



Treball fi de carrera

**ENGINYERIA TÈCNICA EN
INFORMÀTICA DE SISTEMES**

**Facultat de Matemàtiques
Universitat de Barcelona**

**INSTALACIÓN (SEMI)AUTOMÁTICA
DE NODOS EN UN CLUSTER**

Francesc Sebastià Martínez

Director: Jaume Timoneda Salat
Realitzat a: Departament de Matemàtica
Aplicada i Anàlisi. UB
Barcelona, 15 de setembre de 2005

A mis padres,
por la educación que me han proporcionado

Tabla de contenidos

Resumen.....	7
1. Introducción.....	1
1.1. Motivación.....	1
1.2. Objetivos.....	2
1.3. Agradecimientos.....	2
2. Conceptos previos.....	3
2.1. Introducción.....	3
2.2. Descripción y conceptos.....	3
2.3. Funcionamiento general.....	4
2.4. Características.....	5
2.5. Recomendaciones.....	7
2.5.1. Lea las páginas del manual (man pages).....	7
2.5.2. Guarde copias de los ficheros.....	7
2.5.3. Comente todos los cambios.....	8
2.5.4. Ficheros centralizados.....	9
2.6. Convenciones utilizadas en este documento.....	9
3. Instalando Debian GNU/Linux.....	11
3.1. Introducción.....	11
3.2. La secuencia de instalación.....	11
3.3. Arrancando Debian GNU/Linux por primera vez.....	14
4. Configurando los servicios.....	17
4.1. Introducción.....	17
4.2. Servicios.....	17
4.2.1. xinetd.....	17
4.2.2. rsh (remote shell).....	18
4.2.3. ntpd (Network Time Protocol).....	19
4.2.4. rwhod (server status daemon).....	20
4.2.5. bind9 (Berkeley Internet Name Domain).....	20
4.3. Servicios para FAI.....	22
4.3.1. apt-proxy.....	22
4.3.2. tftp-hpa (Trivial File Transfer Protocol de HPA).....	25
4.3.3. NFS (Network File System).....	25
4.3.4. DHCP (Dynamic Host Configuration Protocol).....	26
4.4. Otros paquetes.....	27
5. Instalando FAI.....	29
5.1. Introducción.....	29

5.2. Instalando FAI.....	29
5.3. Configurando FAI.....	30
5.3.1. Configurando /etc/fai/sources.list.....	30
5.3.2. Configurando fai.conf (/etc/fai/fai.conf).....	30
5.3.3. Configurando make-fai-nfsroot.conf (/etc/fai/make-fai-nfsroot.conf).....	33
5.4. fai-setup (creando el nfsroot).....	35
5.5. Reconfigurando el servidor DHCP.....	39
6. Preparando el arranque.....	41
6.1. Introducción.....	41
6.2. Recopilando las direcciones MAC (direcciones físicas).....	41
7. Secuencia de instalación.....	45
7.1. Introducción.....	45
7.2. La secuencia de instalación.....	45
7.3. Monitorizando la instalación.....	46
7.4. Configurando FAI (set up).....	47
7.5. Definiendo clases, variables y cargando los módulos del núcleo.....	47
7.6. Particionando los discos duros y creando los sistemas de ficheros.....	48
7.7. Instalando paquetes.....	49
7.8. Configuración específica del host.....	49
7.9. Grabando los archivos de registro (logs).....	50
7.10. Reiniciando el nuevo sistema.....	50
8. Configurando los clientes.....	51
8.1. Introducción.....	51
8.2. El espacio de configuraciones.....	52
8.3. Configuración	53
8.3.1. Reconfigurando DHCP y /etc/hosts.....	54
8.3.2. Definiendo clases.....	54
8.3.3. Definiendo variables.....	55
8.3.4. Configuración de los discos duros locales.....	57
8.3.5. Configuración de los paquetes.....	58
8.3.6. Configuración de sus propios paquetes.....	60
8.3.7. Scripts de personalización.....	61
8.3.8. Cambiando el dispositivo de arranque.....	64
8.3.9. Hooks (ganchos).....	64
8.3.10. Buscando errores.....	64
9. Conclusiones.....	67

A. Instalar un mirror Debian.....	69
B. Referencia fai.conf.....	71
C. Referencia make-fai-nfsroot.conf.....	75
D. Referencia de FAI_FLAGS.....	77
E. Las tareas por defecto.....	79
F. El concepto de clase.....	83
G. Referencia de comandos	85
Bibliografia.....	89

Índice de cuadros

Cuadro 1. Particionado.....	13
Cuadro 2. Variables de fai.conf.....	33
Cuadro 3. Variables de make-fai-nfsroot.conf.....	35
Cuadro 4. El espacio de configuraciones.....	53
Cuadro 5. Referencia de /etc/fai/fai.conf.....	73
Cuadro 6. Referencia de /etc/fai/make-fai-nfsroot.conf.....	76
Cuadro 7. Referencia de la variable FAI_FLAGS.....	77

Resumen

Este proyecto intenta llevar a la práctica, de la forma más automatizada y mejor posible, la instalación de nodos en un cluster. Un nodo es cada uno de los ordenadores que cooperan en un cluster. Un cluster es un grupo de múltiples ordenadores (nodos) unidos, mediante una red de alta velocidad, de tal forma que el conjunto es visto como un único ordenador más potente. Principalmente, su utilización es para el cálculo intensivo, en operaciones que un sólo ordenador tardaría muchísimo más tiempo.

El cluster en cuestión, es el utilizado por el departamento de MAiA (Matemàtica Aplicada i Anàlisi)¹ de la Universitat de Barcelona². Está basado en el Sistema Operativo GNU/Linux en su distribución Debian, y usa arquitectura x86, tanto procesadores Intel como AMD, lo que lo convierte en un cluster semi-homogéneo.³

Para llevar a cabo esta tarea se ha utilizado el paquete Debian FAI (Fully Automatic Installation), que ha sido de gran ayuda pero que es de difícil configuración, debido a todas las tecnologías de las que hace uso (y que son necesarias para las instalaciones).

En esta memoria se procuran dar las directrices básicas y apropiadas para la instalación de un nuevo nodo en el cluster. Intentando usar explicaciones simples, metódicas y ordenadas y evitando caer en tecnicismos, o en largas explicaciones sobre las tecnologías utilizadas, para que la instalación pueda llevarse a cabo, incluso, por un neófito no experto en Linux.

Una vez seguidas estas instrucciones y después de la correcta configuración, añadir un nodo al cluster será cuestión de minutos.

1 Departamento de MAiA de la UB - <http://www.maia.ub.es/>

2 Universitat de Barcelona (UB) - <http://www.ub.edu/>

3 Siguiendo definiciones de la wikipedia - http://es.wikipedia.org/wiki/Cluster_de_computadores.

1. Introducción

1.1. Motivación

Instalar una nueva máquina siempre ha sido una tarea fascinante para mí. Desde sacarla de su envoltorio a instalarle todas las aplicaciones de usuario, incluyendo el sistema operativo, no sin antes haber trasteado un poco mirando sus entrañas: la disposición de la placa, los conectores, los buses de expansión, la CPU, el número de ventiladores, los módulos de memoria, etc. Para luego comprobar, sorprendido, su rendimiento, viendo como aquellas antiguas tareas que se me hacían tan pesadas realizar en mi antiguo ordenador, por las esperas en la respuesta entre clic y clic, ahora el nuevo las maneja con soltura y rapidez, volviendo a lo “lógico”, que él me espere a mí y no yo a él (de sobras es sabido que los ordenadores tienen más paciencia que las personas). Y poniéndolo a prueba con algunos test y aplicaciones críticas, para regocijarme con las prestaciones que me ofrece mi nueva adquisición.

Si bien, en otros entornos, como en un cluster, donde las necesidades son poner en funcionamiento lo antes posible susodicha máquina y perder el menor tiempo posible en la instalación y configuración de la herramienta para seguir centrándonos en nuestro trabajo, se hace necesario un sistema más o menos automático que nos permite tales objetivos. Más aun, cuando queremos añadir varias máquinas y no queremos perder horas y horas haciendo las mismas tareas: particionar discos duros, instalar sistema operativo, agregar software necesario, configurar la red, etc. Además, la repetición de una tarea, a veces tan ardua y aburrida, puede llevar a errores. ¡Para las tareas repetitivas ya se inventaron los ordenadores!

Lo ideal sería que la instalación de un nuevo nodo fuese de forma automática y en el menor tiempo posible; lo que este proyecto lleva a cabo, si bien, siempre se de-

ben invertir pocos minutos para configurar el nombre y el interfaz de red del nuevo nodo, entre otras pequeñas menudencias.

1.2. Objetivos

Con este proyecto se pretende automatizar al máximo la instalación de un nuevo nodo en un cluster. Se pretende que se realice sin apagar en ningún momento el cluster, mientras éste se encuentra en producción y sin que la adición de este nodo interfiera en su normal funcionamiento. De forma que una vez instalado el nodo, éste se reinicie automáticamente y forme parte ya del cluster, sin ninguna configuración posterior, disponiendo de él para el trabajo productivo.

1.3. Agradecimientos

Deseo agradecer al director de mi proyecto, el profesor Jaume Timoneda del departamento de MAiA de la Universitat de Barcelona, su dedicación y las horas que ha perdido aconsejándome y dándome explicaciones sobre GNU/Linux, que han hecho de este proyecto una realidad. Sin su ayuda, probablemente aún estaría batallando con tecnologías que antes no conocía y que él me ha expuesto pacientemente de forma clara y concisa.

También deseo mostrar mis agradecimientos a Thomas Lange⁴ por el paquete Debian FAI⁵ (Fully Automatic Installation) y su documentación, así como a todos aquellos que han contribuido en su desarrollo.

4 Thomas Lange - <http://www.informatik.uni-koeln.de/~lange>

5 FAI (Fully Automatic Installation) - <http://www.informatik.uni-koeln.de/fai/>

2. Conceptos previos

2.1. Introducción

Antes de empezar con la instalación (semi)automática de nodos en un cluster se van a exponer ciertos conceptos necesarios para su realización. También se explicará como realizar una instalación básica de un sistema GNU/Debian que hará las veces de servidor de instalaciones para los nuevos nodos.

2.2. Descripción y conceptos

FAI es un software para la instalación no interactiva y desatendida de sistemas Debian GNU/Linux. Tanto puede usarse para instalar un nuevo PC, como para instalar un cluster entero de equipos, es un método totalmente escalable.

Los scripts que usa FAI están programados en shell, cfengine, expect y Perl⁶, lo cual lo convierten en un sistema altamente flexible, potente y adaptable. Está desarrollado para facilitar la tarea de instalación de sistemas Debian GNU/Linux a administradores de sistemas. Además, al ser un herramienta de instalación de propósito general, permite instalaciones de muy distinta índole: instalación de un cluster Beowulf, una granja de renderización, un laboratorio Linux o un aula con software de escritorio.

La utilización de distinto hardware o la necesidad de ajustar configuraciones para los distintos equipos no es problema alguno para FAI; como mecanismo para su ajuste se utiliza el concepto de clases (que incluso pueden estar definidas de forma dinámica).

Lista de conceptos y terminología necesaria:

6 Perl - www.perl.org

- **servidor de instalaciones (install server / servidor):** Es el host (equipo) donde el paquete FAI se instalará. Provee los servicios y datos necesarios para que los **clientes de instalación** puedan ejecutar la instalación automática.
- **clientes de instalación (install client / cliente):** Son aquellos hosts que serán instalados usando FAI.
- **Configuración:** Son los detalles de cómo se llevará a cabo la instalación de los clientes. Esta incluye información acerca de:
 - Particionado de discos
 - Sistemas de ficheros, puntos y opciones de montaje
 - Paquetes (software)
 - Disposición de teclado, zona de tiempo, NIS, configuración de las X, sistemas de ficheros remotos, cuentas de usuarios, etc.
- **nfsroot:** Es un árbol de ficheros (chroot) del servidor. Es todo el árbol de ficheros que utilizarán los **clientes** durante el proceso de instalación. Todos los clientes lo comparten mediante NFS⁷ y lo montan con permisos de sólo lectura.

2.3. Funcionamiento general

El funcionamiento a grandes rasgos de FAI es el siguiente: EL **cliente** carga desde un disco flexible o mediante red. Y solicita una IP al servidor, carga un núcleo (kernel) Linux y monta su sistema de ficheros raíz mediante NFS.

⁷ Más información acerca de NFS:
Wikipedia - <http://es.wikipedia.org/wiki/NFS> y en The Linux Documentation Project - <http://tldp.org/>

Después de la carga del sistema, FAI empieza a ejecutar scripts para la instalación automática del **cliente**, sin necesidad de ningún tipo de interacción (no es necesario que el **cliente** tenga ni disquetera, ni CDROM, ni monitor, ni teclado ni ratón).

Primero, los discos duros son particionados, los sistemas de ficheros creados y el software instalado (incluyendo el núcleo del **cliente**). Seguidamente, se configura el sistema operativo instalado según las necesidades locales, mediante algunos scripts.

Finalmente, el nuevo sistema será cargado desde el disco duro local.

Los detalles de cómo instalar el **cliente** (la configuración) están almacenados en el **espacio de configuraciones** (configuration space) del servidor. Los ficheros de configuración son compartidos entre distintos clientes, si son similares, usando el concepto de clase. Por tanto, no es necesario crear una configuración por cada nuevo equipo.

