6-II-1 Jhon the ripper

Utilizar la herramienta Jhon the ripper para hacer un chequeo de las claves de nuestro servidor. Para ello debemos instalar la herramienta tal como aparece en los apuntes.

Se pide localizar la clave de, al menos, un usuario. Para ello (usando nuestros privilegios de administrador del sistema) haremos un chequeo de claves utilizando las palabras almacenadas en el fichero `password.lst`, previamente modificado.

Como resultado de la práctica será necesario enviar el proceso seguido para realizar el chequeo de claves, incluyendo los parámetros utilizados y los ficheros de configuración modificados. El fichero a subir tendrá por nombre `e6-ii-1.sxw`

E6-II-2 ClamAV

Instalar el antivirus ClamAV, integrándolo con un agente de transporte (postfix, por ejemplo). Para comprobar que funciona, debéis mandar un correo a un usuario local de vuestro sistema (no hace falta que estéis en Internet), incorporando como archivo adjunto el fichero `hsqejoq.exe`.

Como resultado, debéis subir en un fichero de nombre `e6-ii-2.sxw` el correo que se envía al postmaster de vuestro sistema, indicando que se ha detectado un virus, así como cualquier otro comentario que estiméis oportuno.

Bibliografía

[1] *Seguridad en Unix y redes v2.0* Antonio Villalón Huerta

[2] *Seguridad práctica en Unix e Internet* Garfinkel y Spafford

[3] *Hacking Exposed Linux* Brian Hatch y James Lee

[4] Información de la NSA sobre SELinux en http://www.nsa.gov/selinux/

[5] HOWTO - Getting Started with new SELinux

Parte VII

Apéndices

Apéndice A

Soluciones a las prácticas

En este apartado hemos incluido las soluciones a los cuestionarios y a las prácticas de las distintas entregas. Solo adjuntamos las soluciones que no son inmediatas, no así aquellas que consisten en una captura gráfica o se encuentran totalmente detalladas en los mismos apuntes. Comentar también que para las prácticas sobre Amanda y Nagios, para una mejor comprensión de las mismas y dada su extensión, hemos optado por incluir los ficheros completos en el CD de los cursos.

Soluciones a los cuestionarios

E1-I-1

1. a	3. d	5. d	7. b	9. c
2. c	4. a	6. d	8. d	10. a

E2-I-1

1. d	3. b	5. d	7. c	9. b
2. c	4. b	6. a	8. a	10. a

E3-I-1

1. a	3. c	5. d	7. b	9. a
2. a	4. b	6. a	8. d	10. b

E4-I-1

1. a	3. a	5. a	7. a	9. b
2. a	4. a	6. b	8. a	10. b

E5-I-1

1. d	3. d	5. c	7. b	9. c
2. c	4. c	6. a	8. a	10. b

E6-I-1

1. c	3. a	5. a	7. d	9. c
2. b	4. c	6. d	8. a	10. a

Soluciones a las prácticas de la 1ª entrega

E1-I-2

La principal diferencia entre los comandos `netstat` y `nmap` es que el primero (`netstat`) tiene una visión interna de lo que ocurre con las conexiones de nuestra máquina, mientras que el comando `nmap` nos muestra una visión desde el exterior.

Comando `nmap`:

```
#nmap -sTU localhost
Starting nmap 3.55 ("http://www.insecure.org/nmap/" tar-
get=newpage>http://www.insecure.org/nmap/) at 2005-04-03 13:00 CEST
Interesting ports on guadalinex (127.0.0.1):
(The 3131 ports scanned but not shown below are in state: closed)
PORT STATE SERVICE
22/tcp open ssh
137/udp open netbios-ns
138/udp open netbios-dgm
139/tcp open netbios-ssn
445/tcp open microsoft-ds
631/tcp open ipp
631/udp open unknown
Nmap run completed -- 1 IP address (1 host up) scanned in 4.056 seconds
```

Se observan todos los puertos (tanto tcp como udp) que están a la espera de recibir conexiones. Esta es la vista desde el exterior de los servicios que ofrece nuestro servidor. Debe verse, como indica el enunciado, que el puerto 22 (ssh) se encuentra abierto para recibir conexiones.

Comando `netstat`

```
#netstat -atu
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address Foreign Address State
tcp 0 0 *:netbios-ssn *:* LISTEN
tcp 0 0 *:ipp *:* LISTEN
tcp 0 0 *:microsoft-ds *:* LISTEN
tcp 0 0 guadalinex:32774 guadalinex:ssh ESTABLISHED
tcp 0 0 *:ssh *:* LISTEN
tcp 0 0 guadalinex:ssh guadalinex:32774 ESTABLISHED
udp 0 6160 192.168.1.4:netbios-ns *:*
udp 0 0 *:netbios-ns *:*
udp 0 1692 192.168.1.4:netbios-dgm *:*
udp 0 0 *:netbios-dgm *:*
udp 0 0 *:ipp *:*
```

Esta es la vista interna de nuestro servidor. Aparecen los puertos que están a la espera, como podíamos ver con `nmap`, pero además vemos las conexiones establecidas.

Como caso particular, la conexión ssh que habíamos establecido, muestra los dos sockets. Uno es del puerto 22 (ssh) al 32774, y el otro es del puerto 32774 al puerto 22.

E1-II-1

La sesión telnet ha sido iniciada por el usuario `legolas` con password `elfo`.
Los comandos ejecutados son:

```
ls
who
su (password administrador)
ls -la
exit
exit
```

La sesión de correo electrónico nos muestra un correo con origen `legolas@sid` y destino `hugo@sid`.

```
Asunto: Los fantasmas del Windsor utilizaban Windows, y que mostra-
mos a continuación.
Cuerpo:
Hola:.
Te voy a decir quienes son los fantasmas del Windsor ... aun-
que no se si.
alguien podría leer este mensaje ... mejor te lo digo en persona..
Un saludo.
```

La sesión ssh nos proporciona los siguientes datos:

```
mac origen: 00:c0:49:d6:cf:c9
mac destino: 00:c0:29:49:a2:a9
ip origen: 192.168.0.13
ip destino: 192.168.0.50
puertos origen: 1803
puertos destino: 22(ssh)
```

E1-II-2

Instalar un Firewall Personal
El método usado se basa en el método clásico de configuración de iptables, ya que las otras alternativas requieren de un entorno gráfico del que no siempre se va a disponer, este método consiste en crear un script de sh que será colocado en la carpeta `/etc/rc2.d` (donde se contienen los scripts que se ejecutarán al entrar al nivel de ejecución 2, correspondiente al modo texto de linux.).
Este script contendrá los comandos a ejecutar para implementar las siguientes políticas.

- Hacia el interior (eth0) la política será permitir todas.
- Hacia el exterior (eth1) la política será denegar todas las entrantes excepto los servicios http 80 (UDP y TCP), https 443 (UDP y TCP), y smtp 25 (UDP y TCP).
- Entre el interior y el exterior la política para compartir la conexión de internet, para tener acceso desde la red local.

Accedemos con el usuario root para así poder escribir en las configuraciones del directorio `/etc/`.

- abrimos el editor: `# nano /etc/rc2.d/S99iptables`
- el prefijo S99 se coloca para especificar el orden en que queremos que se ejecute, ya que la carga de los scripts se realiza por orden alfabético y se quiere colocar la política del *firewall* al final.

```
<----- Comienzo del script S99iptables ----->
#!/bin/sh
# Con estos comandos se borran las reglas anteriores que hayan podi-
do ser establecidas por otros scripts
iptables -F
iptables -X
iptables -Z
iptables -t nat -F
# Aceptamos todas las conexiones aunque después las vamos a filtrar
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD ACCEPT
iptables -t nat -P PREROUTING ACCEPT
iptables -t nat -P POSTROUTING ACCEPT
# Permitimos al servidor el acceso a su propia direc-
ción de red (127.0.0.1) interfaz lo
iptables -A INPUT -i lo -j ACCEPT
# Permitimos acceso desde la red local en todos los puertos
iptables -A INPUT -s 192.168.5.0/24 -i eth0 -j ACCEPT
# Permitimos compartir la conexion desde la red local y lo redireccio-
namos a internet eth1
iptables -t nat -A POSTROUTING -s 192.168.5.0/24 -o eth1 -j MASQUERADE
# Permitimos el redireccionamiento activando el forward
echo 1 > /proc/sys/net/ipv4/ip_forward
# Aceptamos todo el trafico de http
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 80 -j ACCEPT
# Aceptamos todo el trafico de https
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 443 -j ACCEPT
# Aceptamos todo el trafico para el servidor SMTP
iptables -A INPUT -p tcp --dport 25 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 25 -j ACCEPT
<----- Final del script de iptables ----->
```

una vez guardado en `/etc/rc2.d/` con el nombre `S99redinit`. Se le establecen privilegios de ejecucion a este con el comando `chmod` y listo.

Soluciones a las prácticas de la 2ª entrega

E2-I-2

1. Para el servicio `http://www.juntadeandalucia.es`, obtener su dirección IP y el nombre al que le corresponde el registro de tipo A.

```
#dig www.juntadeandalucia.es
; <<>> DiG 9.2.4rc5 <<>> www.juntadeandalucia.es
;; global options: printcmd
;; Got answer:
;; >>HEADER<< - opcode: QUERY, status: NOERROR, id: 10124
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 3, ADDITIO-
NAL: 3
;; QUESTION SECTION:
;www.juntadeandalucia.es. IN A
```

```
;; ANSWER SECTION:
www.juntadeandalucia.es. 172152 IN A 217.12.16.222
www.juntadeandalucia.es. 172152 IN A 217.12.16.221
;; AUTHORITY SECTION:
juntadeandalucia.es. 172152 IN NS ns.juntadeandalucia.es.
juntadeandalucia.es. 172152 IN NS ns1.juntadeandalucia.es.
juntadeandalucia.es. 172152 IN NS nso.nic.es.
;; ADDITIONAL SECTION:
ns.juntadeandalucia.es. 171311 IN A 217.12.16.33
ns1.juntadeandalucia.es. 171311 IN A 217.12.16.34
nso.nic.es. 705 IN A 194.69.254.2
;; Query time: 1276 msec
;; SERVER: 62.37.228.20#53(62.37.228.20)
;; WHEN: Mon Aug 1 15:26:03 2005
;; MSG SIZE rcvd: 178
```

Existen dos direcciones IP que responden al nombre dns `www.juntadeandalucia.es`. Es normal para conseguir una mayor disponibilidad.