Como valor añadido, FAI puede usarse como un sistema de rescate de red (sin realizar ninguna instalación) permitiendo entrar remotamente en el sistema cliente y solucionar los problemas del host.

2.4. Características

- **Instalación muy rápida y completa**, totalmente automatizada. Depende de las configuraciones, cantidad de software instalado y capacidades del equipo. En todas nuestras pruebas, en el departamento de MAiA, siempre ha tardado **menos de 6 minutos** en realizar una instalación completa.
- **Actualización** de equipos en producción sin reinstalaciones.
- Los **clientes** pueden arrancar desde un disco flexible, tarjeta de red o CDROM.

- Están soportados los protocolos DHCP, BOOTP y PXE.
- Funciona en equipos con pocos recursos, 486, ya que no necesita un disco RAM inicial.
- Identificación remota (login) mediante `ssh`⁸ durante la instalación.
- Se dispone de dos terminales virtuales durante el proceso de instalación.
- Configuraciones similares compartidas, mediante clases.
- Los registros (logs) de la instalación son guardados en el servidor.
- Permite el uso de scripts shell, perl, expect y cfengine.
- Acceso al mirror Debian mediante NFS, FTP o HTTP.
- Disposición de teclado configurable.
- Puede usarse como sistema de rescate.
- Soporta clientes sin disco (diskless).
- Permite cambiar el comportamiento por defecto de FAI, usando ganchos (hooks).
- Soporta los gestores de arranque lilo y grub.
- Soporta los sistemas de ficheros ext3, ReiserFS y XFS.
- Detección automática del hardware.
- Dispone de clases Beowulf predefinidas y varios ejemplos.

⁸ SSH - <http://es.wikipedia.org/wiki/SSH>

2.5. Recomendaciones

2.5.1. Lea las páginas del manual (*man pages*)

En este manual, como ocurre en otros manuales, al lado de algunos comandos o ficheros puede que encuentre, entre paréntesis, un número. Este número indica en que sección del manual del sistema hay más información sobre el comando, de forma que si quiere consultarla, abra un terminal virtual en su ordenador y teclee:

```
# man [número] comando
```

Un ejemplo podría ser:

```
# man 8 fai-chboot
```

Estas páginas son ampliamente utilizadas por los usuarios y administradores de sistemas Unix/Linux, ya que incorporan información importante acerca de los comandos, las configuraciones y otras cuestiones relacionadas con el sistema. Sin lugar a dudas, es una fuente de información muy importante que le sacará de muchos apuros y le ayudará en el uso diario de su sistema. Para empezar con ellas, puede consultar:

```
# man man
```

Algunas de estas páginas se encuentran traducidas al castellano, otras sin embargo han de consultarse en inglés.

2.5.2. Guarde copias de los ficheros

Antes de realizar cualquier cambio en cualquier fichero crítico (configuración, script, etc.) es muy recomendable que haga una copia de seguridad. Personalmente, recomiendo la siguiente notación:

```
# cp nombre-fichero nombre-fichero.old.AAAAMMDD
```

donde *nombre-fichero* es el nombre del archivo donde se van a realizar los cambios y *AAAA* indica el año en número, *MM* indica el mes en número, *DD* indica el día en número. Si en un mismo día, realiza varios cambios en un archivo, añada como sufixo *-R* (guión, R), donde *R* es el número de revisión.

2.5.3. Comente todos los cambios

Cualquier cambio que realice dentro de un fichero, ha de comentarse y explicarse convenientemente. Normalmente los comentarios van precedidos del carácter almohadilla (#) . Indique también su nombre y la fecha en la que realizó el cambio, y no olvide explicar la razón de dicho cambio; puede que lo que hoy le parezca totalmente evidente en un futuro le parezca sumamente confuso o incluso un error.

Comentar bien todos los cambios le ahorrará muchos quebraderos de cabeza.

Posteriormente, cuando empiece a trabajar con FAI, será importante que tenga una copia de todos los ficheros que cambie en `nfsroot` y en el **espacio de configuraciones**. Ya que el **espacio de configuraciones** no tiene porque perderse, ya que FAI no lo sobrescribe en ningún momento, no debe preocuparse por él. Pero el `nfsroot` es borrado por FAI cada vez que ejecuta el comando `fai-setup` o `make-fai-nfsroot`, por tanto, le recomiendo que cree un directorio, por ejemplo, con el nombre `nfsroot_cambios`, y cada vez que cambie un fichero del `nfsroot` lo copie en el sitio indicado de `nfsroot_cambios`. Por ejemplo, si cambia el fichero `/etc/fai/fai.conf` dentro de su `nfsroot`, recuerde copiarlo con su ruta relativa dentro de `nfsroot_cambios`.

2.5.4. *Ficheros centralizados*

Después de ejecutar el comando **make-fai-nfsroot** podrá restaurar los cambios con:

```
# cp -a -R /ruta/nfsroot_cambios /ruta/nfsroot
```

Como recomendación final, ponga tanto el **nfsroot** como el **espacio de configuraciones**, como el **nfsroot_cambios**, dentro del mismo subdirectorio, para navegar rápidamente entre ellos y no tener que recordar las rutas menos homogéneas que FAI establece por defecto. Por ejemplo, nosotros en nuestras pruebas hemos utilizado el siguiente árbol de directorios para ello:

```
/home/fai/config           para el espacio de configuraciones
/home/fai/nfsroot         para el nfsroot
/home/fai/nfsroot_cambios para los cambios de nfsroot
```

2.6. *Convenciones utilizadas en este documento*

Los comandos se muestran en negrita y letra Courier de paso fijo, cuando son explicados dentro del texto.

Los nombres de ficheros y de directorios, el texto de consola y el contenido de los ficheros se muestra en Courier.

Cuando se omite un trozo de fichero, se indica con ... (tres puntos consecutivos).

Cuando algo es opcional, se indica introduciéndolo entre corchetes.

Para resaltar información o conceptos importantes se utiliza la negrita.

3. Instalando Debian GNU/Linux

3.1. Introducción

Seguidamente, se explicará como se instaló el sistema Debian GNU/Linux en lo que posteriormente se convertiría en el servidor de instalaciones semiautomáticas, aquel ordenador dónde se instala el paquete FAI.

La instalación de una distribución Debian GNU/Linux si bien no es tan sencilla cómo una instalación de Microsoft Windows u otras distribuciones Linux, no debería acarrear mayores problemas. Casi todos los pasos, son del tipo “Siguiente” de Windows, en los que básicamente el usuario ha de clicar el botón “siguiente” sin hacer nada más. Si bien, en otros pasos, se nos pide cierta información que, casi siempre, puede contestarse sin tener grandes conocimientos de informática ni de Linux.

Si bien, para una instalación “seria”, en un equipo de producción que dé servicios, se recomienda que sea llevada a cabo por el administrador de sistemas del lugar o una persona con unos conocimientos mínimos.

Para la instalación del sistema Debian GNU/Linux hemos utilizado la versión Sarge 3.1 que es la estable en estos momentos (Agosto 2005).

3.2. La secuencia de instalación

Se explicará brevemente como instalar un sistema Debian GNU/Linux. Cómo se llevó a cabo en este proyecto.

Al principio de todo, nos aparece una pantalla que nos pide que introduzcamos algunas opciones y luego pulsemos Enter para iniciar la instalación.

No seleccionamos ninguna opción y pulsamos Enter para iniciar la instalación.

Nos muestra un mensaje en inglés, que nos pide que seleccionemos el idioma de una lista, seleccionamos el idioma Español (Spanish).

Seguidamente, nos pide que introduzcamos el País o Región donde nos encontramos, seleccionamos de una lista España. Y luego nos pide el mapa de teclado, la distribución de éste, seleccionamos Español.

El instalador nos muestra varios mensajes: Detectando hardware, analizando CD-ROM, Cargando módulos del instalador, de forma consecutiva.

Se nos insta a la Selección del Interfaz de Red Primaria. En nuestra caso, este servidor estará conectado externamente con otros equipos a través del interfaz de red eth1 que seleccionamos.

Ahora el instalador intentará configurar la red por DHCP, por si algún servidor nos ofrece una dirección IP, en nuestro caso recibe una IP incorrecta, así que posteriormente la cambiaremos. Se nos pregunta el nombre de la máquina y el nombre del dominio al que pertenece: `hydra` y `maia.ub.es`, respectivamente. Aplicar los que corresponda en cada caso, si no se dispone de dominio puede inventarse uno del tipo `cualesquiera.pt` o dejar el campo en blanco.

El proceso continúa detectando el hardware del equipo, y luego nos muestra una tabla nombrada `Particionado` con las opciones `Borrar hda`, `Borrar hdc` y `editar manualmente`. Depende de como esté configurado nuestro disco duro y de los discos que tengamos nos mostrará éstas u otras opciones.

Luego, se muestra un cuadro que nos pide que elijamos un `Esquema de particionado`: `Novatos`, `todo en la misma partición (recomendado)`, `Equipo ofimática` y `Equipo multi-usuario`. La primera opción, en principio,

debería ser válida para cualquier equipo monousuario o PC casero, si bien, para el caso que nos trae no es una opción recomendable. Así que elegimos la última opción, Equipo multi-usuario.

Nos crea una tabla de particiones que no nos sirve de nada, así que la borramos y creamos la siguiente, utilizando los pasos que se detallan a continuación:

1. Pulsamos Enter encima de la partición que queramos modificar o del espacio de disco no particionado.
2. Seleccionamos Crear una nueva partición, luego el tamaño absoluto o relativo, y decimos su punto de anclaje (en que directorio la montamos), su sistema de ficheros y si queremos que el arranque esté activado o no, por defecto no activaremos el arranque en ninguna.

Con esto y después de varios pasos nos queda:

<i>Disco duro y Partición</i>	<i>Espacio</i>	<i>Punto de anclaje</i>	<i>Sistema de Ficheros</i>
hda1	2 GB	/	ext3
hda5	5 GB	/usr	ext3
hda6	3,5 GB	/var	ext3
hda7	300 MB	intercambio	intercambio
hdc1	10 GB	/home	ext3
hdc5	500 MB	/tmp	ext3
hdc6	300 MB	intercambio	intercambio

Cuadro 1. Particionado

Con lo que se puede observar que nuestro servidor dispondrá de dos discos duros (hda y hdc), cada uno de 10 GB.

Una vez hecho esto, seleccionamos la opción Finalizar el particionado y escribir los cambios en disco, pidiéndonos confirmación. Confirmamos y seguimos con el proceso.

Se nos muestran sucesivos mensajes informativos del proceso de instalación: Creando sistemas de ficheros ext3, Instalando el sistema base Debian, Instalando cargador GRUB.

En este punto se nos preguntará “¿Instalar el cargador de arranque GRUB en el registro principal de arranque?” y seleccionamos que Sí.

Se nos mostrará un mensaje indicándonos que la instalación se ha completado y que saquemos el CD/DVD con el que lo hemos instalado para reiniciar el ordenador. Lo retiramos y pulsamos Enter. Seguidamente, se nos pedirá alguna información acerca de la configuración del sistema.

3.3. Arrancando Debian GNU/Linux por primera vez

Una vez reiniciado el equipo, nos mostrará el mensaje “Arrancando...” y un cuadro que nos dirá que ahora se procederá a la configuración del sistema base Debian.

Nos pregunta si el reloj está fijado a GMT, contestamos Sí. Lo cual quiere decir que la hora del reloj está fijada según GMT y no según nuestra hora local.

Luego nos pide que seleccionemos la Zona Horaria, Madrid es la zona horaria que nos corresponde.

Nos pide la clave para el superusuario (root) y la confirmación, la volvemos a introducir. Seguidamente, nos pide los datos para crear una cuenta de usuario: Nombre completo, nombre cuenta, contraseña, confirmación.

Se solicita el método de acceso para apt (el sistema de paquetes de Debian). Seleccionamos todas y cada una de las fuentes que queramos añadir para el acceso a apt: cdrom, http, ftp, sistema de ficheros local. En nuestro caso, como disponemos de un mirror en la red lo editamos manualmente, e introducimos dos entradas, una para los paquetes estándar y otra para los paquetes de seguridad.

Debido a que el servicio DHCP no nos ha dado una IP válida, hemos tenido que abrir otro terminal (**Alt-F2**) y configurar la red manualmente. Editando el fichero `/etc/network/interfaces` y luego reiniciando los servicios de red con los comandos **ifdown -a e ifup -a**.

```
auto lo
iface lo inet loopback

# The primary network interface
auto eth1
iface eth1 inet static
    address 10.1.1.195
    netmask 255.255.255.0
    network 10.1.1.0
    broadcast 10.1.1.255
    gateway 10.1.1.254
    # dns-* options are implemented by the resolvconf package, if installed
    dns-nameservers 10.1.1.1
    dns-search maia.ub.es
auto eth0
iface eth0 inet static
    address 192.168.1.1
    netmask 255.255.255.0
    network 192.168.1.0
    broadcast 192.168.1.255
```

Y volvemos a la consola de instalación presionando **Alt-F1**.

Instalación semiautomática de nodos en un cluster

Se nos pide que seleccionemos los programas que queramos instalar, no seleccionamos ninguno y continuamos.

Configuramos el correo con la opción, solamente entrega local; sin red. Introducimos el destinatario del correo de `root` y `postmaster`, seleccionamos el usuario que hemos creado durante el proceso de instalación.

Configuramos el teclado seleccionando las opciones: según arquitectura -> qwerty y Spanish.

Seguidamente, se nos muestran muchos mensajes en consola, el sistema nos va informando sobre la instalación de varios paquetes software hasta que la instalación finaliza.

El proceso de instalación ha finalizado.

Para asegurarnos que el sistema está actualizado, ejecutamos las siguientes instrucciones:

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

Mediante el comando `apt-get install kernel-image-2.6.8-2-686-smp` instalamos un núcleo del sistema para equipos duales, con dos procesadores, que se encuentra en nuestro mirror y ha sido compilado según nuestras necesidades locales.

```
# apt-get install kernel-image-2.6.8-2-686-smp
```

4. Configurando los servicios

4.1. Introducción

En este capítulo se explicará como configurar los servicios (servidores, demonios, o daemons si se prefiere) de nuestro servidor de instalaciones semiautomáticas. Los servicios son programas que se ejecutan en segundo plano de forma continua y ofrecen, como su nombre indica, servicios específicos como el correo electrónico, servidores de nombres de dominio (DNS), etc. Estos programas no tienen relación con ningún terminal o consola y, por tanto, no interactúan con el usuario.

Se diferenciarán los servicios que necesita FAI para su instalación, de los que son necesarios para nuestro servidor en concreto, si bien, estos no tienen porque ser disjuntos. Esta diferenciación se hace para que queden claros los requisitos que necesita el paquete FAI.

4.2. Servicios

4.2.1. *xinetd*

Primero de todo se ha sustituido al viejo `inetd`, el superdemonio de internet, por el nuevo `xinetd` (8). Este se encarga de escuchar peticiones para distintos servicios que no disponen de su propio daemon corriendo en segundo plano, y de ejecutar el programa adecuado según la petición.

Esta sustitución se debe a las mejoras en seguridad y configuración de `xinetd`. En nuestro caso porque deseamos que `xinetd` se limite a escuchar las peticiones de un servicio a través de sólo un interfaz de red (tarjeta de red), el interfaz que conecta con la red local.

Para ello hemos ejecutado:

Instalación semiautomática de nodos en un cluster

```
# apt-get install xinetd
```

Para más información consultar `xinetd.conf` (5) y los servicios disponibles en `/etc/services`, el cual nos muestra entre otras cosas el nombre del servicio, el puerto lógico que utiliza y el tipo de datagramas (TCP y UDP básicamente).

4.2.2. *rsh (remote shell)*

Seguidamente, y como se necesitará una comunicación rápida entre los nodos de nuestro cluster, instalamos el módulo `rsh` (remote shell), para ejecutar una shell de forma remota. Es lo mismo que un Secure Shell pero sin encriptar los datos, y por tanto más rápida.

```
# apt-get install rsh-client rsh-server
```

En este caso hemos instalado la versión cliente (no daemon, claro) que nos servirá para conectarnos remotamente a otro equipo y la servidor, para que se conecten al equipo servidor de instalaciones.

Para configurarlas hemos creado los archivos `login` y `shell` en `/etc/xinetd.d/`:

```
service login
{
    socket_type      = stream
    protocol        = tcp
    wait            = no
    user            = root
    server          = /usr/sbin/in.rlogind
    bind            = 192.168.1.1
}

service shell
{
    socket_type      = stream
    protocol        = tcp
    wait            = no
    user            = root
    server          = /usr/sbin/in.rshd
    bind            = 192.168.1.1
}
```

Como puede observarse, estos servicios sólo podrán utilizarse desde la red interna (bind=192.168.1.1) y no desde fuera, por cuestiones de seguridad.

Hemos actualizado la configuración de seguridad en el fichero `rsh` de la carpeta `/etc/pam.d/` sustituyendo la 2 línea no comentada por:

```
auth    required      pam_rhosts_auth.so hosts_equiv_rootok
```

Sirve para cuando se intente acceder vía `rsh`, también se consulte fichero `/etc/hosts.equiv` para comprobar los equipos en los que se confía, cuenta de superusuario (`root`) incluida.

4.2.3. *ntpd (Network Time Protocol)*

Este protocolo es necesario para sincronizar los relojes de todos los nodos de nuestra red. Lo cual es necesario para poder sincronizar los ficheros de distintos hosts.

```
# apt-get install ntp-simple ntp-date
```

El primer paquete es el servidor, el servicio de hora, el segundo es el comando para sincronizarse.

Para configurarlo adecuadamente, hemos tenido que actualizar el nombre del servidor horario en `/etc/ntp.conf`.

```
...
server zeus.maia.ub.es
...
```

4.2.4. *rwhod (server status daemon)*

Este servicio se utiliza, básicamente, para comprobar las máquinas que están funcionando mediante el comando `ruptime` (1). Además también puede mostrar quien ha iniciado sesión en cualquier equipo de la red local.

Para instalarlo, junto con el comando `rwho`, se ha procedido de la siguiente manera:

```
# apt-get install rwho rwhod
```

Para configurar este paquete, hemos editado el script de inicio `rwhod` en `/etc/init.d`. Ya que sólo queremos que el demonio `rwhod` escuche por la tarjeta de red de la red interna. Hemos cambiando la siguiente línea de `/etc/init.d/rwhod`:

```
--exec /usr/sbin/rwhod -- -b
```

Por esta otra:

```
--exec /usr/sbin/rwhod -- -i eth0
```

4.2.5. *bind9 (Berkeley Internet Name Domain)*

El paquete `bind9` ofrece el servicio de DNS (Domain Name System). El sistema de nombres de dominios es el servicio que se encarga de conversión de direcciones IP a nombres y viceversa. Es el servicio que se encarga que podamos acceder a `http://www.google.es` sin tener que recordar la dirección IP real del ordenador (66.102.9.99). Es el servicio de “páginas amarillas” de Internet.

Su instalación es un tanto compleja, y escapa al ámbito de este trabajo. Se recomienda su uso si tenemos muchos equipos en la red y queremos el acceso a estos mediante un nombre del tipo `h01`, `amadeus`, `zeta`, `ganimedes`, `orion` o cualquier otro. Si

no siempre podemos usar nuestro pequeño gran amigo `/etc/hosts`, editando allí todos nuestros equipos con sus correspondientes IPs.

Para instalarlo procedemos con:

```
# apt-get install bind9
```

Una vez instalado, hay que configurarlo el demonio. En su configuración intervienen varios ficheros, empecemos por el principio.

Primero, hay que notificarlo al fichero `/etc/nsswitch.conf`, que es el fichero de configuración de las Bases de Datos del Sistema y del sistema de Conmutación de los Servicios de Nombres (Name Service Switch). Para ello, tenemos que asegurarnos que aparezca esta línea en él:

```
# hosts:          files dns
```

Esto le dice a los programas que en caso de no conocer los nombres de host, primero mire en `/etc/hosts` (indicado por files) y luego, sólo si no puede resolver el nombre, mira al servidor de DNS. Se ponen en este orden porque la búsqueda a través de `/etc/hosts` suele ser más rápida. Pero no tiene la potencia ni la centralización que un DNS nos ofrece para la correcta gestión de nombres.

Si no disponemos del fichero `nsswitch.conf` (5), deberemos modificar el fichero `/etc/host.conf` (5) y nos aseguramos que aparezca esta línea:

```
order hosts,bind
```

Luego modificamos el fichero `/etc/resolv.conf` es el fichero que le dice al servicio DNS como trabajar, en nuestro caso concreto:

```
search . granja.maia.ub.es maia.ub.es
nameserver 192.168.1.1
nameserver 10.1.1.1
```

Significa que primero busque el nombre sin cualificar, si no lo encuentra pruebe de añadirle el sufijo `granja.maia.ub.es` (precedido de un punto, claro), y si sigue sin funcionar lo intente con el nombre más el sufijo `maia.ub.es` (con el punto también).

Finalmente, las bases de datos del servidor DNS se configuran en `/etc/bind/named.conf (5)`, `/etc/bind/named.conf.local` y en otros archivos que son incluidos por estos primeros.

Para comprobar que el servicio está funcionando, utilice el siguiente comando:

```
# ps -edaf | grep named | grep -v grep
```

Si el servicio no está funcionando, el comando no mostrará ninguna salida. Para arrancar el servicio ejecute:

```
# /etc/init.d/bind9 start
```

Si cambia la configuración del servicio tendrá que ejecutar:

```
# /etc/init.d/bind9 reload
```

Para más información consultar el fantástico [DNS Como](#)⁹.

4.3. Servicios para FAI

4.3.1. apt-proxy

Este servicio, a pesar de no ser estrictamente necesario, es muy recomendable para la instalación de varios nodos usando FAI. Con este demonio podremos ahorrar ancho de banda en nuestras comunicaciones externas y tener todos los paquetes de forma casi inmediata, ya que el tráfico dentro de nuestra red local (y en general en todas las redes locales) es mucho más rápido. Se encarga de hacer de proxy de un mi-

⁹ DNS Como - <http://es.tldp.org/COMO-INSFLUG/es/pdf/DNS-Como.pdf>

rror Debian, esto es, se solicita un paquete software al servidor proxy, si éste no lo tiene, lo solicita al servidor mirror que tenga configurado el paquete apt-proxy y una vez que ha bajado el paquete lo almacena en caché (disco duro) por si alguna otra máquina necesita instalar este mismo paquete posteriormente.

Podría pensarse que se está perdiendo o desperdiciando espacio en disco, si bien, las ventajas son mucho mayores: se genera menos tráfico externo, no ocupa tanto como crear un mirror Debian completo y permite, una vez bajado el paquete, una acceso mucho más rápido, a la velocidad máxima que nos permita nuestra red local, además nos aísla de los posibles problemas de las redes externas (cortes, saturación, etc.). Lo cual nos ahorrará mucho tiempo y puede que también dinero en las comunicaciones.

Para instalarlo:

```
# apt-get install apt-proxy
```

Una vez instalado, hay que configurarlo, lo cual no es muy complicado. Tendremos que retocar el fichero **apt-proxy-v2.conf** (man apt-proxy.conf), para ello antes haremos una copia de seguridad con el comando **cp (1)**. Editamos el fichero y realizamos los siguientes cambios:

```
;; Server IP to listen on
address = 192.168.1.1

;; Server port to listen on
port = 9999
...
;; Backend servers, in order of preference
backends = http://jane.uab.es/debian/
...
[debian-non-US]
;; Debian debian-non-US archive
;timeout will be the global value
backends = http://jane.uab.es/debian-non-US/
...
[security]
```

Instalación semiautomática de nodos en un cluster

```
;; Debian security archive
backends = http://security.debian.org/
...
```

address debe contener la dirección del interfaz de red del servidor proxy por el que se escuchan las peticiones. **port** es el puerto por donde escuchan las peticiones, por aquel por donde se presta el servicio. Y en cada sección de **backends** debe aparecer la lista de URLs de nuestros mirrors preferidos, puede aparecer una sola o varias URLs en cada sección. En nuestro caso, usamos el mirror Debian de la Universidad Autónoma de Barcelona y el servidor de Debian de seguridad.

Luego editamos el fichero `/etc/apt/sources.list` (5), para que todas nuestras nuevas peticiones de paquetes las gestione el mismo proxy, y añadimos las entradas que apunten al equipo que ofrece el proxy, en nuestro caso, **h01** y por el puerto **9999**.

```
# Proxy server
deb http://h01:9999/debian sarge main contrib non-free
deb http://h01:9999/security/ sarge/updates main
```

Nos aseguraremos que en el fichero **hosts**, en `/etc/` aparece este nombre correctamente.

```
127.0.0.1      localhost.localdomain  localhost
192.168.1.1    h01
```

Ahora sólo nos queda reiniciar el servicio mediante:

```
# /etc/init.d/apt-proxy restart
```

Si desea instalar un mirror Debian en vez de usar un proxy, diríjase al apéndice A, donde se detalla la forma de instalarlo.

4.3.2. *tftp-hpa (Trivial File Transfer Protocol de HPA)*¹⁰

Este servicio es necesario para la transferencia del núcleo del sistema del servidor de instalaciones a los clientes. Sólo es necesario cuando se arranca vía tarjeta de red, por tanto puede omitirse si se va a utilizar un CDROM o un disco flexible.

Para instalarlo:

```
# apt-get install tftp-hpa
```

Luego para configurarlo, deberemos crear el fichero `tftp` dentro del directorio `/etc/xinetd.d/` con el siguiente contenido:

```
service tftp
{
    socket_type      = dgram
    protocol        = udp
    wait            = yes
    user            = root
    bind            = 192.168.1.1
    server          = /usr/sbin/in.tftpd
    server_args     = -s /boot/fai
}
```

4.3.3. *NFS (Network File System)*

Es un sistema de archivos distribuido para un entorno de red local. Posibilita que distintos sistemas conectados a una misma red accedan a ficheros remotos como si se tratara de locales. Originalmente fue desarrollado por Sun Microsystems.

Es uno de los pilares sobre los que se apoya FAI.

Para instalarlo procederemos con:

```
# apt-get install nfs-user-server
```

Y para configurar una parte del servicio, nos aseguraremos que la línea del fichero `/etc/nsswitch.conf` (5) aparece tal como a continuación:

```
netgroup:          files nis
```

¹⁰ Acerca de TFTP - <http://es.wikipedia.org/wiki/TFTP>

Para completar su configuración esperaremos que FAI haga parte de su trabajo al ser instalado y configurado.