Estas direcciones son 217.12.16.222 y 217.12.16.221.

Esta respuesta varió durante el curso y a los primeros que realizaron la práctica, la salida correspondiente era:

```
;; ANSWER SECTION:
www.juntadeandalucia.es. 1229 IN CNAME inv.juntadeandalucia.es.
inv.juntadeandalucia.es. 1048 IN A 217.12.16.37
```

2. Nombres y direcciones IP de los intercambiadores de correo y servidores de nombres para el dominio `juntadeandalucia.es`.

   Le preguntamos al comando dig por todos los registros de dominio (también podíamos haber preguntado específicamente por los registros de tipo MX y DNS)

```
#dig any juntadeandalucia.es
; <<>> DiG 9.2.4rc5 <<>> any juntadeandalucia.es
;; global options: printcmd
;; Got answer:
;; >>HEADER<<- opcode: QUERY, status: NOERROR, id: 11050
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 3, ADDITIO-
NAL: 3
;; QUESTION SECTION:
;juntadeandalucia.es. IN ANY
;; ANSWER SECTION:
juntadeandalucia.es. 171858 IN NS nso.nic.es.
juntadeandalucia.es. 171858 IN NS ns.juntadeandalucia.es.
juntadeandalucia.es. 171858 IN NS ns1.juntadeandalucia.es.
juntadeandalucia.es. 72660 IN MX 10 mx.juntadeandalucia.es.
juntadeandalucia.es. 72660 IN MX 20 mx2.juntadeandalucia.es.
;; AUTHORITY SECTION:
juntadeandalucia.es. 171858 IN NS nso.nic.es.
juntadeandalucia.es. 171858 IN NS ns.juntadeandalucia.es.
juntadeandalucia.es. 171858 IN NS ns1.juntadeandalucia.es.
;; ADDITIONAL SECTION:
ns.juntadeandalucia.es. 171017 IN A 217.12.16.33
ns1.juntadeandalucia.es. 171017 IN A 217.12.16.34
nso.nic.es. 411 IN A 194.69.254.2
```

```
;; Query time: 156 msec
;; SERVER: 62.37.228.20#53(62.37.228.20)
;; WHEN: Mon Aug 1 15:30:57 2005
;; MSG SIZE rcvd: 223
```

Los intercambiadores de correo son (sus IP las averiguamos con los comandos "dig mx.juntadeandalucia.es" y "dig mx2.juntadeandalucia.es"):

```
mx.juntadeandalucia.es (217.12.17.247)
mx2.juntadeandalucia.es (217.12.17.247)
```

Curioso que sean la misma dirección IP. Puede tener un balanceador de carga y estar varias máquinas a la escucha.

Los servidores de nombres son:

```
ns.juntadeandalucia.es (217.12.16.33)
ns1.juntadeandalucia.es (217.12.16.34)
nso.nic.es (194.69.254.2)
```

E2-II-1

Presentamos solo las modificaciones básicas en el fichero `/etc/cups/cupsd.conf` para realizar la práctica:

```
...
Listen 80.32.123.200:631
#Especificaciones básicas
<Location />
    Order Deny,Allow
    #Define el control de acceso por defecto
    #Deniegas a todos
    Deny From All
    #Permites a lo y la red local
    Allow From 127.0.0.1
    #De igual forma podríamos-
    haber usado 192.168.0.0/24 o 192.168.0.*
    Allow From @LOCAL
    #Permitimos el acceso la IP externa
    Allow From 80.32.134.123
</Location>
# Ahora la administración
<Location /admin>
    AuthType Basic
    AuthClass System
    Order Deny,Allow
    Deny From All
    Allow From 127.0.0.1
    # Y para permitir la administración desde la ip que nos propo-
    ne la práctica
    Allow From 80.32.134.123
</Location>
# Impresora color
<Location /printers/color>
```

```
    # Tipo y nivel de autenticación (ninguno)
    AuthType none
    Order Deny,Allow
    Deny From All
    Allow From 127.0.0.1
    #Permitimos imprimir desde la red local
    Allow From @LOCAL
</location>
#Especificaciones para la impresora laser
<Location /printers/laser>
    #Usamos autenticación básica y a nivel de usuario
    AuthType Basic
    AuthClass User
    Order Deny,Allow
    Deny From All
    Allow From 127.0.0.1
    # Permitimos imprimir en la láser desde la IP remota
    Allow From 80.32.134.123
</location>
```

También hay que cambiar la configuración del fichero `/etc/cups/printers.conf`

```
# Printer configuration file for CUPS v1.1.17
# Written by cupsd on dom 27 oct 2005 00:10:36 GMT
<defaultprinter color>
    DeviceURI paralell:/dev/lp0
    State Idle
    Accepting Yes
    JobSheets none none
    QuotaPeriod 0
    PageLimit 0
    KLimit 0
</defaultprinter>
<printer laser>
    DeviceURI paralell:/dev/lp1
    State Idle
    Accepting Yes
    JobSheets none none
    QuotaPeriod 0
    PageLimit 0
    KLimit 0
    #Deniega la impresión a  'Muchacara'
    DenyUser muchacara
</printer>
```

Soluciones a las prácticas de la 3ª entrega

E3-II-1

Daremos la solución para Debian, para Fedora no presenta mayor dificultad. Partiremos de la idea de que Apache está instalado y que funciona perfectamente.

Host Virtuales

1. Modificamos el fichero `/etc/hosts` con la línea:

```
127.0.0.1 matematicas.micentro.org lenguaje.micentro.org
```

2. Es el momento de crear los directorios donde se publicarán los contenidos de los host virtuales

```
# mkdir /var/www/matematicas
# mkdir /var/www/lenguaje
```

3. Creemos los ficheros de configuración de los host virtuales. Los ficheros de error no es obligatorio incluirlos en la configuración.

```
#cat /etc/apache2/sites-available/matematicas
NameVirtualHost 127.0.0.1
<VirtualHost matematicas.micentro.org>
 ServerAdmin webmaster@micentro.org
 DocumentRoot /var/www/matematicas
 ServerName matematicas.micentro.org
 ErrorLog /var/log/apache2/matematicas-error_log
 CustomLog /var/log/apache2/matematicas-access_log common
</VirtualHost>
#cat /etc/apache2/sites-available/lenguaje
NameVirtualHost 192.168.1.250
<VirtualHost lenguaje.micentro.org>
 ServerAdmin webmaster@micentro.org
 DocumentRoot /var/www/lenguaje
 ServerName lenguaje.micentro.org
 ErrorLog /var/log/apache2/lenguaje-error_log
 CustomLog /var/log/apache2/lenguaje-access_log common
</VirtualHost>
```

4. Activémoslos y reiniciemos apache

```
# a2ensite matematicas
# a2ensite lenguaje
# /etc/init.d/apache2 reload
```

Autenticación

1. En primer lugar creemos el directorio en donde almacenar las claves de acceso.

```
# mkdir /var/www/passwd
```

2. Es el momento de crear los usuarios. Notar que el parámetro `-c` sólo se escribe una vez

```
# cd /var/www/passwd
# htpasswd2 -c .htpasswd thales
# htpasswd2 .htpasswd mileto
# htpasswd2 .htpasswd pitagoras
# htpasswd2 .htpasswd quevedo
# htpasswd2 .htpasswd cervantes
```

3. Creamos el fichero dptos para definir los grupos

```
#cat dptos
mate: thales mileto pitagoras
lengua: quevedo cervantes
```

4. Ya casi, es el momento de crear los ficheros `.htaccess`

```
#cat /var/www/matematicas/.htaccess
 AuthType Basic
 AuthName "Página de matemáticas"
 AuthUserFile /var/www/passwd/.htpasswd
 AuthGroupFile /var/www/passwd/dptos
 require group mate

#cat /var/www/lenguaje.htaccess
 AuthType Basic
 AuthName "Página de lenguaje"
 AuthUserFile /var/www/passwd/.htpasswd
 AuthGroupFile /var/www/passwd/dptos
 require group lengua
```

La captura gráfica es ya un "juego de niños"

Soluciones a las prácticas de la 4ª entrega

E4-I-2

1.