4.3.4. DHCP (*Dynamic Host Configuration Protocol*)

Es un protocolo de red en el que un servidor provee los parámetros de configuración a las computadoras conectadas a la red informática que los requieran (máscara, puerta de enlace y otros) y también incluye un mecanismo de asignación de direcciones de IP).

Es otro de los grandes pilares de FAI.

Para instalarlo simplemente hay que:

```
# apt-get install dhcp3-server
```

Una vez instalado nos aparecerá una pantalla con un cuadro de diálogo que nos preguntará a que interfaz (tarjeta) de red debe responder a las peticiones DHCP. En nuestro caso, hemos puesto `eth0` que es el interfaz de la red interna.

La versión del `dhcp.conf` que se instala es autoexplicativa, pero puede que un tanto larga y difícil de entender, por lo que se expondrán los cambios y añadidos que se han hecho en el fichero (`/etc/dhcp3/dhcp.conf`).

Se ha eliminado la línea `ddns-update-style`.

```
server-identifier h01;

deny unknown-clients;

allow booting;
allow bootp;
...
# option definitions common to all supported networks...
option domain-name "";
option domain-name-servers 192.168.1.1;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
```

```

option ntp-servers 192.168.1.1;
use-host-decl-names on;
default-lease-time 4294967295;
max-lease-time 4294967295;
...
subnet 192.168.1.0 netmask 255.255.255.0 {}

host h02 { fixed-address 192.168.1.2; hardware ethernet
00:50:DA:3C:FC:07;}

```

Como puede verse, el fichero se divide en dos secciones lógicas, la parte de opciones, que va desde el principio hasta justo antes de la línea que empieza por subnet, y la parte de hosts, en los que debemos indicar la IP que asignaremos a cada host según su dirección física (MAC address). En esta segunda sección, deberemos añadir todos los hosts de nuestro cluster.

Más adelante, al instalar y configurar FAI, volveremos sobre este fichero.

4.4. Otros paquetes

Adicionalmente, se han instalado otros paquetes software que no entran en la categoría de servicios y no son necesario para FAI, pero ayudan en la gestión de los nodos.

Entre ellos cabe destacar `dsb` (1), dancer's (o distributed) shell, que se utiliza para ejecutar comandos remotamente en varias máquina y a la vez.

Otro de los interesantes es `rsync` (1), que se utiliza para sincronizar ficheros entre hosts. Viene a ser como un `rccp` (remote copy) pero mucho más rápido, flexible y eficiente, ya que únicamente transmite por la red las diferencias entre los ficheros usando un potente algoritmo.

5. Instalando FAI

5.1. Introducción

En este capítulo se explica, de la forma más sencilla y concisa posible, cómo se instalará y configurará FAI, intentando no caer en excesivos tecnicismos ni grandes párrafos pero siendo suficientemente explicativos para sentar unas buenas bases acerca de su funcionamiento, de forma que el usuario que lo utilice pueda adaptarlo a sus necesidades y cubrir todas las expectativas en la instalación de nuevos nodos. Se intenta que este manual no sea una receta cerrada y dé soluciones o alternativas en caso de dificultades. Este trabajo pretende ser útil para todo aquel que quiera usar FAI, procurando evitar las dificultades que un servidor se ha encontrado con la documentación oficial.

Se considera oportuno de todos modos, que se tenga siempre en cuenta y presente la página web de FAI (<http://www.informatik.uni-koeln.de/fai/>) en todo momento, ya que la documentación oficial y el paquete son actualizados de forma periódica. Además en ella aparecen ficheros de registro (logs), comentarios, referencias a canales de IRC donde nos darán apoyo, listas de correo, instalaciones realizadas con éxito, entre otros.

5.2. Instalando FAI

Bien, empezaremos instalando el paquete principal de FAI y otros paquetes que nos serán útiles más adelante para dejar atrás ya el comando `apt-get` (8).

```
# apt-get install fai fai-kernels syslinux debootstrap mknbi
```

Con esto ya tenemos todo lo necesario para nuestro servidor de instalaciones.

5.3. Configurando FAI

Antes de que FAI cree toda la estructura de directorios y ficheros necesaria para su funcionamiento, debemos configurarlo adaptando los ficheros `/etc/fai/sources.list`, `/etc/fai/fai.conf` y `/etc/fai/nfsroot.conf`.

5.3.1. Configurando `/etc/fai/sources.list`

Tanto el servidor como los clientes utilizan el contenido de `/etc/fai/sources.list` y la variable `FAI_DEBMIRROR`, si se utiliza un mirror vía NFS, definida en `/etc/fai/fai.conf`. Si su servidor tiene instaladas varias tarjetas de red, y diferentes nombres de host o IPs para cada una, debe usar el nombre o IP que los clientes conozcan, es decir, el nombre o IP de la red interna.

FAI utiliza comando `debootstrap` (8) para crear el árbol de ficheros `nfsroot` en `/usr/lib/fai/nfsroot`, o en la ruta que indicada en la variable `NFSROOT` definida en `/etc/fai/fai.conf`. El cual necesita unos 230 MB de espacio en disco.

Ya que utilizamos el proxy para los obtener los paquetes, simplemente tenemos que copiar el fichero `sources.list` de `/etc/apt` a `/etc/fai`, con el siguiente comando:

```
# cp /etc/apt/sources.list /etc/fai/sources.list
```

5.3.2. Configurando `fai.conf` (`/etc/fai/fai.conf`)

Este fichero se utiliza para configurar el paquete FAI (no el espacio de configuraciones para los clientes).

En nuestro caso, hemos cambiado las siguientes y establecido las siguientes variables, marcadas en negrita. Para una descripción completa, consulte el apéndice B.

Fichero `/etc/fai/fai.conf`.

```

# $Id: fai.conf,v 1.86 2005/05/19 09:28:33 lange Exp $

# /etc/fai/fai.conf -- configuration for FAI

# installserver must be the name seen by the install clients
installserver=h01
SERVER=h01

# the name of the Debian mirror, this is also hardcoded in
/etc/fai/sources.list
mirrorhost=h01

# added by Frank 20050914
mirrorport=9999

# Don't use the variable FAI_SOURCES_LIST any more.
# Instead use /etc/fai/sources.list

# Access to Debian mirror via NFS mounted directory
# If FAI_DEBMIRROR is defined, install clients mount it to $MNTPOINT
#FAI_DEBMIRROR=$mirrorhost:/files/scratch/debmirror

# if your install server has multiple ethernet device, use this one to
# determine its hostname. Default eth0. Set to the interface to which
# the Beowulf clients are connected.
SERVERINTERFACE=eth0

...
LOGUSER=fai
# use ssh or rsh for copying log files to user fai and for logging in
# from install clients to install server
FAI_REMOTESH=rsh
FAI_REMOTECP=rcp

# set protocol type for saving logs, default is rcp/scp. Set to ftp if
desired.
FAI_LOGPROTO=
# Name of log-server. If undefined, the install server will be used.
LOGSERVER=h01
# writable directory on remote server, when using FTP protocol
LOGREMOTEDIR="upload"
# password for login to log server, when using FTP protocol
LOGPASSWD=

# the configuration space on the install server
FAI_CONFIGDIR=/home/fai/config
# the location of the config space, as seen by the install client
# it can also be overwritten with T170 via BOOTP
FAI_LOCATION=$installserver:$FAI_CONFIGDIR

# the following variables are read only for most users

# mount point where the mirror will be mounted
MNTPOINT=/mnt2

# directory on the install server where the nfsroot for FAI is

```

Instalación semiautomática de nodos en un cluster

```
# created, approx size: 160MB, also defined in bootptab or dhcp.conf
#NFSROOT=/usr/lib/fai/nfsroot

# Frank 20050914
NFSROOT=/home/fai/nfsroot

# the local configuration directory on the install client
FAI=/fai

# the type of operating system (linux, sunos)
OS_TYPE=`uname -s | tr /A-Z/ /a-z/`
```

<i>Variable</i>	<i>Explicación</i>
installserver	Debe contener el nombre del servidor que ven los clientes .
SERVER	<p>Sólo debe establecer esta variable en caso que haga las instalaciones de los clientes usando un disco flexible y quiera monitorizarlas con el comando faimond.</p> <p>El valor de la variable ha de ser igual que installserver, o en su caso, el servidor a través del que quiera monitorizar las instalaciones.</p> <p>Esta variable no está explicada en la documentación oficial de FAI. Para entender su funcionamiento hemos habido de consultar los fuentes (scripts) de FAI y las listas de correo de FAI.</p> <p>Si se arranca vía tarjeta de red, se define esta variable a través de la opción <code>server-identifier</code> de la configuración del dhcp <code>/etc/dhcp3/dhcp.conf</code>.</p>
mirrorhost	El nombre del mirror Debian.
mirrorport	<p>El número de puerto a través del cual se accede al mirrorhost.</p> <p>Esta variable la hemos creado nosotros, ya que no existía en FAI. Y nos es necesaria ya que el mirror Debian, nuestro proxy, sirve por el puerto 9999 (y no por el 80, que es el valor por defecto de las peticiones http).</p>
LOGUSER	Nombre de la cuenta de usuario en el servidor donde se almacenarán los ficheros de registro (logs) de las

<i>Variable</i>	<i>Explicación</i>
	instalaciones y que puede cambiar el núcleo que será arrancado a través de la red cuando se use el dhcp para arrancar los clientes . Si no se define, no se guardarán los registros en el servidor , lo cual no es nada recomendable.
LOGSERVER	Nombre del servidor que se utilizará para guardar los registros de las instalaciones. Toma el valor de installserver si no se establece.
FAI_CONFIGDIR	Directorio del espacio de configuraciones del servidor .
NFSROOT	Directorio del servidor donde se creará el nfsroot para FAI, que luego será utilizado por los clientes.

Cuadro 2. Variables de `fai.conf`

Recuerde respetar las mayúsculas y minúsculas en los nombres de las variables.

5.3.3. Configurando `make-fai-nfsroot.conf` (`/etc/fai/make-fai-nfsroot.conf`)

Es el fichero de configuración donde se establecen las opciones para crear el árbol de ficheros `nfsroot` del servidor. Es el árbol de ficheros que utilizarán los clientes durante el proceso de instalación. Todos los clientes lo comparten mediante NFS y lo montan con permisos de sólo lectura.

Nuestro `/etc/fai/make-fai-nfsroot.conf`, con las variables nuevas o cambiadas resaltadas en negrita. Para una descripción completa, consulte el apéndice C.

```
# these variables are only used by make-fai-nfsroot(8)
# here you can use also variables defined in fai.conf (like $mirrorhost)

# Add a line for mirrorhost and installserver when DNS is not available
# on the clients. This line(s) will be added to $nfsroot/etc/hosts.
NFSROOT_ETC_HOSTS="192.168.1.1 $mirrorhost"

FAI_DEBOOTSTRAP="sarge http://$mirrorhost:$mirrorport/debian"
#FAI_DEBOOTSTRAP="sarge file:/files/scratch/debian"

# your extra packages which will be installed into the nfsroot, space
# separated
NFSROOT_PACKAGES="cfengine expect"
```

Instalación semiautomática de nodos en un cluster

```
# this local repository holds your local packages that can be installed to
# the install clients. Don't forget to create the index file Packages.gz!
FAI_LOCAL_REPOSITORY="deb file:/fai/files packages/"

# the encrypted root password on all install clients during
# installation process; used when log in via ssh; pw is: fai
FAI_ROOTPW="56hNVqht51tzc"

# this kernel package will be used when booting the install clients
KERNELPACKAGE=/usr/lib/fai/kernel/kernel-image-2.6.8-fai_1_i386.deb
#KERNELPACKAGE=/usr/lib/fai/kernel/kernel-image-2.4.27-fai_1_i386.deb

# location of a identity.pub file; this user can log to the install
# clients in as root without a pass; only useful with FAI_FLAGS="sshd"
#SSH_IDENTITY=/home/admin/.ssh/identity.pub

# which of DHCP and/or BOOTP should the server create setups for.
# Default are to create setups for both
FAI_BOOT="dhcp bootp"

# export $NFSROOT to this netgroup or this range of IP addresses
# (eg. FAICLIENTS="192.168.1.0/24")
#FAICLIENTS=`hostname -i | sed -e 's/\(.*\)\. [0-9]*\/1.0\/24/' | tr -d ' '
#`

# Frank 20050914
FAICLIENTS=`hostname -i | sed -e 's/\(.*\)\. [0-9]*\/1.0\/255.255.255.0/'
| tr -d ' '`

...