- Crear una base de datos de nombre alumnos.

```
CREATE DATABASE ‘alumnos‘;
```

- Crear una tabla de nombre datos en la base de datos alumnos con sólo cuatro campos.

```
USE alumnos;
CREATE TABLE ‘datos‘ (
‘nombre‘ VARCHAR( 20 ) NOT NULL ,
‘apellidos‘ VARCHAR( 30 ) NOT NULL ,
‘edad‘ SMALLINT( 2 ) UNSIGNED DEFAULT ’14’ NOT NULL ,
‘nif‘ CHAR( 9 ) NOT NULL ,
INDEX ( ‘nombre‘ , ‘apellidos‘ )
);
```

- Obtener un listado ordenado por nombre y apellidos de los datos de la tabla.

```
SELECT * FROM datos ORDER BY nombre, apellidos;
```

- Borrar la tabla.

```
DROP TABLE datos;
```

- Borrar la base de datos.

```
DROP DATABASE alumnos;
```

2.

```
<?php
  echo "<center><H1>La Tabla del ocho.</H1></center>";
  echo "<br>";
```

```
      echo "<table align = center border = 2>";
      echo "<tr align = center><td>Tabla</td><td>Resultado</td></tr>";
      for ($i = 0; $i < 10; $i ++)
        {
          echo "<tr align = center>";
          echo "<td> 8 x ".($i+1) ."</td>";
          echo "<td>".(($i+1) * 8)."</td>";
          echo "</tr>";
        }
      echo "</table>";
    ?>
```

E4-II-2

forumulario.html

```
    <html>
     <head>
       <title>Formulario de entrada de registros</title>
     </head>
     <body>
      <center><h1>FICHA DEL ALUMNO</h1></center>
      <form action = 'altas.php' method = 'post' >
       <table border=0>
        <tr>
          <td>NOMBRE: </td>
          <td><input type = "text" name = "nombre" size = 20 ></td>
         <tr>
           <td>APELLIDOS: </td>
           <td><input type = "text" name = "apellidos" size =30 ></td>
         <tr>
           <td>EDAD: </td>
           <td><input type = "text" name = "edad" size = 2 > </td>
          <tr>
            <td>NIF: </td>
            <td><input type = "text" name = "nif" size = 9 > </td>
          <tr>
            <td><input type = "submit" value= "ENVIAR" ></td>
            <td><input type = "reset" value = "BORRAR" ></td>
       </table>
      </form>
     </body>
    </html>
```

altas.php

```
    <?
      #Recogemos en una variable el nombre de la BASE DE DATOS
      $base="alumnos";

      #Recogemos en una variable el nombre de la TABLA
      $tabla="datos";

      #Recoger y adaptar las variables pasadas desde el formulario
      $v1=$HTTP_POST_VARS['nombre'];
```

```
  $v2=$HTTP_POST_VARS['apellidos'];
  $v3=$HTTP_POST_VARS['edad'];
  $v4=$HTTP_POST_VARS['nif'];

  #Establecemos la conexión con el servidor
  $c=mysql_connect("localhost","contraseña");

  #Asignamos la conexión a una base de datos determinada
  mysql_select_db($base,$c);

  # Añadimos el nuevo registro
  mysql_query("INSERT $tabla (nombre,apellidos,edad,nif) VA-
LUES('$v1','$v2','$v3','$v4')",$c);

  # cerramos la conexion
  mysql_close();
?>
```

Apéndice B

httpd.conf

Material adicional de la 3ª entrega

```
#
# Basado en los archivos de configuración del servidor NACSA, por Rob McCool.
#
# Este es el archivo de configuración principal del servidor Apache.
# Contiene las directivas de configuración que proporcionan al servidor sus
# instrucciones.
# Véase <URL:http://httpd.apache.org/docs-2.0/>para una información
# detallada sobre las directivas.
#
# NO trate sólo de leer las instrucciones sin comprender lo que hacen. Sólo
# son pistas o recordatorios. Si no está seguro, consulte los documentos
# en línea. Está avisado.
#
# Las directivas de configuración se agrupan en tres secciones básicas:
# 1. Directivas que controlan el funcionamiento del proceso del servidor
# Apache como un todo (el "entorno global")
# 2. Directivas que definen los parámetros del servidor "principal" o
# "predeterminado", que responde a las solicitudes que no maneja un
# host virtual.
# Estas directivas también ofrecen valores predeterminados como
# parámetros de todos los hosts virtuales.
# 3. Parámetros de hosts virtuales, que permiten enviar solicitudes Web
# a las distintas direcciones IP o nombres de host y que las maneje
# el mismo proceso del servidor Apache.
#
# Configuración y nombres de archivo de registro: si los nombres de archivo
# especificados para muchos de los archivos de control del servidor
# empiezan por "/" (o "drive:/" en Win32), el servidor usará esa ruta explícita.
# Si los nombres de archivo *no* empiezan por "/", el valor del ServerRoot
# se antepondrá -- así "logs/foo.log" con ServerRoot establecido a
# "/etc/httpd" será interpretado por el servidor como
# "/etc/httpd/logs/foo.log".
#
### Sección 1: Entorno global
#
# Las directivas de esta sección afectan al funcionamiento general de Apache,
# por ejemplo al número de solicitudes simultáneas que puede manejar o dónde
# puede encontrar sus archivos de configuración.
#
#
```

```
# No revele demasiada información sobre todos los subcomponentes que está
# utilizando. Descomente esta línea si no le importa que descubran desde otros
# sitios remotos cuáles son los módulos opcionales más importantes que está ejecutando
ServerTokens OS
#
# ServerRoot: La parte superior del árbol de directorios, donde se mantienen
# la configuración, los errores y los archivos de registro del servidor.
#
# ¡NOTA! Si pretende colocar ésto en una NFS (u otra red)
# monte el sistema de archivos y lea la documentación del LockFile
# (disponible en <URL:http://httpd.apache.org/docs-2.0/mod/core.html#lockfile>);
# se ahorrará muchos problemas.
#
# NO añada una barra al final de la ruta de directorio.
#
ServerRoot "/etc/httpd"
#
# ScoreBoardFile: Archivo utilizado para almacenar información interna del proceso
# del servidor. Si no está especificado (por defecto), el scoreboard se
# almacenará en un segmento anónimo de memoria compartida, y no estará
# disponible para terceras aplicaciones.
# Si está especificado, asegúrese de que no haya dos invocaciones de Apache
# compartiendo el mismo archivo scoreboard. El archivo scoreboard DEBE
# ALMACENARSE EN UN DISCO LOCAL.
#
#ScoreBoardFile run/httpd.scoreboard
#
# PidFile: El archivo donde el servidor debe registrar su número de
# identificación de proceso al arrancar.
#
PidFile run/httpd.pid
#
# Timeout: El número de segundos que transcurre antes de que se agoten las
# recepciones y los envíos.
#
Timeout 300
#
# KeepAlive: Permitir o no conexiones persistentes (más de una solicitud por conexión)
# Ajústelo a "Off" para desactivarlo.
#
KeepAlive Off
#
# MaxKeepAliveRequests: El número máximo de solicitudes a permitir durante
# una conexión persistente. Establézcalo a 0 para permitir una cantidad ilimitada.
# Se recomienda que este número sea alto para un rendimiento óptimo.
#
MaxKeepAliveRequests 100
#
# KeepAliveTimeout: Número de segundos a esperar para la nueva solicitud
# del mismo cliente en la misma conexión.
#
KeepAliveTimeout 15
##
## Server-Pool Size Regulation (MPM specific)
##
# prefork MPM
# StartServers: número de procesos del servidor con el que iniciar
# MinSpareServers: número mínimo de procesos del servidor que se mantienen
```

```
#        en espera
# MaxSpareServers: número máximo de procesos del servidor en espera
# MaxClients: número máximo de procesos del servidor permitidos para iniciar
# MaxRequestsPerChild: número máximo de solicitudes que puede procesar un
#        servidor hijo
<IfModule prefork.c>
StartServers 8
MinSpareServers 5
MaxSpareServers 20
MaxClients 150
MaxRequestsPerChild 1000
</IfModule>
# worker MPM
# StartServers: número inicial de procesos del servidor para iniciar
# MaxClients: número máximo de conexiones simultáneas de clientes
# MinSpareThreads: número mínimo de hilos parados
# MaxSpareThreads: número máximo de hilos parados
# ThreadsPerChild: número constante de hilos en cada proceso del servidor
# MaxRequestsPerChild: número máximo de solicitudes que puede procesar un
#         servidor hijo
<IfModule worker.c>
StartServers 2
MaxClients 150
MinSpareThreads 25
MaxSpareThreads 75
ThreadsPerChild 25
MaxRequestsPerChild 0
</IfModule>
# perchild MPM
# NumServers: número constante de procesos del servidor
# StartThreads: número inicial de hilos en cada proceso del servidor
# MinSpareThreads: número mínimo de hilos parados
# MaxSpareThreads: número máximo de hilos parados
# MaxThreadsPerChild: número máximo de hilos en cada proceso hijo
# MaxRequestsPerChild: número máximo de solicitudes que puede procesar un
#         servidor hijo
<IfModule perchild.c>
NumServers 5
StartThreads 5
MinSpareThreads 5
MaxSpareThreads 10
MaxThreadsPerChild 20
MaxRequestsPerChild 0
</IfModule>
#
# Listen: Permite enlazar Apache con direcciones IP específicas y/o
# puertos, aparte de los valores predeterminados. Ver también la directiva <VirtualHost>
#
# Cambie esto a Listen sobre unas direcciones IP específicas, como
# se muestra debajo, para evitar que Apache se quede cogido en un
# rango de direcciones IP (0.0.0.0)
#
#Listen 12.34.56.78:80
Listen 80
#
# Soporte para Objetos Dinámicos Compartidos (DSO)
#
# Para poder usar la funcionalidad de un módulo que fue construido como un DSO,
```

```
# hay que colocar las líneas correspondientes 'LoadModule' en esta ubicación,
# para que las directivas contenidas en él estén disponibles antes de ser usadas.
# Los módulos compilados estáticamente (aquellos listados por 'httpd -l') no
# necesitan ser cargados aquí.
#
# Ejemplo:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule access_module modules/mod_access.so
LoadModule auth_module modules/mod_auth.so
LoadModule auth_anon_module modules/mod_auth_anon.so
LoadModule auth_dbm_module modules/mod_auth_dbm.so
LoadModule auth_digest_module modules/mod_auth_digest.so
LoadModule ldap_module modules/mod_ldap.so
LoadModule auth_ldap_module modules/mod_auth_ldap.so
LoadModule include_module modules/mod_include.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule env_module modules/mod_env.so
LoadModule mime_magic_module modules/mod_mime_magic.so
LoadModule cern_meta_module modules/mod_cern_meta.so
LoadModule expires_module modules/mod_expires.so
LoadModule deflate_module modules/mod_deflate.so
LoadModule headers_module modules/mod_headers.so
LoadModule usertrack_module modules/mod_usertrack.so
LoadModule unique_id_module modules/mod_unique_id.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule mime_module modules/mod_mime.so
LoadModule dav_module modules/mod_dav.so
LoadModule status_module modules/mod_status.so
LoadModule autoindex_module modules/mod_autoindex.so
LoadModule asis_module modules/mod_asis.so
LoadModule info_module modules/mod_info.so
LoadModule dav_fs_module modules/mod_dav_fs.so
LoadModule vhost_alias_module modules/mod_vhost_alias.so
LoadModule negotiation_module modules/mod_negotiation.so
LoadModule dir_module modules/mod_dir.so
LoadModule imap_module modules/mod_imap.so
LoadModule actions_module modules/mod_actions.so
LoadModule speling_module modules/mod_speling.so
LoadModule userdir_module modules/mod_userdir.so
LoadModule alias_module modules/mod_alias.so
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_ftp_module modules/mod_proxy_ftp.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule cache_module modules/mod_cache.so
LoadModule suexec_module modules/mod_suexec.so
LoadModule disk_cache_module modules/mod_disk_cache.so
LoadModule file_cache_module modules/mod_file_cache.so
LoadModule mem_cache_module modules/mod_mem_cache.so
LoadModule cgi_module modules/mod_cgi.so
#
# Carga archivos de configuración desde el directorio de configuración "/etc/httpd/conf.d".
#
Include conf.d/*.conf
#
# ExtendedStatus controla si Apache va a generar información de estado
```

```
# "total" (ExtendedStatus On) o sólo información elemental (ExtendedStatus
# Off) cuando se llama al manipulador "server-status". Por defecto está en Off.
#
#ExtendedStatus On
### Section 2: Configuración del servidor 'principal'
#
# las directivas de esta sección configuran los valores utilizados por el
# servidor 'principal', que responde a cualquier solicitud que no se
# se maneje por una definición <VirtualHost>. Estos valores también
# proporcionan valores predeterminados para los contenedores <VirtualHost>
# que pueda definir luego en el archivo.
#
# Todas estas directivas pueden aparecer dentro de los contenedores <VirtualHost>,
# en cuyo caso, estos valores predeterminados serán omitidos en el host
# virtual que se está definiendo.
#
#
# Si desea que httpd se ejecute como usuario o grupo diferente, deberá ejecutar
# httpd como root inicialmente, y luego cambiará.
#
# User/Group: El nombre (o #número) del usuario/grupo bajo el que se ejecuta httpd
# . En SCO (ODT 3) use "User nouser" y "Group nogroup".
# . En HPUX es posible que no pueda usar memoria compartida como nadie,
# y la solución sugerida consiste en crear un www de usuario y usarlo.
# OBSERVE que algunos kernels rechazan setgid(Group) o semctl(IPC_SET)
# cuando el valor de (sin firma)Group es superior a 60000;
# ¡no use Group #-1 en estos sistemas!
#
User apache
Group apache
#
# ServerAdmin: Su dirección, donde los problemas con el servidor se deben
# enviar por correo-e. Esta dirección aparece en alguna páginas generadas por
# el servidor, tales como documentos de error. e.g. admin@your-domain.com
#
ServerAdmin root@localhost
#
# ServerName proporciona el nombre y puerto que usa el servidor para identificarse
# él mismo. A menudo puede determinarse automáticamente, pero recomendamos que lo
# especifique explícitamente para evitar problemas durante el arranque.
#
# Si no da un nombre DNS name válido para su host, el redireccionamiento
# generado por el servidor no funcionará. Vea también la directiva UseCanonicalName.
#
# Si su host no tiene un nombre DNS registrado, introduzca aquí su dirección IP
# Tendrá que acceder a ella por su dirección, y esto hará que los redireccionamientos
# funcionen de forma sensible.
#
#ServerName new.host.name:80
#
# UseCanonicalName: Determina cómo Apache construye las URLs de auto-referencia
# y las variables SERVER_NAME y SERVER_PORT.
# Cuando está en "Off", Apache usará el Hostname y Port proporcionado
# por el cliente. Cuando está en "On", Apache usará el valor de la directiva
# ServerName.
#
UseCanonicalName Off
#
```

```
# DocumentRoot: El directorio desde el que se sirven los documentos.
# Por defecto, todas las solicitudes se toman de este directorio, pero los
# vínculos simbólicos y los alias se pueden usar para señalar a otras ubicaciones.
#
DocumentRoot "/var/www/html"
#
# Cada directorio al que tiene acceso Apache puede configurarse con respecto
# a qué servicios y opciones están permitidos y/o desactivados en ese
# directorio (y sus subdirectorios).
#
# Primero, vamos a establecer que la configuración predeterminada sea un
# conjunto restrictivo de permisos.
#
<Directory />
Options FollowSymLinks
AllowOverride None
</Directory>
#
# Observe que, de aquí en adelante, hay que permitir específicamente
# la activación de características concretas, por lo que si algo no
# funciona como podría esperar, asegúrese de que lo ha activado específicamente.
#
#
# Esto debe cambiarse a lo que se establezca para DocumentRoot.
#
<Directory "/var/www/html">
#
# Los posibles valores para la directiva Options son "None", "All",
# o cualquier combinación de:
# Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI Multiviews