```

<i>Variable</i>	<i>Explicación</i>
NFSROOT_ETC_HOSTS	En esta variable hay que añadir una línea para el <code>\$mirrorhost</code> y otra para el <code>\$installserver</code> , cuando no hay un servidor de nombres disponible. Estas líneas serán añadidas al <code>\$nfsroot/etc/hosts</code> . Hemos establecido el valor en 192.168.1.1 <code>\$mirrorhost</code> , que es la IP tanto de nuestro <code>\$mirrorhost</code> como de nuestro <code>\$installserver</code> , h01.
FAI_DEBOOTSTRAP	Se utiliza el comando <code>debootstrap</code> (8) para construir el <code>nfsroot</code> . El cual necesita el nombre de la distribución y la dirección del mirror Debian. En nuestro caso, queremos que la distribución sea la sarge y le pasamos el nombre y el puerto del proxy.
FAI_LOCAL_REPOSITORY	Esta variable indica el repositorio local para aquellos

<i>Variable</i>	<i>Explicación</i>
	<p>paquetes no estándar que deseemos instalar.</p> <p>Al crear el <code>nfsroot</code> se añadirá esta dirección a las fuentes de apt.</p> <p>Hemos establecido el valor en:</p> <pre>"deb file:/fai/files packages/"</pre>
FAI_ROOTPW	<p>Esta variable contiene el valor encriptado de la clave de la cuenta de superusuario (root) en el sistema del cliente.</p> <p>Por defecto es <code>fai</code>. Si desea cambiarlo, utilice el comando <code>mkpasswd (1)</code> del siguiente modo:</p> <pre># mkpasswd clave_a_encriptar</pre> <p>La salida del comando es lo que debe asignar a esta variable, no olvide poner las comillas.</p>
KERNELPACKAGE	<p>Es el núcleo que se utilizará para arrancar a los clientes durante el proceso de instalación. Utilice alguno del paquete <code>fai-kernels</code>.</p> <p>Puede encontrar los <code>fai-kernels</code> en <code>/usr/lib/fai/kernel/</code>.</p>
FAICLIENTS	<p>Se encarga de añadir las líneas al <code>/etc/exports</code> para exportar el <code>nfsroot</code>. En nuestro caso, lo único que hemos cambiado es el formato de la máscara de subred de <code>/24</code> a <code>/255.255.255.0</code>, que es el formato que necesita nuestro servidor NFS.</p>

Cuadro 3. Variables de `make-fai-nfsroot.conf`

5.4. *fai-setup* (creando el `nfsroot`)

Para crear y configurar el `nfsroot` según las indicaciones de los ficheros `/etc/fai/fai.conf` y `/etc/fai/make-fai-nfsroot.conf` debe ejecutar `fai-setup (8)`:

```
# fai-setup
```

Instalación semiautomática de nodos en un cluster

```
Creating FAI nfsroot can take a long time and will need more than 230MB
disk space in /usr/lib/fai/nfsroot.
Creating nfsroot for sarge using debootstrap
dpkg: base-passwd: dependency problems, but configuring anyway as you
request:
  base-passwd depends on libc6 (>= 2.3.2.ds1-4); however:
    Package libc6 is not installed.
dpkg: base-files: dependency problems, but configuring anyway as you
request:
.
.
.
Creating base.tgz
`/etc/fai/sources.list' -> `/usr/lib/fai/nfsroot/etc/apt/sources.list'
Upgrading /usr/lib/fai/nfsroot
Adding additional packages to /usr/lib/fai/nfsroot:
module-init-tools dhcp3-client ssh file rdate hwinfo portmap
bootpc rsync wget rsh-client less dump reiserfsprogs usbutils
hdparm smartmontools parted mdadm lvm2
dnsutils ntpdate dosfstools cvs jove xfsprogs xfsdump
sysutils dialog discover mdetect libnet-perl netcat libapt-pkg-perl
grub lilo read-edid kudzu hwtools dmidecode
Backing up any LVM2 metadata that may exist...done.
Creating SSH2 RSA key; this may take some time ...
Creating SSH2 DSA key; this may take some time ...
Restarting OpenBSD Secure Shell server: sshd.
Checking available versions of rmt, updating links in /etc/alternatives
...
(You may modify the symlinks there yourself if desired - see `man ln'.)
Updating rmt (/usr/sbin/rmt) to point to /usr/sbin/rmt-dump.
Updating rmt.8.gz (/usr/share/man/man8/rmt.8.gz) to point to
/usr/share/man/man8/rmt-dump.8.gz.
Recovering jove files ... Done.
Error : Temporary failure in name resolution
`/etc/fai/fai.conf' -> `/usr/lib/fai/nfsroot/etc/fai/fai.conf'
`/etc/fai/make-fai-nfsroot.conf' ->
`/usr/lib/fai/nfsroot/etc/fai/make-fai-nfsroot.conf'
`/etc/fai/menu.lst' -> `/usr/lib/fai/nfsroot/etc/fai/menu.lst'
`/etc/fai/sources.list' -> `/usr/lib/fai/nfsroot/etc/fai/sources.list'
Shadow passwords are now on.
DHCP environment prepared. If you want to use it, you have to enable the
dhcpcd and the tftpd-hpa daemon.

Image Creator for MBA ROMs v1.00
Usage: imggen [OPTION] inputfile outputfile
  -a,   Add 3Com MBA/BootWare support
  -r,   Remove 3Com MBA/BootWare support from image file
  -i,   Show information on an image
  -h,   Help screen

In filename: /boot/fai/installimage
Out filename: /boot/fai/installimage_3com
Adding MBA support...
MBA support has been succesfully added
BOOTP environment prepared.
Removing `diversion of /sbin/discover-modprobe to
/sbin/discover-modprobe.distrib by fai'
make-fai-nfsroot finished properly.      <= *
Re-exporting directories for NFS kernel daemon...done.
```

```
You have no FAI configuration. Copy the simple examples with:
cp -a /usr/share/doc/fai/examples/simple/* /usr/local/share/fai
Then change the configuration files to meet your local needs.
FAI setup finished.                <= *
```

Para asegurarse de que las cosas han ido bien, debe ver las líneas que están marcadas con un asterisco y resaltadas con negrita, de otro modo, algo ha ido mal. Si sólo obtiene líneas blancas puede que sea debido a un bug de konsole, en su caso utilice `xterm`.

Puede encontrar los ficheros de registro (logs) enteros de `fai-setup` en la página web de FAI¹¹.

Ignore los mensajes de advertencia (warnings) de `dpkg` acerca de las dependencias. Si tiene problemas ejecutando `fai-setup`, la fuente probablemente esté en el fichero de configuración `/etc/fai/make-fai-nfsroot.conf`. Para solucionarlo, edítelo, corrija los errores y en vez de ejecutar `fai-setup`, ejecute `make-fai-nfsroot -r` (recover/recuperar) (8). Si añade el parámetro `-v` el comando le mostrará por pantalla mucha más información acerca de lo que está haciendo. Si quiere crear un fichero de registro (log) para revisarlo posteriormente puede ejecutar:

```
# sudo /usr/sbin/make-fai-nfsroot -r -v 2>&1 | tee make-fai-nfsroot.log
```

`fai-setup` añade algunas líneas a `/etc/exports` para exportar `nfsroot` y el espacio de configuraciones a todos los hosts que pertenecen al netgroup `faiclients`.

En nuestro caso, las líneas creadas en `/etc/exports` eran incorrectas, ya que contenían la IP externa del servidor, por lo que hemos cambiado estos valores a la IP interna y reiniciado el servicio NFS usando las instrucciones que se detallan un poco

¹¹ Página Web de FAI - <http://www.informatik.uni-koeln.de/fai/>

más adelante. Puede que pueda solucionar este error cambiando el fichero de configuración `make-fai-nfsroot.conf`, asignando el siguiente valor a **FAICLIENTS**:

```
FAICLIENTS=`host h01 | sed -e 's/.*address \([0-9\.]*\)\/\1/' | sed -e 's/\(.*\)\/\.[0-9]*\/\1.0\/255.255.255.0/`
```

Donde **h01** es el nombre interno de su servidor. De todos modos, no se ha verificado su funcionamiento y sólo se pretende dar como ejemplo de posible solución.

Todos los **clientes** han de pertenecer a dicho grupo. Los **netgroups** están definidos en `/etc/netgroup` o en su correspondiente mapa NIS. Un ejemplo de éste puede encontrarse en `/etc/share/doc/fai/examples/etc/netgroup`. Para más información, **netgroup** (5). Después de cambiar los **netgroups** debe recargar las configuraciones del servidor NFS usando:

```
# /etc/init.d/nfs-user-server reload
```

En nuestro caso, no disponíamos de dicho fichero, por lo que copiamos el ejemplo de `/etc/share/doc/fai/examples/etc/netgroup` en `/etc/netgroup` y lo editamos según nuestras necesidades. Realizar cambios en éste es trivial, no le será difícil. Recuerde siempre reiniciar el servidor NFS como se ha explicado.

Además, **fai-setup** crea una cuenta de usuario **fai** (definida por `$LOGUSER`) si no existe. Los ficheros de registro (logs) de todos los **clientes** se graban al directorio de este usuario (`/home/fai`). A su vez, esta cuenta ha de tener permisos de escritura en `/boot/fai` para cambiar los enlaces simbólicos a la imagen del núcleo que son arrancados por los **clientes**, cuando éstos usan el arranque a través de tarjeta de red.

Para cambiar los propietarios de los ficheros, ejecutamos:

```
# chown -R fai:nogroup /home/fai/*  
# chown -R fai:nogroup /boot/fai/*
```

Para cambiar los permisos, es necesario:

```
# chmod -R g+w /boot/fai/*
```

Después de esto, FAI está instalado correctamente en el **servidor**, pero no tiene las configuraciones para las instalaciones de los **clientes**, el **espacio de configuraciones**. Es recomendable empezar con los ejemplos `/usr/share/doc/fai/examples/simple/` e ir cambiándolos según nuestras propias necesidades. Para copiarlos y establecer los permisos adecuados ejecute:

```
# cp -a /usr/share/doc/fai/examples/simple/* /home/fai/
# chown -R fai:nogroup /home/fai
```

Si realiza cambios en el fichero `/etc/fai/fai.conf` o en el fichero `/etc/fai/make-fai-nfsroot.conf`, deberá reconstruir el `nfsroot` ejecutando `make-fai-nfsroot`. Si simplemente quiere reemplazar el núcleo antiguo por uno nuevo, añade las banderas `-k` o `-K` al comando `make-fai-nfsroot`. Esto no recreará todo el `nfsroot`, sólo actualizará el núcleo y los módulos de éste dentro del `nfsroot`.

5.5. Reconfigurando el servidor DHCP

Llegados a este punto debe reconfigurar el servidor de DHCP, añade estas líneas en su fichero `/etc/dhcp3/dhcp.conf`, antes de la definición de los hosts. Esta definición también puede hacerse dentro de las definiciones de subred (`subnet`) y dentro de las definiciones de equipos (`host`), si son necesarias configuraciones distintas.

```
option root-path "/home/fai/nfsroot";
```

Como puede observar, en `option root-path` hemos puesto el valor de donde está nuestro `nfsroot`, si ha utilizado otro valor, no olvide introducir el correspondiente.

Instalación semiautomática de nodos en un cluster

Seguidamente, reiniciamos el servidor DHCP mediante el comando:

```
# /etc/init.d/dhcp3-server restart
```

6. Preparando el arranque

6.1. Introducción

Antes de arrancar por primera vez cualquier **cliente**, tenemos que escoger que medio usaremos para ello. En nuestro caso hemos probado primero el arranque por medio de disco flexible. Y una vez hemos comprobado que todo funciona correctamente, hemos optado por un arranque más avanzado, el que se hace a través de red usando boot PROM, lo cual es mucho más práctico ya que evita que tengamos que retirar físicamente el disco flexible o CDROM del **cliente**.

Antes de poder arrancar debemos configurar el servicio de **dhcp** para que reconozca a los **clientes**, y les asigne la dirección IP que les corresponda según su dirección física (o dirección MAC). Para ello, tenemos que recopilar las direcciones físicas. En las nuevas máquinas, normalmente en la BIOS ya se muestran estas direcciones, pero en máquinas antiguas esta opción no existe, por tanto, deberemos utilizar otro método para ello.

6.2. Recopilando las direcciones MAC (direcciones físicas)

En esta sección se describirá como recopilar las direcciones MAC (direcciones físicas) de las tarjetas de red de nuestros **clientes**.

Para poder instalar los **clientes** de forma automática es necesario que un demonio **BOOTP** o **DHCP** reconozca los hosts, de otro modo, los **clientes** no arrancarán. Puede aprovechar un primer arranque para obtener las direcciones MAC de las tarjetas de red.

Debe obtener todas las direcciones MAC de los **clientes** y asignarles un nombre y una dirección IP a cada uno, mediante **BOOTP** o **DHCP**. Se recomienda **DHCP** al ser un servicio más avanzado. En nuestro caso hemos utilizado **DHCP**.

Como se ha dicho antes, en los equipos nuevos, normalmente estas direcciones son mostradas a través de la BIOS, si bien, en los antiguos esta opción no existe.

Por tanto, si disponemos de un equipo nuevo, accederemos a la BIOS (pulsando la tecla pertinente, normalmente <Supr>, <F1> o <F2>) y buscaremos en ella, si no la encontramos utilizaremos el siguiente método.

El otro método consiste en crear un disco de arranque con el servidor, por tanto, introduzca un disco flexible en la disquetera de su equipo (disquetera de 3½ pulgadas y 1.44 MB) e introduciremos el siguiente comando:

```
# make-fai-bootfloppy
```

Este comando dispone páginas del manual, en la sección 8.

Como no hay información específica de ningún **cliente**, este disco flexible es idóneo para todos los **clientes**. Pueden especificarse parámetros adicionales al núcleo de arranque del disco flexible.