#
# Observe que "MultiViews" debe ser *explícitamente* llamado --- "Options All"
# no se lo proporciona.
#
# La directiva Options es a la vez complicada e importante. Por favor, vea
# http://httpd.apache.org/docs-2.0/mod/core.html#options
# para más información.
#
Options Indexes FollowSymLinks
#
# AllowOverride controla qué directivas pueden situarse en los archivos .htaccess.
# Puede ser "All", "None", o cualquier combinación de:
# Options FileInfo AuthConfig Limit
#
AllowOverride None
#
# Controla quién puede obtener la respuesta de este servidor.
#
Order allow,deny
Allow from all
</Directory>
#
# UserDir: El nombre del directorio que se adjunta a un directorio
# de usuario si se recibe una solicitud ~user.
#
# El path para el final del directorioto 'public_html' de la cuenta de usuario
# debe ser accesible para el servidor web userid. Normalmente esto significa que
# ~userid debe tener permisos 711, ~userid/public_html debe tener permisos 755,
```

```
# y los documentos que contiene deben ser de lectura para todos.
# De otro modo, el cliente sólo recivirá el mensaje "403 Forbidden".
#
# Ver también: http://httpd.apache.org/docs/misc/FAQ.html#forbidden
#
<IfModule mod_userdir.c>
#
# UserDir está inhabilitado por defecto ya que puede confirmar la presencia
# en el sistema de un username (dependiendo de los permisos del
# directorio home).
#
UserDir disable
#
# Para habilitar peticiones de /~user/ al directorio de usuario public_html,
# quite la opción "UserDir disable" de la línea de arriba y, en su lugar,
# descomente la línea siguiente:
#
#UserDir public_html
</IfModule>
#
# Controla el acceso a los directorios UserDir. Lo siguiente es un ejemplo
# de un sitio donde estos directorios están limitados a sólo lectura.
#
#<Directory /home/*/public_html>
# AllowOverride FileInfo AuthConfig Limit
# Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
# <Limit GET POST OPTIONS>
# Order allow,deny
# Allow from all
# </Limit>
# <LimitExcept GET POST OPTIONS>
# Order deny,allow
# Deny from all
# </LimitExcept>
#</Directory>
#
# DirectoryIndex: fija el archivo que Apache servirá si hay una petición
# de un directorio.
#
# El archivo index.html.var (a type-map) se usa para distribuir documentos negociados
# sobre la base del contenido. MultiViews Option puede usarse con el mismo
# propósito, pero es mucho más lento.
#
DirectoryIndex index.html index.html.var
#
# AccessFileName: El nombre del archivo en el que buscar cada directorio la
# información del control de acceso. Ver también la directiva AllowOverride.
#
AccessFileName .htaccess
#
# Las siguientes líneas impiden que los archivos .htaccess y .htpasswd
# sean vistos por los clientes web.
#
<Files ~ "^\.ht">
Order allow,deny
Deny from all
</Files>
#
```

```
# TypesConfig describe dónde se va a encontrar el archivo mime.types (o equivalente).
#
TypesConfig /etc/mime.types
#
# DefaultType es el tipo MIME predeterminado que usará el servidor en un
# documento si no puede determinar uno, como a partir de las extensiones de
# nombre de archivo. Si su servidor contiene sobre todo texto o documentos
# HTML, "text/plain" será un buen valor. Si la mayor parte de su contenido es
# binario, como aplicaciones o imágenes, es posible que quiera usar
# "application/octet-stream" en vez de evitar que los navegadores traten de
# mostrar archivos binarios como si se tratara de texto.
#
DefaultType text/plain
#
# El módulo mod_mime_magic permite al servidor usar las distintas pistas del
# contenido del propio archivo para determinar su tipo. La directiva MIMEMagicFile
# le indica al módulo dónde se encuentran las definiciones de las pistas.
#
<IfModule mod_mime_magic.c>
# MIMEMagicFile /usr/share/magic.mime
MIMEMagicFile conf/magic
</IfModule>
#
# HostnameLookups: Registra los nombres de los clientes o sus direcciones IP
# ejemplo, www.apache.org (on) o 204.62.129.132 (off).
# El valor predeterminado es off, ya que es mejor para la red si la gente tuviera
# que activar esta opción, ya que activarla significa que la solicitud de cada
# cliente request acabará siendo una solicitud de búsqueda AT LEAST para
# nameserver.
#
HostnameLookups Off
#
# EnableMMAP: Controla si el mapeo de memoria se usa para distribuir
# archivos (asumiendo que el SO subyacente lo soporta).
# Por defecto está en on; cámbielo a off si sirve desde un sistema de
# archivos NFS. En algunos sistemas, poniéndolo en off (a pesar del sistema
# de archivos) puede mejorar el rendimiento; para más detalles vea
# http://httpd.apache.org/docs-2.0/mod/core.html#enablemmap
#
#EnableMMAP off
#
# EnableSendfile: Controla si el soporte para envío de archivos del kernel
# se usa para distribuir archivos (asumiendo que el SO lo soporta).
# Por defecto está en on; cámbielo a off si sirve from desde un sistema de
# archivos NFS. Por favor, mire en
# http://httpd.apache.org/docs-2.0/mod/core.html#enablesendfile
#
#EnableSendfile off
#
# ErrorLog: La ubicación del archivo de registro de errores.
# Si no especifica una directiva ErrorLog en un contenedor <VirtualHost>,
# los mensajes de error que se relacionen con ese host virtual quedarán
# registrados aquí. Si define un error logfile en un contenedor <VirtualHost>
# los errores de ese host quedarán registrados ahí y no aquí.
#
ErrorLog logs/error_log
#
# LogLevel: Controla el número de mensajes registrados en el error_log.
```

```
# Los valores posibles son: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn
#
# Las directivas siguientes definen algunos apodos de formato para ser
# empleados con una directiva CustomLog (véase más abajo).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
#
# La ubicación y el formato del archivo de registro de acceso
# (Common Logfile Format).
# Si no define archivos de registro de acceso en un contenedor <VirtualHost>
# quedarán registrados aquí. Al contrario, si define archivos de
# registro de acceso per-<VirtualHost>, las transacciones quedarán
# registradas ahí y *no* en este archivo.
#
# CustomLog logs/access_log common
CustomLog logs/access_log combined
#
# Si prefiere tener archivos de registro de agente y de referente, descomente
# las siguientes directivas.
#
#CustomLog logs/referer_log referer
#CustomLog logs/agent_log agent
#
# Si prefiere un sólo archivo de registro con acceso, información de
# agente y de referente (Formato de Archivo de registro Combinado) podrá
# usar la siguiente directiva.
#
#CustomLog logs/access_log combined
#
# Opcionalmente, añada una línea que contenga la versión del servidor y el
# nombre del host virtual a las páginas generadas por el servidor
# (documentos de error, listados de directorios FTP, salidas mod_status y
# mod_info etc., pero no documentos generados por CGI). Establezca a
# "EMail" para incluir también un mailto: vínculo con el ServerAdmin.
# Establecer a uno de: On | Off | EMail
#
ServerSignature On
#
# Aliases: Añada aquí tantos aliases como necesite (sin límite). El formato es
# Alias fakename realname
#
# Observe que si incluye una barra / al final en el nombre falso, el servidor
# le obligará a estar presente en el URL. Así "/icons" no tiene alias en este
# ejemplo, sólo "/icons/". Si fakename (nombre falso) termina en /, entonces
# el nombre real debe terminar también en /, y si fakename omite la barra (/),
# el nombre real también debe omitirla.
#
# Hemos incluido el alias /icons/ para los listados de directorios FancyIndexed.
# Si no usa FancyIndexing, debe descomentarlo.
#
Alias /icons/ "/var/www/icons/"
<Directory "/var/www/icons">
```

```
Options Indexes MultiViews
AllowOverride None
Order allow,deny
Allow from all
</Directory>
#
# Esto debería cambiarse a ServerRoot/manual/. El alias proporciona el manual,
# incluso si elige cambiar la situación de DocumentRoot. Debe comentar
# esto si no quiere la documentación.
#
AliasMatch ^/manual(?:/(?:de|en|fr|ja|ko|ru))?(/.*)?$ "/var/www/manual$1"
<Directory "/var/www/manual">
Options Indexes FollowSymLinks MultiViews
AllowOverride None
Order allow,deny
Allow from all
<Files *.html>
SetHandler type-map
</Files>
SetEnvIf Request_URI ^/manual/de/ prefer-language=de
SetEnvIf Request_URI ^/manual/en/ prefer-language=en
SetEnvIf Request_URI ^/manual/fr/ prefer-language=fr
SetEnvIf Request_URI ^/manual/ja/ prefer-language=ja
SetEnvIf Request_URI ^/manual/ko/ prefer-language=ko
SetEnvIf Request_URI ^/manual/ru/ prefer-language=ru
RedirectMatch 301 ^/manual(?:/(de|en|fr|ja|ko|ru)){2,}(/.*)?$ /manual/$1$2
</Directory>
<IfModule mod_dav_fs.c>
# Location of the WebDAV lock database.
DAVLockDB /var/lib/dav/lockdb
</IfModule>
#
# ScriptAlias: Controla qué directorios contienen scripts de servidor.
# Los ScriptAliases son esencialmente lo mismo que los Aliases, con la excepción
# de que los documentos del directorio realname se consideran aplicaciones y
# los ejecuta el servidor cuando se solicitan, en vez de como documentos que se
# envían al cliente. Las mismas reglas acerca de barras finales "/" se aplican
# tanto a las directivas ScriptAlias como a las Alias.
#
ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
#
# "/var/www/cgi-bin" debe cambiarse al directorio CGI con ScriptAliased,
# si lo tiene configurado.
#
<Directory "/var/www/cgi-bin">
AllowOverride None
Options None
Order allow,deny
Allow from all
</Directory>
#
# Redirect le permite indicar a los clientes sobre los documentos que solían
# existir en los espacios de nombres del servidor, pero ya no lo hacen.
# Esto permite indicarle a los clientes dónde buscar es documento reubicado.
# Ejemplo:
# Redirect permanent /foo http://www.example.com/bar
#
# Las directivas que controlan la muestra de listados de directorios generados
```

```
# por el servidor.
#
# IndexOptions: Controla la apariencia de los listados de directorios generados
# por el servidor.
#
IndexOptions FancyIndexing VersionSort NameWidth=*
#
# Las directivas AddIcon* le indican al servidor qué icoono mostrar en los
# distintos archivos o extensiones de nombre de archivo. Sólo se muestran en
# los directorios FancyIndexed.
#
AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip
AddIconByType (TXT,/icons/text.gif) text/*
AddIconByType (IMG,/icons/image2.gif) image/*
AddIconByType (SND,/icons/sound2.gif) audio/*
AddIconByType (VID,/icons/movie.gif) video/*
AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
AddIcon /icons/layout.gif .html .shtml .htm .pdf
AddIcon /icons/text.gif .txt
AddIcon /icons/c.gif .c
AddIcon /icons/p.gif .pl .py
AddIcon /icons/f.gif .for
AddIcon /icons/dvi.gif .dvi
AddIcon /icons/uuencoded.gif .uu
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
AddIcon /icons/tex.gif .tex
AddIcon /icons/bomb.gif core
AddIcon /icons/back.gif ..
AddIcon /icons/hand.right.gif README
AddIcon /icons/folder.gif ^^DIRECTORY^^
AddIcon /icons/blank.gif ^^BLANKICON^^
#
# DefaultIcon es el icono a mostrar para los archivos que no tienen
# establecido explícitamente un icono.
#
DefaultIcon /icons/unknown.gif
#
# AddDescription permite colocar una descripción breve tras un archivo en
# índices generados por el servidor. Sólo se muestran en directorios FancyIndexed
# Formato: AddDescription "description" filename
#
#AddDescription "GZIP compressed document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar archive" .tgz
#
# ReadmeName es el nombre del archivo README en el que por defecto va a buscar
# el servidor y adjuntar a los listados de directorios.
#
# HeaderName es el nombre de archivo que debe anteponerse a los
# índices de directorios.
ReadmeName README.html
HeaderName HEADER.html
#
```

```
# IndexIgnore es una serie de nombres de archivos que la indexación de
# directorios debe ignorar y no incluir en el listado. Se permiten los
# comodines de estilo shell.
#
IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t
#
# DefaultLanguage y AddLanguage permiten especificar el lenguaje de un
# documento. Puede usar la negociación de contenido para dar al navegador
# un archivo en un lenguaje que el usuario pueda entender.
#
# Especifica un lenguaje por defecto. Esto significa que todos los datos
# que salgan sin un tag de lenguaje específico (ver debajo) se marcarán
# con éste. Probablemente NO quiera habilitarlo a menos que
# esté seguro de que es correcto para todos los casos.
#
# * ¡Generalmente es mejor no marcar una página con un
# * lenguaje correcto que marcarla con un lenguaje erróneo!
#
# DefaultLanguage nl
#
# Nota 1: El sufijo no tiene por qué ser el mismo que la palabra clave
# del lenguaje --- aqullos con documentos en Polaco (cuyo código de lenguaje
# es pl) pueden querer usar "AddLanguage pl .po" para evitar la
# ambigüedad con el sufijo común para scripts perl.
#
# Note 2: El ejemplo de abajo muestra que en algunos casos los dos
# caracteres de la abreviatura 'Language' no son idénticos a los dos
# caracteres del código 'Country' para su país,
# E.g. 'Danmark/dk' versus 'Danish/da'.
#
# Nota 3: En el caso de 'ltz' incumplimos el RFC al usar tres caracteres
# de especificación. Hay un 'trabajo en curso' para fijar esto y obtener
# los datos de referencia para un rfc1766 limpio.
#
# Catalan (ca) - Croatian (hr) - Czech (cs) - Danish (da) - Dutch (nl)
# English (en) - Esperanto (eo) - Estonian (et) - French (fr) - German (de)
# Greek-Modern (el) - Hebrew (he) - Italian (it) - Japanese (ja)
# Korean (ko) - Luxembourgeois* (ltz) - Norwegian Nynorsk (nn)
# Norwegian (no) - Polish (pl) - Portugese (pt)
# Brazilian Portuguese (pt-BR) - Russian (ru) - Swedish (sv)
# Simplified Chinese (zh-CN) - Spanish (es) - Traditional Chinese (zh-TW)
#
AddLanguage ca .ca