Una vez creado el disco, realizamos lo siguiente en el **servidor**:

```
# tcpdump -qte broadcast and port bootpc >/tmp/mac.list
```

Y arrancamos el **cliente** con el disco flexible.

El cliente enviará peticiones **DHCP** (paquetes broadcast) para recibir una IP, como ningún servidor los reconocerá, fallará el arranque.

Entonces, vuelva a la consola del **servidor** y aborte **tcpdump** (9) usando **Ctrl-C**. Obtendrá una lista de todas las MAC del cliente ejecutando este comando:

```
# perl -ane 'print "\U$F[0]\n"'/tmp/mac.list|sort|uniq
```

Después de esto, asigne un nombre y una IP a cada una de estas direcciones MAC en la configuración del servidor DHCP y en el fichero **/etc/hosts**. Se recomienda

encarecidamente que apunte las direcciones MAC y el nombre del host en una etiqueta pegada al chasis del ordenador.

En nuestro caso, hemos añadido las siguientes líneas en el fichero **/etc/dhcp3/dhcp.conf**:

```
host h02 { fixed-address 192.168.1.2;
           hardware ethernet 00:50:DA:3C:FC:07; }
```

También hemos añadido la línea adecuada en nuestro **/etc/hosts**:

```
192.168.1.2    h02
```

Una vez hecho esto, sólo ha de reiniciar el servidor DHCP mediante:

```
# /etc/init.d/dhcp3-server restart
```


7. Secuencia de instalación

7.1. Introducción

En este capítulo se explicará brevemente la secuencia de instalación. A pesar de que su lectura no es necesaria para el uso de FAI, se recomienda leer este capítulo, el cual es básicamente teórico, para entender el funcionamiento del paquete FAI. Los conocimientos que se adquirirán en este capítulo nos servirán para saber a donde dirigarnos en caso de querer cambiar las configuraciones de los clientes almacenadas en el **espacio de configuraciones**.

7.2. La secuencia de instalación

Durante la instalación, después que el núcleo de Linux haya arrancado en el **cliente**, se efectúan las siguientes tareas:

1. Configuración de FAI (set up)
2. Carga de los módulos del núcleo (kernel modules)
3. Definición de clases
4. Definición de variables
5. Particionado de los discos locales
6. Creación y montaje de los sistemas de ficheros (filesystems) locales
7. Instalación de paquetes software
8. Ejecución de scripts de configuración específicos
9. Se graban ficheros de registro (logs)
10. Reinicio del nuevo sistema instalado en el **cliente**

Puede definir programas o scripts adicionales, los **hooks** o ganchos. Los **hooks** se detallarán más adelante.

El tiempo de duración de la instalación depende más del software instalado, de la velocidad del disco duro y del procesador que de la velocidad de la red. Por tanto, no debe preocuparnos que la red sea un cuello de botella cuando se instalen varios clientes simultáneamente. El tiempo variará de 3 minutos a unos 30 minutos o más.

En nuestro caso particular, la instalación duró 6 minutos, en la cual se instalaron más de 300 paquetes.

7.3. Monitorizando la instalación

Puede monitorizar la instalación de los **clientes** usando el comando **faimond** (8). Todos los **clientes** comprueban si el demonio se está ejecutando en el **servidor**. Cada vez que una tarea empieza o acaba, se envía un mensaje al servidor donde se encuentra **faimond**. Este demonio muestra los mensajes en la salida estándar.

Si quiere usar esta característica de FAI y utiliza el arranque desde disco flexible o CDRom, defina la variable **SERVER** en el archivo **fai.conf** tal como se describe en la sección 5.3.2, en este mismo manual. Si utiliza el arranque vía tarjeta de red, debe especificar el servidor en la configuración del servidor dhcp (**/etc/dhcp3/dhcp.conf**), como se muestra a continuación:

```
server-identifier h01;
```

Y reiniciar al servidor de dhcp. Esto ya se realizó en la sección 4.3.4 de este documento.

7.4. Configurando FAI (set up)

Una vez que el cliente ha arrancado, FAI sólo ejecuta el script `/usr/sbin/fai` (cuelga del `nfsroot` en el servidor). Éste es el script principal, y controla la secuencia de instalación de FAI. No se ejecuta ningún otro script de `/etc/init.d`.

Se crea un disco RAM (ramdisk) y se monta en `/tmp`, el cual es el único directorio con permisos de escritura hasta que se montan los sistemas de ficheros locales.

Se reciben parámetros adicionales del demonio DHCP o BOOTP y del espacio de configuraciones si se monta vía NFS desde el servidor a `/fai`, como en nuestro caso.

Cuando se han creado los dos terminales virtuales y el demonio `ssh` se arranca bajo demanda finaliza la configuración de FAI .

7.5. Definiendo clases, variables y cargando los módulos del núcleo

En este punto el script `fai-class` (1) se encarga de definir las clases. Para ello, se ejecutan varios scripts de `/fai/class` (`/home/fai/config/class` en el servidor), concretamente todos aquellos que coinciden con el patrón `[0-9]*` (los que empiezan con un dígito) son interpretados como nombres de clases. Los scripts que terminan con `.source` son cargados, por tanto, puede definir nuevas clases añadiéndolas a la variable `newclasses` (pueden ver un ejemplo en `20-hwdetect.source`). La salida de estos scripts es ignorada. El cliente define estas clases para indicar que pertenece a ellas.

Una **clase** está definida o no definida, pero no tiene ningún valor. Sólo aquellas que están definidas son de interés para el **cliente**. Puede encontrar la descripción de todas las clases en `/usr/share/doc/fai/classes_description.txt`. Si va a crear nuevas clases, es recomendable que las documente convenientemente.

El script `20-hwdetect.source` carga los módulos del núcleo bajo demanda.

Después de definir las clases, cualquier fichero `*.var` con un prefijo que coincida con una de las clases definidas será ejecutado para definir variables. En alguna de ellos podría definir la variable `FAI_ACTION` y otras. Por defecto, `FAI_ACTION` se define a través del comando `fai-chboot (8)`, pero sólo podrá utilizar `fai-chboot` cuando arranque vía tarjeta de red. Cuando arranque a través de disco flexible o CDROM, defina estas variables en `/fai/class/LAST.var`.

7.6. Particionando los discos duros y creando los sistemas de ficheros

Para particionar los discos duros locales, sólo un fichero es seleccionado de `/fai/disk_config` (`/home/fai/config/disk_config` en el **servidor**) usando clases. Este fichero incluirá la descripción de las particiones de todos los discos duros, donde se crearán los sistemas de ficheros y su tipo, y como serán montados.

También es posible conservar las particiones actuales o los datos de ciertas particiones. Esta tarea se lleva a cabo mediante el comando `setup_harddisks`, que usa el comando `sfdisk (8)`. El formato para la configuración del fichero se explica en `/usr/share/doc/fai/README.disk_config`. La configuración de nuestro caso concreto será expuesta en el siguiente capítulo.

Durante el proceso de instalación los sistemas de archivos locales se montan de forma relativa a `/tmp/target`. Así, por ejemplo, `/tmp/target/home` será `/home` en el nuevo sistema.

7.7. Instalando paquetes

Una vez creados los sistemas de ficheros, éstos se encuentran vacíos. Ahora se instalará el sistema base Debian y los paquetes solicitados.

Primero el archivo base es desempaquetado, el comando `install_packages` (8) instala todos los paquetes utilizando `apt-get` (8) o `aptitude` (1), que se encargan de resolver las dependencias entre paquetes.

Las **clases** también son usadas en la selección de los ficheros de configuración en `/fai/package_config/` para instalar los paquetes. El formato de estos ficheros se describirá posteriormente, en el próximo capítulo.

7.8. Configuración específica del host

Una vez terminada la instalación de los paquetes solicitados. Éstos deben configurarse adecuadamente, para ello puede ejecutar varios scripts. Los scripts que coincidan con el nombre de una **clase** definida en `/fai/scripts` serán ejecutados. Si `/fai/scripts/nombre_de_clase/` es un directorio, todos los scripts que coincidan con `[0-9][0-9]*` del directorio serán ejecutados. Estos scripts serán descritos posteriormente.

7.9. Grabando los archivos de registro (logs)

Cuando finalizan todas las tareas de instalación, los ficheros de registros se escriben en `/var/log/fai/$HOSTNAME/install/` en el nuevo sistema y en la cuenta del usuario del **servidor**, si se definió la variable `$LOGUSER` en el fichero `/etc/fai/fai.conf`.

Este método usa `rsh/rcp` o `ssh/scp` por defecto. Puede seleccionar otro método, por ejemplo `FTP`, para grabar los ficheros de registro (logs) en el **servidor** remoto, definiendo `$FAI_LOGPROTO` en `/etc/fai/fai.conf`. Para más información consulte la documentación de FAI.

7.10. Reiniciando el nuevo sistema

Finalmente, el sistema se reinicia de forma automática si la palabra “**reboot**” fue añadida a la variable `FAI_FLAGS`.

Para evitar que vuelva a arrancar otra vez a través de la red, puede usar el comando `fai-chboot (8)`, su funcionamiento será explicado en el siguiente capítulo. Si usó un disco flexible, recuerde que debe quitarlo físicamente del **cliente**. De otro modo, la instalación volverá a ejecutarse.

8. Configurando los clientes

8.1. Introducción

En este capítulo se explicará como hemos configurado la instalación de los clientes. FAI dispone de un potente sistema, este sistema se basa en el uso de clases. Para hacerlo se trabaja sobre el **espacio de configuraciones**, en nuestro caso se encuentra en `/home/fai/config`.

Antes de ponerse manos a la obra deben planearse convenientemente las instalaciones. Para ello, intente encontrar las respuestas a estas preguntas:

- ¿Mis **clientes** necesitan un núcleo especial?
- ¿Mi hardware es uniforme? ¿Lo será en el futuro?
- ¿Cuáles son los nombres de los **clientes**?
- ¿Qué tipología de red usarán?
- ¿Qué IPs quiero asignarles?
- ¿Cómo quiero particionar los discos duros?
- ¿Qué paquetes deben ser instalados?
- ¿Qué demonios/servicios tienen que ofrecer mis **clientes**?
- ¿Necesito usar clases o sólo necesito una configuración para todos los **clientes**?
- ¿Cómo llevaré a cabo las copias de seguridad?
- ¿Tengo corriente suficiente para todos los **clientes**?
- ¿Cómo refrigeraré los **clientes**?
- ¿Qué hay acerca de las cuentas de usuario de los **clientes**?

- ¿Cómo imprimiré?

Otras cuestiones importantes son: disposición del teclado, zona de tiempo, tarjeta gráfica, sistema de correo electrónico, directorios exportados e importados vía NFS, etc.

8.2. El espacio de configuraciones

En el espacio de configuraciones se almacena toda la información sobre como instalar los clientes. En nuestro caso, se encuentra en `/home/fai/config` y en sus subdirectorios. Y es montado en los clientes en `/fai`.

Los siguientes subdirectorios aparecen en él e incluyen varios ficheros:

<i>Subdirectorio</i>	<i>Descripción</i>
class	scripts y ficheros para definir clases y variables y para la carga de los módulos del núcleo.
disk_config	Ficheros de configuración para el particionado de los discos duros y la creación del sistema de ficheros.
debconf	Contiene todos los datos para <code>debconf</code> (8). El formato es el mismo que usa <code>debconf-set-selections</code> (8). En nuestro caso, no hemos modificado nada de este directorio.
package_config	Ficheros con la lista de paquetes que serán instalados o desinstalados.
scripts	Scripts para la configuración personalizada del cliente.
files	Ficheros usados por los scripts de personalización (scripts). La mayoría de ellos se encuentra en una estructura de subárbol que refleja el árbol ordinario de directorios. Por ejemplo, las plantillas del fichero <code>nsswitch.conf</code> están en el directorio (sí, directorio) <code>/fai/files/etc/nsswitch.conf</code> , en éste se encuentran los <code>nsswitch.conf</code> con el nombre de la clase que corresponde. Por ejemplo, para el fichero <code>nsswitch.conf</code>

<i>Subdirectorio</i>	<i>Descripción</i>
	para todos los equipos de la clase ABC, se deberá copiar con el fichero a <code>/fai/etc/files/nsswitch.conf/ABC</code> . Estos ficheros no se copian automáticamente en el cliente, para ello use el comando <code>fcopy (8)</code> en los scripts de personalización.
<code>files/packages</code>	Este directorio es el referenciado por defecto en el fichero de configuración de FAI <code>/etc/fai/make-fai-nfsroot.conf</code> como <code>FAI_LOCAL_REPOSITORY</code> y es añadido al apt sources del nfsroot por el comando <code>make-fai-nfsroot (8)</code> . Igual que antes, estos paquetes no se instalan automáticamente, pero estarán disponibles durante la instalación en sus scripts de personalización usando <code>install_packages</code> o <code>apt-get (8)</code> . Estos paquetes también pueden indicarse en <code>package_config</code> .
<code>hooks</code>	Los <code>hooks</code> o <code>ganchos</code> son programas o scripts definidos por el usuario, y se ejecutan durante el proceso de instalación.

Cuadro 4. El espacio de configuraciones

El comando principal de instalación `fai (8)` usa todos estos subdirectorios en el orden listado, excepto en el caso de los `hooks`.

En nuestro caso, trabajaremos, como hemos dicho antes, a partir de los ejemplos (consulte la sección 5.4 de este documento).