AddLanguage cs .cz .cs
AddLanguage da .dk
AddLanguage de .de
AddLanguage el .el
AddLanguage en .en
AddLanguage eo .eo
AddLanguage es .es
AddLanguage et .et
AddLanguage fr .fr
AddLanguage he .he
AddLanguage hr .hr
AddLanguage it .it
AddLanguage ja .ja
AddLanguage ko .ko
AddLanguage ltz .ltz
```

```
   AddLanguage nl .nl
   AddLanguage nn .nn
   AddLanguage no .no
   AddLanguage pl .po
   AddLanguage pt .pt
   AddLanguage pt-BR .pt-br
   AddLanguage ru .ru
   AddLanguage sv .sv
   AddLanguage zh-CN .zh-cn
   AddLanguage zh-TW .zh-tw
   #
   # LanguagePriority permite dar prioridad a algunos lenguajes
   # en caso de empate durante la negociación de contenido.
   #
   # Enumere los lenguajes en orden de prioridad decreciente. Aquí
   # están más o menos ordenados alfabéticamente. Probablemente quiera cambiarlo.
   #
   LanguagePriority en ca cs da de el eo es et fr he hr it ja ko ltz nl nn no pl pt pt-BR ru
sv zh-CN zh-TW
   #
   # ForceLanguagePriority le permite servir una página de resultados en lugar de
   # MULTIPLE CHOICES (Prefer) [en caso de un vínculo] or NOT ACCEPTABLE (Fallback)
   # [en caso de lenguajes no aceptados que coinciden con las variantes disponibles]
   #
   ForceLanguagePriority Prefer Fallback
   #
   # Especifica un conjunto de caracteres por defecto para todas las páginas de
   # salida. Esto siempre es una buena idea y abre la puerta a futuras
   # internalizaciones de su web, debería elegirlo siempre. Especificarlo por
   # defecto hace poco daño; como el estándar impone que una página esté en
   # iso-8859-1 (latin1) a menos que especifique otra cosa está simplemente
   # declarando lo obvio. Hay también algunas razones de seguridad
   # en los buscadores, relativas a javascript y análisis de URL
   # que le animan a usar siempre un conjunto de caracteres por defecto.
   #
   AddDefaultCharset UTF-8
   #
   # Se suelen usar extensiones del archivo de nombres para el conjunto de caracteres.
   # Probablemente quiera evitar colisiones con las extensiones de lenguaje, a menos
   # que sea bueno testeando cuidadosamente su estructura después de cada cambio.
   # Vea http://www.iana.org/assignments/character-sets para
   # la lista oficial de nombres de conjuntos de caracteres y sus respectivos RFCs
   #
   AddCharset ISO-8859-1 .iso8859-1 .latin1
   AddCharset ISO-8859-2 .iso8859-2 .latin2 .cen
   AddCharset ISO-8859-3 .iso8859-3 .latin3
   AddCharset ISO-8859-4 .iso8859-4 .latin4
   AddCharset ISO-8859-5 .iso8859-5 .latin5 .cyr .iso-ru
   AddCharset ISO-8859-6 .iso8859-6 .latin6 .arb
   AddCharset ISO-8859-7 .iso8859-7 .latin7 .grk
   AddCharset ISO-8859-8 .iso8859-8 .latin8 .heb
   AddCharset ISO-8859-9 .iso8859-9 .latin9 .trk
   AddCharset ISO-2022-JP .iso2022-jp .jis
   AddCharset ISO-2022-KR .iso2022-kr .kis
   AddCharset ISO-2022-CN .iso2022-cn .cis
   AddCharset Big5 .Big5 .big5
   # Para el ruso, se usa más de un conjunto de caracteres (dependiendo del cliente):
   AddCharset WINDOWS-1251 .cp-1251 .win-1251
```

```
AddCharset CP866 .cp866
AddCharset KOI8-r .koi8-r .koi8-ru
AddCharset KOI8-ru .koi8-uk .ua
AddCharset ISO-10646-UCS-2 .ucs2
AddCharset ISO-10646-UCS-4 .ucs4
AddCharset UTF-8 .utf8
# El conjunto de abajo no mapea para un estándar específico (iso)
# pero trabaja sobre un amplio rango de buscadores. Observe que el
# escribir cin mayúsculas realmente importa (no debería, pero es así
# en algunos buscadores).
#
# Vea http://www.iana.org/assignments/character-sets
# para una lista. Pero los buscadores soportan pocos.
#
AddCharset GB2312 .gb2312 .gb
AddCharset utf-7 .utf7
AddCharset utf-8 .utf8
AddCharset big5 .big5 .b5
AddCharset EUC-TW .euc-tw
AddCharset EUC-JP .euc-jp
AddCharset EUC-KR .euc-kr
AddCharset shift_jis .sjis
#
# AddType permite añadir o retocar la configuración MIME de archivos
# mime.types para tipos de archivos específicos.
#
AddType application/x-tar .tgz
#
# AddEncoding permite hacer que ciertos navegadores (Mosaic/X 2.1+) descompriman
# la información sobre la marcha. Nota: No todos los navegadores lo soportan.
# A pesar de la similitud de nombres, las directivas Add* siguientes no tienen
# nada que ver con las directivas de personalización FancyIndexing de arriba.
#
#AddEncoding x-compress Z
#AddEncoding x-gzip gz tgz
# Si las directivas AddEncoding de arriba están descomentadas, probablemente
# deberá definir esas extensiones para indicar tipos multimedia:
#
AddType application/x-compress .Z
AddType application/x-gzip .gz .tgz
#
# AddHandler permite asignar determinadas extensiones de archivo a "manipuladores":
# acciones no relacionadas con el tipo de archivo. Pueden estar construidas en
# el servidor o añadidas con el comando Action (ver debajo)
#
# Para usar scripts CGI fuera de los directorios ScriptAliased:
# (También necesitará añadir "ExecCGI" a la directiva "Options".)
#
#AddHandler cgi-script .cgi
#
# Para archivos que incluyen sus propios encabezados HTTP:
#
#AddHandler send-as-is asis
#
# Para archivos imagemap analizados sintácticamente por el servidor:
#
AddHandler imap-file map
#
```

```
# Para activar asignaciones de tipos (recursos negociados):
# (Está habilitado por defecto para permitir que la página de Apache "It Worked"
# sea distribuida en múltiples lenguajes.)
#
AddHandler type-map var
# Filters le permite procesar el contenido antes de enviarlo al cliente.
#
# Para analizar archivos .shtml para includes del lado del servidor (SSI):
# (Necesitará añadir también "Includes" a la directiva "Options".)
#
AddType text/html .shtml
AddOutputFilter INCLUDES .shtml
#
# Action le permite definir tipos multimedia que ejecutan un script siempre
# que se llame a un archivo coincidente. Con esto se evitan los nombres de rutas
# URL repetidos en procesadores de archivos CGI muy usados.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-script/location
#
#
# Las respuestas de error personalizables se presentan en tres gamas:
# 1) texto plano 2) redireccionamientos locales 3) redireccionamientos externos
#
# Algunos ejemplos:
#ErrorDocument 500 "El servidor falló."
#ErrorDocument 404 /missing.html
#ErrorDocument 404 "/cgi-bin/missing_handler.pl"
#ErrorDocument 402 http://www.example.com/subscription_info.html
#
#
# Poniendo todo esto junto, podemos Internacionalizar respuestas de error.
#
# Usamos Alias para redireccionar cualquier respuesta /error/HTTP_<error>.html.var
# a nuestro conjunto multi-lenguaje de mensajes por-error. Usamos
# inclusiones para sustituir el texto apropiado.
#
# Puede modificar la apariencia de los mensajes sin cambiar ninguno de
# lo archivos por defecto HTTP_<error>.html.var añadiendo la línea
#
# Alias /error/include/ "/your/include/path/"
#
# que le permite crear su propio conjunto de archivos comenzando con
# los archivos /var/www/error/include/ y
# copiándolos a /your/include/path/, incluso sobre la base de per-VirtualHost.
#
Alias /error/ "/var/www/error/"
<IfModule mod_negotiation.c>
<IfModule mod_include.c>
<Directory "/var/www/error">
AllowOverride None
Options IncludesNoExec
AddOutputFilter Includes html
AddHandler type-map var
Order allow,deny
Allow from all
LanguagePriority en es de fr
ForceLanguagePriority Prefer Fallback
</Directory>
```

```
# ErrorDocument 400 /error/HTTP_BAD_REQUEST.html.var
# ErrorDocument 401 /error/HTTP_UNAUTHORIZED.html.var
# ErrorDocument 403 /error/HTTP_FORBIDDEN.html.var
# ErrorDocument 404 /error/HTTP_NOT_FOUND.html.var
# ErrorDocument 405 /error/HTTP_METHOD_NOT_ALLOWED.html.var
# ErrorDocument 408 /error/HTTP_REQUEST_TIME_OUT.html.var
# ErrorDocument 410 /error/HTTP_GONE.html.var
# ErrorDocument 411 /error/HTTP_LENGTH_REQUIRED.html.var
# ErrorDocument 412 /error/HTTP_PRECONDITION_FAILED.html.var
# ErrorDocument 413 /error/HTTP_REQUEST_ENTITY_TOO_LARGE.html.var
# ErrorDocument 414 /error/HTTP_REQUEST_URI_TOO_LARGE.html.var
# ErrorDocument 415 /error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 500 /error/HTTP_INTERNAL_SERVER_ERROR.html.var
# ErrorDocument 501 /error/HTTP_NOT_IMPLEMENTED.html.var
# ErrorDocument 502 /error/HTTP_BAD_GATEWAY.html.var
# ErrorDocument 503 /error/HTTP_SERVICE_UNAVAILABLE.html.var
# ErrorDocument 506 /error/HTTP_VARIANT_ALSO_VARIES.html.var
</IfModule>
</IfModule>
#
# Las siguientes directivas modifican el comportamiento normal de la respuesta
# HTTP para manejar problemas conocidos con implementaciones del buscador.
#
BrowserMatch "Mozilla/2" nokeepalive
BrowserMatch "MSIE 4\.0b2;" nokeepalive downgrade-1.0 force-response-1.0
BrowserMatch "RealPlayer 4\.0" force-response-1.0
BrowserMatch "Java/1\.0" force-response-1.0
BrowserMatch "JDK/1\.0" force-response-1.0
#
# La siguiente directiva desactiva redireccionamientos sobre peticiones no-GET
# para un directorio que no incluye la barra trasera. Esto arregla un
# problema con Microsoft WebFolders que no maneja adecuadamente los
# redireccionamientos para carpetas con métodos DAV.
#
BrowserMatch "Microsoft Data Access Internet Publishing Provider" redirect-carefully
BrowserMatch "^WebDrive" redirect-carefully
BrowserMatch "^WebDAVFS/1.[012]" redirect-carefully
BrowserMatch "^gnome-vfs" redirect-carefully
#
# Permitir los informes de estado del servidor, con el URL http://servername/server-status
# Cambie el ".example.com" para que se adapte a su dominio.
#
#<Location /server-status>
# SetHandler server-status
# Order deny,allow
# Deny from all
# Allow from .example.com
#</Location>
#
# Permitir informes de configuración del servidor, con el URL
# http://servername/server-info (requiere que mod_info.c esté cargado).
# Cambie el ".example.com" para que se adapte a su dominio.
#
#<Location /server-info>
# SetHandler server-info
# Order deny,allow
# Deny from all
# Allow from .example.com
```

```
#</Location>
#
# Directivas del servidor proxy. Descomente las lineas siguientes
# para activar el servidor proxy:
#
#<IfModule mod_proxy.c>
#ProxyRequests On
#
#<Proxy *>
# Order deny,allow
# Deny from all
# Allow from .example.com
#</Proxy>
#
# Activar/desactivar el manejo de los encabezados HTTP/1.1 "Via:".
# ("Full" añade la versión del servidor; "Block" quita todos los encabezados
# Via: salientes) Set to one of: Off | On | Full | Block
#
#ProxyVia On
#
# Para activar también la caché, edite y descomente las líneas siguientes:
# Vea http://httpd.apache.org/docs-2.0/mod/mod_cache.html para más detalles.
#
#<IfModule mod_disk_cache.c>
# CacheEnable disk /
# CacheRoot "/var/cache/mod_proxy"
#</IfModule>
#
#</IfModule>
# End of proxy directives.
### Section 3: Virtual Hosts
#
# VirtualHost: Si desea mantener múltiples dominios/nombres de host en su
# máquina puede configurar contenedores VirtualHost. La mayoría de las
# configuraciones sólo usan hosts virtuales basados en nombres, así el servidor
# no tiene que preocuparse de las direcciones IP. Esto está indicado por los
# asteriscos en las directivas de abajo.
#
# Por favor, vea la documentación en
# <URL:http://httpd.apache.org/docs-2.0/vhosts/>
# para más detalles antes de intentar configurar hosts virtuales.
#
# Puede usar la opción de línea de comandos '-S' para verificar la
# configuración del host virtual.
#
# Para usar los hosts virtuales basados en nombres.
#
#NameVirtualHost *:80
#
# Ejemplo de VirtualHost:
# Casi todas las directivas Apache pueden entrar en un contenedor VirtualHost.
# La primera sección VirtualHost se usa para peticiones sin un
# servidor de nombres conocido.
#
#<VirtualHost *:80>
# ServerAdmin webmaster@dummy-host.example.com
# DocumentRoot /www/docs/dummy-host.example.com
# ServerName dummy-host.example.com
```

```
# ErrorLog logs/dummy-host.example.com-error_log
# CustomLog logs/dummy-host.example.com-access_log common
#</VirtualHost>
```

Apéndice C

Licencia de Documentación Libre GNU (traducción)

C.1. GFDL

Versión 1.1, Marzo de 2000

Esta es la GNU Free Document License (GFDL), versión 1.1 (de marzo de 2.000), que cubre manuales y documentación para el software de la Free Software Foundation, con posibilidades en otros campos. La traducción[1] no tiene ningún valor legal, ni ha sido comprobada de acuerdo a la legislación de ningún país en particular. Vea el original `http://www.gnu.org/copyleft/fdl.html`

Los autores de esta traducción son:

- Igor Támara `mailto:ikks@bigfoot.com`
- Pablo Reyes `mailto:reyes_pablo@hotmail.com`
- Revisión : Vladimir Támara P. `mailto:vtamara@gnu.org`

0. Preámbulo

El propósito de esta licencia es permitir que un manual, libro de texto, u otro documento escrito sea "libre" en el sentido de libertad: asegurar a todo el mundo la libertad efectiva de copiarlo y redistribuirlo, con o sin modificaciones, de manera comercial o no. En segundo término, esta licencia preserva para el autor o para quien publica una manera de obtener reconocimiento por su trabajo, al tiempo que no se consideran responsables de las modificaciones realizadas por terceros.

Esta licencia es una especie de "copyleft" que significa que los trabajos derivados del documento deben a su vez ser libres en el mismo sentido. Esto complementa la Licencia Pública General GNU, que es una licencia de copyleft diseñada para el software libre.

Hemos diseñado esta Licencia para usarla en manuales de software libre, ya que el software libre necesita documentación libre: Un programa libre debe venir con los manuales que ofrezcan la mismas libertades que da el software. Pero esta licencia no se limita a manuales de software; puede ser usada para cualquier trabajo textual, sin tener en cuenta su temática o si se publica como libro

[1] N. del T. Derechos Reservados en el sentido de GNU `http://www.gnu.org/copyleft/copyleft.es.html`

impreso. Recomendamos esta licencia principalmente para trabajos cuyo fin sea instructivo o de referencia.

1. Aplicabilidad y definiciones

Esta Licencia se aplica a cualquier manual u otro documento que contenga una nota del propietario de los derechos que indique que puede ser distribuido bajo los términos de la Licencia. El "Documento", en adelante, se refiere a cualquiera de dichos manuales o trabajos. Cualquier miembro del público es un licenciatario, y será denominado como "Usted".

Una "Versión Modificada" del Documento significa cualquier trabajo que contenga el Documento o una porción del mismo, ya sea una copia literal o con modificaciones y/o traducciones a otro idioma.

Una "Sección Secundaria" es un apéndice titulado o una sección preliminar al prólogo del Documento que tiene que ver exclusivamente con la relación de quien publica o, los autores del Documento o, el tema general del Documento (o asuntos relacionados) y cuyo contenido no entra directamente en este tema general. (Por ejemplo, si el Documento es en parte un texto de matemáticas, una Sección Secundaria puede no explicar matemáticas.) La relación puede ser un asunto de conexión histórica, o de posición legal, comercial, filosófica, ética o política con el tema o la materia del texto.

Las "Secciones Invariantes" son ciertas Secciones Secundarias cuyos títulos son denominados como Secciones Invariantes, en la nota que indica que el documento es liberado bajo esta licencia.

Los "Textos de Cubierta" son ciertos pasajes cortos de texto que se listan, como Textos de Portada o Textos de Contra Portada, en la nota que indica que el documento es liberado bajo esta Licencia.

Una copia "Transparente" del Documento, significa una copia para lectura en máquina, representada en un formato cuya especificación está disponible al público general, cuyos contenidos pueden ser vistos y editados directamente con editores de texto genéricos o (para imágenees compuestas por pixeles) de programas genéricos de dibujo o (para dibujos) algún editor gráfico ampliamente disponible, y que sea adecuado para exportar a formateadores de texto o para traducción automática a una variedad de formatos adecuados para ingresar a formateadores de texto. Una copia hecha en un formato de un archivo que no sea Transparente, cuyo formato ha sido diseñado para impedir o dificultar subsecuentes modificaciones posteriores por parte de los lectores no es Transparente. Una copia que no es "Transparente" es llamada "Opaca".

Como ejemplos de formatos adecuados para copias Transparentes están el ASCII plano sin formato, formato de Texinfo, formato de LaTeX, SGML o XML usando un DTD disponible ampliamente, y HTML simple que sigue los estándares, diseñado para modificaciones humanas. Los formatos Opacos incluyen PostScript, PDF, formatos propietarios que pueden ser leídos y editados unicamente en procesadores de palabras propietarios, SGML o XML para los cuáles los DTD y/o herramientas de procesamiento no están disponibles generalmente, y el HTML generado por máquinas producto de algún procesador de palabras solo para propósitos de salida.

La "Portada" en un libro impreso significa, la portada misma, más las páginas siguientes necesarias para mantener la legibilidad del material, que esta Licencia requiere que aparezca en la portada. Para trabajos en formatos que no tienen Portada como tal, "Portada" significa el texto cerca a la aparición más prominente del título del trabajo, precediendo el comienzo del cuerpo del trabajo.

2. Copia literal

Puede copiar y distribuir el Documento en cualquier medio, sea en forma comercial o no, siempre y cuando esta Licencia, las notas de derecho de autor, y la nota de licencia que indica que esta Licencia se aplica al Documento se reproduzca en todas las copias, y que usted no adicione ninguna otra condición a las expuestas en en esta Licencia. No puede usar medidas técnicas para

obstruir o controlar la lectura o copia posterior de las copias que usted haga o distribuya. Sin embargo, usted puede aceptar compensación a cambio de las copias. Si distribuye un número suficientemente grande de copias también deberá seguir las condiciones de la sección 3.

También puede prestar copias, bajo las mismas condiciones establecidas anteriormente, y puede exhibir copias publicamente.

3. Copiado en cantidades

Si publica copias impresas del Documento que sobrepasen las 100, y la nota de Licencia del Documento exige Textos de Cubierta, debe incluir las copias con cubiertas que lleven en forma clara y legible, todos esos textos de Cubierta: Textos Frontales en la cubierta frontal, y Textos Posteriores de Cubierta en la Cubierta Posterior. Ambas cubiertas deben identificarlo a Usted clara y legiblemente como quien publica tales copias. La Cubierta Frontal debe mostrar el título completo con todas las palabras igualmente prominentes y visibles. Además puede adicionar otro material en la cubierta. Las copias con cambios limitados en las cubiertas, siempre que preserven el título del Documento y satisfagan estas condiciones, puede considerarse como copia literal.

Si los textos requeridos para la cubierta son muy voluminosos para que ajusten legiblemente, debe colocar los primeros (tantos como sea razonable colocar) en la cubierta real, y continuar el resto en páginas adyacentes.

Si publica o distribuye copias Opacas del Documento cuya cantidad exceda las 100, debe incluir una copia Transparente que pueda ser leída por una máquina con cada copia Opaca, o entregar en o con cada copia Opaca una dirección en red de computador publicamente-accesible conteniendo una copia completa Transparente del Documento, sin material adicional, a la cual el público en general de la red pueda acceder a bajar anónimamente sin cargo usando protocolos de standard público. Si usted hace uso de la última opción, deberá tomar medidas necesarias, cuando comience la distribución de las copias Opacas en cantidad, para asegurar que esta copia Transparente permanecerá accesible en el sitio por lo menos un año después de su última distribución de copias Opacas (directamente o a través de sus agentes o distribuidores) de esa edición al público.