8.3. Configuración

Omitiendo las largas explicaciones del manual de FAI, en esta sección se explicará como hemos configurado la instalación de nuestros clientes. Como se ha comentado varias veces, se ha partido de los ejemplos y se han hecho los cambios necesarios. Para más información consulte los apéndices y la documentación oficial de FAI.

A pesar de contar con clientes homogéneos, se han utilizado las clases, por tanto, el uso de clientes heterogéneos no debería suponer mayor dificultad.

A partir de este punto, las explicaciones parten de que el arranque e instalación de los clientes se realiza a través de la tarjeta de red, lo cual es mucho más práctico, rápido y óptimo que el uso del disco flexible.

8.3.1. Reconfigurando DHCP y /etc/hosts

Otra vez, nos toca reconfigurar el fichero de configuración del demonio de DHCP, para arrancar a través de red, debemos añadir la opción `filename`, a nivel global o dentro de cada host en `/etc/dhcp3/dhcp.conf`. Como por ejemplo:

```
...
filename "pxelinux.0";
...
```

Además, si no tiene definido el **cliente** donde va a realizar la instalación, hágalo ahora, como se ha descrito anteriormente, en la sección 4.3.4. No olvide reiniciar el servidor dhcp, para que los cambios tengan efecto, ya que hemos cambiado la configuración.

Si no lo ha hecho ya, también defina los **clientes** en `/etc/hosts` y en el servidor de DNS, reiniciando también el servidor de DNS.

8.3.2. Definiendo clases

No se han definido nuevas clases. Como disponemos de clientes homogéneos, no nos ha sido necesaria definir nuevas. Para definir nuevas clases, el script `fai-class` (1) ejecuta todos los scripts que coinciden con el patrón `[0-9][0-9]*` en el directorio `/fai/class` del **cliente**. Adicionalmente, un fichero con el nombre del **cliente** puede contener una lista de las clases. Para más información consulte `fai-class` (1).

Nuestro **cliente** definía las siguientes clases: DEFAULT, h02, FAIBASE, LINUX, I386, DHCP, GRUB y LAST, tomadas de los ejemplos.

La lista de todas las clases definidas se almacena en la variable `$classes` y se graba en `/tmp/fai/FAI_CLASSES`.

8.3.3. Definiendo variables

Primero hemos editado el fichero `/home/fai/class/DEFAULT.var`, en él hemos editado las variables `FAI_KEYMAP`, `UTC`, `time_zone` y `rootpw` para definir, la disposición del teclado, si se sigue el sistema UTC para la fecha y la hora, nuestra zona horaria y la clave del superusuario para el nuevo sistema del cliente, respectivamente.

```
...
FAI_KEYMAP=es

# Set UTC=yes if your system clock is set to UTC (GMT), or UTC=no if not.
UTC=yes
time_zone=Europe/Madrid

# root password for the new installed linux system; pw is "fai"
rootpw='CodificacionDeNuestroPassword'
...
```

Hemos establecido la clave del superusuario (root) usando el comando `mkpasswd`, más información en la sección 5.3.3.

Las variables `FAI_ACTION` y `FAI_FLAGS` se han definido con el comando `fai-chboot (8)`, ya que arrancamos a través de la interfaz (tarjeta) de red.

Primero hemos ejecutado el comando:

```
# fai-chboot -o default
```

Para que, por defecto, si no se especifica nada o si no se encuentra el fichero cuyo nombre es la dirección IP del cliente en hexadecimal, no se arranque desde la interfaz (tarjeta) de red, sino que use el siguiente dispositivo de arranque especificado en la BIOS del sistema del cliente, que normalmente será el disco duro.

Para indicarle que el **cliente**, en nuestro caso el host con nombre **h02**, arranque desde red realizamos lo siguiente:

```
# fai-chboot -Ivf verbose, ssgd, createvt, syslogd, reboot h02
```

Con ello le establecemos las variables antes mencionadas, con **FAI_ACTION=install** y **FAI_FLAGS=verbose, sshd, createvt, syslogd, reboot**. Para más información sobre la variable **FAI_FLAGS**, consulte el apéndice D.

Con ello, se creará un fichero en `/boot/fai/pxelinux.cfg/` con la IP en hexadecimal del equipo **h02**, que una vez al finalizar la instalación se deshabilitará automáticamente según las configuraciones de ejemplo copiadas en nuestro **espacio de configuraciones** para que al reiniciar el equipo no se vuelva a reinstalar el sistema y arranque desde el siguiente dispositivo de arranque de la BIOS, en nuestro caso el disco duro donde se ha realizado la instalación. Recuerde establecer el propietario y los permisos adecuados en el directorio `/boot/fai`, tal como se describió en la sección 5.4.

Hemos establecido la variable **FAI_FLAGS** en `verbose, sshd, createvt, syslogd` y `reboot`, para que respectivamente, nos muestre mucha más información al hacer la instalación en el cliente (que será mostrada en el monitor del cliente y almacenada en los ficheros de registro), se pueda acceder al cliente vía ssh durante el proceso de instalación, cree dos terminales virtuales más, por si accedemos al cliente de forma local y queremos realizar alguna operación en él durante el proceso de instalación, inicie el servidor de registro del sistema y la última opción es para que el cliente reinicie automáticamente sin esperar ninguna interacción nuestra.

Para que todo esto funcione, el primer dispositivo de arranque en nuestra BIOS debe ser la tarjeta de red conectada a la red local, el siguiente dispositivo debe ser el disco duro donde se instalará el sistema Debian GNU/Linux vía FAI. Una vez, establecidos estos valores no tendrá porqué cambiarlos.

Llegados a este punto, nuestro cliente debería arrancar a través de la tarjeta de red y realizar una instalación. Pero antes de que pueda realizar una instalación de forma correcta, debemos configurar los parámetros de ésta según nuestras necesidades.

8.3.4. Configuración de los discos duros locales

Siguiendo la documentación de FAI acerca la configuración de los discos, en el fichero `/usr/share/doc/fai/README.disk_config.gz` y el ejemplo que hemos copiado, hemos configurado el disco duro de nuestro equipo como se describe a continuación en el fichero `/home/fai/config/disk_config/FAIBASE:`

```
# <type> <mountpoint> <size in mb> [mount options]      [;extra options]

disk_config disk1
primary  swap                40-500      rw
primary  /                    3020-3030   rw,errors=remount-ro ; -j ext3
primary  -                    50-        rw,nosuid   ; -m 1 -j ext3
```

La configuración es para el disco 1 (`/dev/hda 0` `/dev/sda`).

La primera columna del fichero indica el tipo de partición, en nuestro caso hemos seleccionado que todas sean primarias. La segunda opción establece el directorio donde será montada, la primera partición es de swap, la segunda será montada en raíz `/`, y la última no será montada. La tercera columna, establece el tamaño mínimo y máximo de cada partición, separamos estos valores con un guión. Cuando no se establece máximo, como hemos hecho en la última partición, es para que seleccione todo el es-

pacio libre del disco. La cuarta columna son las opciones de montaje de estas particiones, para más información `mount` (8). Y la última columna, indica opciones extra. Hemos seleccionado el sistema de ficheros `ext3` para las particiones que no son de `swap`. Y la opción `-m` se utiliza para reservar bloques para el superusuario (`root`), indicada en porcentaje; por lo que hemos reservado un 1% de espacio en la tercera partición para el superusuario (`root`).

Ha de tenerse en cuenta que para la selección del fichero de configuración del particionado de los discos, FAI también tiene en cuenta las clases. Nosotros hemos seleccionado la clase `FAIBASE`, por eso nuestro archivo de configuración tiene ese nombre. Puede utilizarse cualquier nombre que se corresponda con una clase que el **cliente** haya definido. Tenga en cuenta que FAI sólo seleccionará un fichero dentro de `/disk_config` para particionar nuestros discos. Por tanto, si un **cliente** define varias clases procure que sólo una de estas clases defina un particionado de los discos duros locales, de otro modo, los resultados pueden no ser los que usted espera.

8.3.5. Configuración de los paquetes

Después de particionar el disco duro y de que FAI lo haya montado convenientemente. Para instalar paquetes, como no, también se utilizan los conceptos de clase. Cualquier fichero en `/fai/packages_config` en el **cliente** (`/home/fai/config/package_config` en el servidor) que coincida con el nombre de una de las clases definidas por el cliente se utilizará para la instalación de paquetes. La sintaxis de estos ficheros es muy sencilla, consulte la documentación oficial de FAI para una explicación más detallada. En nuestro caso, hemos usado el fichero `/home/fai/config/packages_config/FAIBASE`:

```

PACKAGES aptitude
adduser
anacron
apt
apt-utils
at
atlas2-base
atlas2-sse2
atlas3-base
atlas3-base-dev
atlas3-headers
...
lilo-
...
kernel-image-2.6.12.2
...
ucf
update
util-linux
util-linux-locales
vim
vim-common
wamerican
whiptail
xfree86-common
xlibmesa-dri
xlibmesa-gl
xlibmesa-glu
xlibs
xlibs-data
xml-core
zlib1g

```

El comando `PACKAGES aptitude` le indica a FAI que utilice el comando **aptitude** (8) para instalar todos los paquetes que se listan a continuación. Además, este comando también permite instalar tareas. Cuando el nombre de un paquete de la lista finaliza con el signo menos (-), como en el caso del paquete `lilo`, se le indica a FAI que deseamos que ese paquete, en vez de ser instalado, sea desinstalado.

FAI también permite añadir una lista de nombres de clases después de los comandos, en nuestro caso `PACKAGES aptitude`, de forma que este comando sólo sea ejecutado cuando la clase correspondiente esté definida. Por tanto, es posible combinar muchos ficheros pequeños en el fichero `DEFAULT`.

Si se especifica un paquete que no existe, éste será eliminado de la lista de instalación. Según la documentación de FAI es posible chequear este fichero con la utilidad `/usr/share/doc/fai/samples/utils/chkdebnames`, pero en nuestra instalación del paquete esta utilidad no se encontraba.

En esta lista también debe especificar sus propios paquetes (aquellos que haya creado usted, no sean estándar o no estén disponibles en el mirror Debian), como en nuestro caso el paquete no estándar `kernel-image-2.6.12.2` que contiene un núcleo personalizado. Mire la siguiente sección para más información.

8.3.6. Configuración de sus propios paquetes

Cuando quiera instalar un paquete no estándar, que no esté en el mirror Debian o que haya creado o modificado usted mismo, debe seguir los siguientes pasos:

Dentro del directorio del **espacio de configuraciones** `files/packages` (en nuestro caso, `/home/fai/config/files/packages`) copie todos aquellos paquetes con sufijo `.deb` y ejecute el siguiente comando:

```
# cd /home/fai/config/files
# dpkg-scanpackages packages /dev/null | gzip -9 >packages/Packages.gz
```

Para crear el índice de ficheros necesario para usar este sencillo repositorio.

Recuerde que anteriormente, asignamos a la variable `$FAI_LOCAL_REPOSITORY` dentro de `make-fai-nfsroot.conf` para tener acceso a este repositorio local. Si no lo hizo entonces, hágalo ahora o no podrá instalar estos paquetes.

Compruebe que esta línea esté en el fichero `etc/apt/sources.list` dentro de su `nfsroot` (`/home/fai/nfsroot/etc/apt/sources.list`), si no es así añádala, de otro modo, `aptitude` no encontrará sus paquetes:

```
...
deb file:/fai/files packages/
...
```

Recuerde que el nombre del paquete en el fichero `packages_config/FAIBASE` es igual al nombre del fichero `.deb` omitiendo todos los caracteres desde el primer subrayado. En nuestro caso, el nombre del fichero del paquete `.deb` es `kernel-image-2.6.12.2_x01_i386.deb` y el nombre del paquete a instalar mediante **aptitude** es `kernel-image-2.6.12.2`, como ha visto anteriormente.

8.3.7. Scripts de personalización

Los scripts de personalización de FAI, se encuentran en el **espacio de configuraciones**, en el directorio `/scripts`, en nuestro caso `/home/fai/config/scripts`.

Esos scripts se utilizan para la configuración de los paquetes y otras acciones que deban hacerse después de la instalación de estos. Estos scripts se ejecutan automáticamente después de la instalación de los paquetes. Si un directorio tiene el nombre de una clase definida, todos los scripts que coincidan con el patrón `[0-9][0-9]*` se ejecutarán por orden alfabético. Para estos puede usar distintos lenguajes de programación (shell, cfengine, perl, expect).

Puede encontrar varios ejemplos totalmente válidos y muy útiles en `scripts (/home/fai/config/scripts)`, ya que hemos copiado en ese directorio los ejemplos.

Para copiar ficheros usando las clases, utilice el comando **fcopy (8)**, si quiere extraerlo use el comando **ftar (8)**.

En nuestro caso, hemos usado estos scripts para copiar muchas configuraciones en el **cliente**, como se explica a continuación.