Se solicita, aunque no es requisito, que contacte a los autores del Documento antes de redistribuir cualquier gran número de copias, para permitirle la oportunidad de que le provean una versión del Documento.

4. Modificaciones

Puede copiar y distribuir una Versión Modificada del Documento bajo las condiciones de las secciones 2 y 3 anteriores, siempre que usted libere la Versión Modificada bajo esta misma Licencia, con la Versión Modificada haciendo el rol del Documento, por lo tanto licenciando la distribución y modificación de la Versión Modificada a quienquiera que posea una copia de este. En adición, debe hacer lo siguiente en la Versión Modificada:

- A. Uso en la Portada (y en las cubiertas, si hay alguna) de un título distinto al del Documento, y de versiones anteriores (que deberían, si hay alguna, estar listados en la sección de Historia del Documento). Puede usar el mismo título que versiones anteriores al original siempre que quién publicó la primera versión lo permita.

- B. Listar en la Portada, como autores, una o más personas o entidades responsables por la autoría o las modificaciones en la Versión Modificada, junto con por lo menos cinco de los autores principales del Documento (Todos sus autores principales, si hay menos de cinco).

- C. Estado en la Portada del nombre de quién publica la Versión Modificada, como quien publica.

- D. Preservar todas las notas de derechos de autor del Documento.

- E. Adicionar una nota de derecho de autor apropiada a sus modificaciones adyacentes a las otras notas de derecho de autor.
- F. Incluir, immediatamente después de la nota de derecho de autor, una nota de licencia dando el permiso público para usar la Versión Modificada bajo los términos de esta Licencia, de la forma mostrada en la Adición (LEGAL)abajo.
- G. Preservar en esa nota de licencia el listado completo de Secciones Invariantes y en los Textos de las Cubiertas que sean requeridos como se especifique en la nota de Licencia del Documento
- H. Incluir una copia sin modificación de esta Licencia.
- I. Preservar la sección llamada "Historia", y su título, y adicionar a esta una sección estableciendo al menos el título, el año,los nuevos autores, y quién publicó la Versión Modificada como reza en la Portada. Si no hay una sección titulada "Historia" en el Documento, crear una estableciendo el título, el año, los autores y quien publicó el Documento como reza en la Portada, añadiendo además un artículo describiendo la Versión Modificada como se estableció en el punto anterior.
- J. Preservar la localización en red, si hay , dada en la Documentación para acceder públicamente a una copia Transparente del Documento, tanto como las otras direcciones de red dadas en el Documento para versiones anteriores en las cuáles estuviese basado. Estas pueden ubicarse en la sección "Historia". Se puede omitir la ubicación en red para un trabajo que sea publicado por lo menos 4 años antes que el mismo Documento, o si quien publica originalmente la versión da permiso explícitamente.
- K. En cualquier sección titulada "Agradecimientos" o "Dedicatorias", preservar el título de la sección, y preservar en la sección toda la sustancia y el tono de los agradeimientos y/o dedicatorias de cada contribuyente que estén incluídas.
- L. Preservar todas las Secciones Invariantes del Documento, sin alterar su texto ni sus títulos. Números de sección o el equivalente no son considerados parte de los títulos de la sección. M. Borrar cualquier sección titulada "Aprobaciones". Tales secciones no pueden estar incluidas en las Versiones Modificadas.
- M. Borrar cualquier sección titulada "Aprobaciones". Tales secciones no pueden estar incluidas en las Versiones Modificadas.
- N. No retitular ninguna sección existente como "Aprobaciones" o conflictuar con título con alguna Sección Invariante.

Si la Versión Modificada incluye secciones o apendices nuevos o preliminares al prólogo que califican como Secciones Secundarias y contienen material no copiado del Documento, puede opcionalmente designar algunas o todas esas secciones como invariantes. Para hacerlo, adicione sus títulos a la lista de Secciones Invariantes en la nota de licencia de la Versión Modificada. Tales títulos deben ser distintos de cualquier otro título de sección.

Puede adicionar una sección titulada "Aprobaciones", siempre que contenga unicamente aprobaciones de su Versión Modificada por varias fuentes–por ejemplo, observaciones de peritos o que el texto ha sido aprobado por una organización como un standard.

Puede adicionar un pasaje de hasta cinco palabras como un Texto de Cubierta Frontal, y un pasaje de hasta 25 palabras como un texto de Cubierta Posterior, al final de la lista de Textos de Cubierta en la Versión Modificada. Solamente un pasaje de Texto de Cubierta Frontal y un Texto de Cubierta Posterior puede ser adicionado por (o a manera de arreglos hechos por) una entidad. Si el Documento ya incluye un texto de cubierta para la misma cubierta, previamente adicionado por usted o por arreglo hecho por la misma entidad, a nombre de la cual está actuando,

no puede adicionar otra; pero puede reemplazar la anterior, con permiso explícito de quien publicó anteriormente tal cubierta.

El(los) autor(es) y quien(es) publica(n) el Documento no dan con esta Licencia permiso para usar sus nombres para publicidad o para asegurar o implicar aprobación de cualquier Versión Modificada.

5. Combinando documentos

Puede combinar el Documento con otros documentos liberados bajo esta Licencia, bajo los términos definidos en la sección 4 anterior para versiones modificadas, siempre que incluya en la combinación todas las Secciones Invariantes de todos los documentos originales, sin modificar, y listadas todas como Secciones Invariantes del trabajo combinado en su nota de licencia.

El trabajo combinado necesita contener solamente una copia de esta Licencia, y múltiples Seccions Invariantes Idénticas pueden ser reemplazadas por una sola copia. Si hay múltiples Secciones Invariantes con el mismo nombre pero con contenidos diferentes, haga el título de cada una de estas secciones único adicionándole al final de este, en paréntesis, el nombre del autor o de quien publicó originalmente esa sección, si es conocido, o si no, un número único. Haga el mismo ajuste a los títulos de sección en la lista de Secciones Invariantes en la nota de licencia del trabajo combinado.

En la combinación, debe combinar cualquier sección titulada "Historia" de los varios documentos originales, formando una sección titulada "Historia"; de la misma forma combine cualquier seción titulada "Agradecimientos", y cualquier sección titulada "Dedicatorias". Debe borrar todas las secciones tituladas "Aprobaciones."

6. Colecciones de documentos

Puede hacer una colección consistente del Documento y otros documentos liberados bajo esta Licencia, y reemplazar las copias individuales de esta Licencia en los varios documentos con una sola copia que esté incluida en la colección, siempre que siga las reglas de esta Licencia para una copia literal de cada uno de los documentos en cualquiera de todos los aspectos.

Puede extraer un solo documento de una de tales colecciones, y distribuirlo individualmente bajo esta Licencia, siempre que inserte una copia de esta Licencia en el documento extraido, y siga esta Licencia en todos los otros aspectos concernientes a la copia literal de tal documento.

7. Agregación con trabajos independientes

Una recopilación del Documento o de sus derivados con otros documentos o trabajos separados o independientes, en cualquier tipo de distribución o medio de almacenamiento, no como un todo, cuenta como una Versión Modificada del Documento, teniendo en cuenta que ninguna compilación de derechos de autor sea clamada por la recopilación. Tal recopilación es llamada un "agregado", y esta Licencia no aplica a los otros trabajos auto-contenidos y por lo tanto compilados con el Documento, o a cuenta de haber sido compilados, si no son ellos mismos trabajos derivados del Documento.

Si el requerimiento de la sección 3 del Texto de la Cubierta es aplicable a estas copias del Documento, entonces si el Documento es menor que un cuarto del agregado entero, Los Textos de la Cubierta del Documento pueden ser colocados en cubiertas que enmarquen solamente el Documento entre el agregado. De otra forma deben aparecer en cubiertas enmarcando todo el agregado.

8. Traducción

La Traducción es considerada como una clase de modificación, Así que puede distribuir traducciones del Documento bajo los términos de la sección 4. Reemplazar las Secciones Invariantes con traducciones requiere permiso especial de los dueños de derecho de autor, pero puede incluir traducciones de algunas o todas las Secciones Invariantes adicionalmente a las versiones originales de las Secciones Invariantes. Puede incluir una traducción de esta Licencia siempre que incluya también la versión Inglesa de esta Licencia. En caso de un desacuerdo entre la traducción y la versión original en Inglés de esta Licencia, la versión original en Inglés prevalecerá.

9. Terminación

No se puede copiar, modificar, sublicenciar, o distribuir el Documento excepto por lo permitido expresamente bajo esta Licencia. Cualquier otro intento de copia, modificación, sublicenciamiento o distribución del Documento es nulo, y serán automáticamente terminados sus derechos bajo esa licencia. De todas maneras, los terceros que hayan recibido copias, o derechos, de su parte bajo esta Licencia no tendrán por terminadas sus licencias siempre que tales personas o entidades se encuentren en total conformidad con la licencia original.

10 Futuras revisiones de esta licencia

La Free Software Foundation puede publicar nuevas, revisadas versiones de la Licencia de Documentación Libre GNU de tiempo en tiempo. Tales nuevas versiones serán similares en espíritu a la presente versión, pero pueden diferir en detalles para solucionar problemas o intereses. Vea `http://www.gnu.org/copyleft/`.

Cada versión de la Licencia tiene un número de versión que la distingue. Si el Documento especifica que una versión numerada particularmente de esta licencia o "cualquier versión posterior" se aplica a esta, tiene la opción de seguir los términos y condiciones de la versión especificada o cualquiera posterior que ha sido publicada(no como un borrador)por la Free Software Foundation. Si el Documento no especifica un número de versión de esta Licencia, puede escoger cualquier versión que haya sido publicada(no como un borrador) por la Free Software Foundation.