Los ficheros que hemos copiado, son los siguientes:

```
/etc/passwd
/etc/shadow
/etc/group
/etc/gshadow
/etc/auto.master
/etc/auto_home
/etc/modules
/etc/hosts.allow
/etc/hosts.deny
/etc/hosts
/usr/local/bin/ipmi_sensors
/usr/local/bin/temp
/usr/local/bin/temp_baseboard
/usr/local/sbin
/usr/local/sbin/hwmon
/usr/local/stow/vogle
/usr/local/stow/vogle/include
/usr/local/stow/vogle/include/vogle.h
/usr/local/stow/vogle/lib
/usr/local/stow/vogle/lib/hershey
/usr/local/stow/vogle/lib/hershey/astrology
/usr/local/stow/vogle/lib/hershey/cursive
/usr/local/stow/vogle/lib/hershey/cyrillic
/usr/local/stow/vogle/lib/hershey/futura.l
/usr/local/stow/vogle/lib/hershey/futura.m
/usr/local/stow/vogle/lib/hershey/gothic.eng
/usr/local/stow/vogle/lib/hershey/gothic.ger
/usr/local/stow/vogle/lib/hershey/gothic.ita
/usr/local/stow/vogle/lib/hershey/greek
/usr/local/stow/vogle/lib/hershey/japanese
/usr/local/stow/vogle/lib/hershey/markers
/usr/local/stow/vogle/lib/hershey/math.low
/usr/local/stow/vogle/lib/hershey/math.upp
/usr/local/stow/vogle/lib/hershey/meteorology
/usr/local/stow/vogle/lib/hershey/music
/usr/local/stow/vogle/lib/hershey/script
/usr/local/stow/vogle/lib/hershey/symbolic
/usr/local/stow/vogle/lib/hershey/times.g
/usr/local/stow/vogle/lib/hershey/times.i
/usr/local/stow/vogle/lib/hershey/times.ib
/usr/local/stow/vogle/lib/hershey/times.r
/usr/local/stow/vogle/lib/hershey/times.rb
/usr/local/stow/vogle/lib/libvogle.so
/usr/local/stow/vogle/lib/libvogle.so.1.3
/usr/local/stow/vogle/lib/libvogle.so.1.3.0
/usr/local/stow/vogle/lib/libvogle.a
/usr/local/stow/vogle/man
/usr/local/stow/vogle/man/man3
/usr/local/stow/vogle/man/man3/vogle.3.gz
/usr/local/stow/vogle
```

Para copiar cada uno de estos ficheros, hemos realizado los siguientes pasos, ejemplificaremos su uso usando el fichero `/etc/group`:

1. Creamos un directorio con el nombre del fichero a copiar dentro del directorio files del **espacio de configuraciones**, ejemplo:

```
# mkdir /home/fai/config/files/etc/group
```

2. Luego, copiamos el contenido del fichero en ese directorio con el nombre de la clase la cual a de copiar el fichero, en nuestro caso **FAIBASE**, ejemplo:

```
# cp rutaHasta/etc/group \  
/home/fai/config/files/etc/group/FAIBASE
```

3. Dentro del fichero , que hemos creado anteriormente, con nombre /home/fai/config/scripts/FAIBASE/60-ficheros, añadimos la línea:

```
fcopy -i /etc/group
```

A cada uno de los ficheros de scripts que haya creado, en nuestro caso sólo 60-ficheros, debe asegurarse de los siguientes puntos:

1. No olvide el sha-bang (#!) al principio del script, o programa que se encargará de interpretar el script, ejemplo:

```
#!/bin/sh  
...
```

2. No olvide dar permisos de ejecución al fichero:

```
# chmod a+x scripts/FAIBASE/60-ficheros
```

3. No olvide cambiar el propietario y grupo del fichero:

```
# chwon fai:nogroup scripts/FAIBASE/60-ficheros
```

Con ello ya tenemos casi todo lo que necesitamos para la instalación del nodo en nuestro cluster. Ahora sólo falta cambiar el dispositivo de arranque del **cliente**.

En caso de disponer de muchos ficheros a copiar se recomienda crear un script para automatizar los pasos descritos anteriormente.

8.3.8. Cambiando el dispositivo de arranque

Ya que la secuencia de arranque establecida en la BIOS, no puede alterarse desde el sistema operativo (al menos hasta la fecha, agosto 2005). Se cambia utilizando el comando `fai-chboot` (8). Esta es una tarea que se ejecuta por defecto en todas las instalaciones.

8.3.9. Hooks (ganchos)

En nuestro trabajo no nos ha sido necesario la definición de ningún hook adicional. Para más información acerca de su uso, consulte la documentación oficial de FAI.

8.3.10. Buscando errores

Si el **cliente** no puede arrancar correctamente desde la tarjeta de red, utilice el comando `tcpdump` (8) para mirar los paquetes Ethernet entre el **cliente** y el **servidor**. También busque entradas en varios ficheros de registro (logs) de `tftpd` (8), `dhcpcd3` (8) o `bootpd` (8) en el servidor:

```
# egrep "tftpd|bootpd|dhcpcd" /var/log/*
```

Si el proceso de instalación finaliza, el **hook faiend.LAST** busca errores en todos los ficheros de registro (logs) y los escribe en el fichero **error.log**. Así, puede examinar primero este fichero en caso de errores. También, el fichero **status.log** le dará el

valor de salida (exit code) del último comando ejecutado en un script. Para estar seguro, debe buscar errores en todos los ficheros de registro (logs).

A veces la instalación parece que se detiene, pero, a menudo, es sólo un script de postinstalación de un paquete que requiere alguna pulsación. Cambie a otro terminal virtual y mire que procesos están ejecutándose con comandos como `top (1)` o `ps tree (1)`. Puede añadir la opción `debug` a `$FAI_FLAGS` para que el proceso de instalación le muestre toda la salida de los scripts de postinstalación y obtener la entrada desde la consola. Puede ver ficheros de registro de ejemplo en la web de FAI.

9. Conclusiones

Creo que se han cumplido todos los objetivos propuestos. Después de lidiar con numerosas tecnologías, que no conocía en profundidad, consultar muchas páginas del manual de Linux, la documentación de FAI (algo desordenada e intrincada a mi modo de ver), las listas de correo, incluso el canal de IRC de este paquete, consultado los fuentes del paquete (casi todos desarrollados en Bourne Shell de Linux -un lenguaje de programación que considero poco elegante y demasiado estricto en sintaxis, para que sea fácil y ágil tanto en su desarrollo como en su depuración-), y con la inestimable ayuda del Sr. Jaume Timoneda, el director de este proyecto, se han conseguido instalar de forma óptima dos equipos.

El primero de estos, era un antiguo equipo en el que hemos realizado las pruebas y aprendido el funcionamiento de FAI mientras se trabajaba con él. El segundo era ya un equipo serio, en rack, uno de los nuevos equipos del departamento de MAiA de la Facultat de Matemàtiques de la Universitat de Barcelona, en la que se ha realizado una instalación real y se han conseguido los objetivos: se ha instalado un sistema Debian GNU/Linux para entrar en producción, sin entorpecer el funcionamiento normal del cluster actual y logrando que éste empiece a funcionar recién instalado. Instalar y listo.

Para lograr estos objetivos, en este proyecto nos hemos basado en el paquete software FAI (Fully Automatic Installation), un paquete estándar en la distribución Debian GNU/Linux. Este paquete es muy potente y flexible, pero algo rudimentario, aprender su funcionamiento ha sido arduo para mí. A pesar de que consta de documentación y páginas del manual de Linux, la información está algo desordenada y falta documentar algunos puntos. Además carece de una receta para seguir paso a paso, un Como (o HOWTO) para utilizarlo. Por tanto al documentar este proyecto, se ha

pretendido que estas carencias queden solventadas. Esperamos haberlo conseguido en una buena medida.

Documentar este proyecto ha sido también complicado. Para conseguir unos resultados óptimos y ofrecer la receta deseada, una vez aprendido el funcionamiento, asimilados los conceptos, lidiado con la documentación y realizado una primera instalación muy sencilla, se ha repetido todo el proceso desde el principio y profundizado en los conceptos más avanzados de FAI, procurando ordenarlos de la forma más conveniente, a la vez, que sentando unas buenas bases para utilizarlo en otros casos de uso distintos.

Deseo, y espero, que este documento sea útil a la comunidad de administradores de sistemas, al menos a aquellos de habla hispana.

A. Instalar un mirror Debian

Si prefiere optar por un mirror Debian, FAI proporciona el script `mkdebmirror` en `/usr/share/doc/fai/examples/utils/`, el cual utiliza `debmirror` (1) y `rsync` (1). Un mirror parcial de Debian de la arquitectura i386 (ordenadores Pcs y compatibles, con microprocesadores Intel o AMD x86) puede ocupar unos 9.0 GB de espacio en disco. La mejor y más rápida forma de acceder a este mirror es vía NFS.

Otra forma de crear el mirror parcial es utilizando el comando `fai-mirror` (1), el cual sólo contiene los paquetes que son usados en las clases del **espacio de configuraciones**. Si prefiere usar el protocolo HTTP para acceder al mirror, instale un servidor web y cree un enlace simbólico al directorio local donde su mirror se encuentra:

```
# apt-get install apache2
# ln -s /files/scratch/debmirror /var/www/debmirror
```

Cree un fichero `sources.list` (5) en `/etc/fai` que apunte al mirror Debian. Y añada la dirección IP del servidor web a la variable `NFSROOT_ETC_HOSTS` en `/etc/fai/make-fai-nfsroot.conf` cuando los clientes no tengan acceso DNS.

B. Referencia fai.conf

En este apéndice se detallan las variables que han de configurarse en el fichero de configuración `/etc/fai/fai.conf`.

Es importante respetar las mayúsculas y minúsculas en los nombres y los valores, de otro modo los resultados no serán los esperados.

<i>Variable</i>	<i>Descripción</i>
FAI_LOCATION	Éste es el nombre del host y del directorio remoto del espacio de configuraciones, que será montado vía NFS. Su valor por defecto es <code>/usr/local/share/fai</code> pero puede cambiarse según sus preferencias. Recuerde que este directorio debe exportarse (<code>/etc/exports</code>) a todos los clientes, de forma que todos los ficheros puedan leerse por el superusuario (root).
FAI_DEBMIRROR	Si tiene acceso vía NFS al mirror Debian, especifique el sistema de archivos remoto. Será montado en <code>\$MNTPOINT</code> , que también ha de ser definida. Esta variable no es necesaria si accede mediante FTP o HTTP.
installserver	Debe contener el nombre del servidor que ven los clientes .
SERVER	Sólo debe establecer esta variable en caso que haga las instalaciones de los clientes usando un disco flexible y quiera monitorizarlas con el comando <code>faimond</code> . El valor de la variable ha de ser igual que <code>installserver</code> , o en su caso, el servidor a través del que quiera monitorizar las instalaciones. Esta variable no está explicada en la documentación oficial de FAI. Para entender su funcionamiento hemos habido de consultar los fuentes (scripts) de FAI y las listas de correo de FAI. Si se arranca vía tarjeta de red, se define esta variable a través de la opción <code>server-identifier</code> de la

<i>Variable</i>	<i>Descripción</i>
	configuración del dhcp <code>/etc/dhcp3/dhcp.conf</code> .
mirrorhost	El nombre del mirror Debian.
mirrorport	El número de puerto a través del cual se accede al mirrorhost . Esta variable la hemos creado nosotros, ya que no existía en FAI.
SERVERINTERFACE	Si el servidor de instalaciones tiene múltiples tarjetas de red, utilice esta variable para determinar su nombre del host. El valor por defecto es eth0 . Establezca el valor del interfaz (tarjeta) de red por el que se vayan a conectar los clientes.
LOGUSER	La cuenta de usuario en el servidor que se encargará de almacenar todos los ficheros de registro (logs) de las instalaciones de los clientes. Y que además puede cambiar el núcleo que es arrancado vía red. Es necesario configurar el fichero <code>.rhosts</code> para esta cuenta y PAM, de manera que el superusuario (root) pueda iniciar sesión desde los clientes sin necesidad de introducir la clave. Esta cuenta ha de tener permisos de escritura en <code>/boot/fai</code> . Si la variable no está definida, esta característica se deshabilita.
LOGSERVER	El nombre del servidor de registro. Si no se define, se usará el nombre del servidor de instalaciones.
LOGREMOTEDIR	Directorio de escritura en el servidor remoto, cuando se usa el protocolo FTP.
LOGPASSWD	Clave para iniciar sesión en el servidor de registro, cuando se usa el protocolo FTP.
FAI_REMOTESH	Utilice <code>ssh</code> o <code>rsh</code> para copiar los ficheros de registro (logs) en la cuenta de usuario y para iniciar sesión en el

<i>Variable</i>	<i>Descripción</i>
	servidor desde los clientes.
FAI_REMOTECP	Lo mismo que la variable anterior, pero para copiar ficheros.
FAI_LOGPROTO	Establece el tipo de protocolo usado para grabar los archivos de registro, el valor por defecto es rcp/scp. Puede elegirse ftp.
FAI_CONFIGDIR	El directorio del espacio de configuraciones en el servidor de instalaciones.
MNTPOINT	El punto de montaje donde el mirror será montado si se usa NFS para el acceso a éste.
NFSROOT	Directorio en el servidor donde FAI creará el nfsroot .
FAI	El directorio de configuración local en los clientes . Esto es, donde se montará el espacio de configuraciones .
OS_TYPE	El tipo de sistema operativo (linux, sunos).

Cuadro 5. Referencia de /etc/fai/fai.conf

C. Referencia make-fai-nfsroot.conf

En este apéndice se detallan las variables que han de configurarse en el fichero de configuración `/etc/fai/make-fai-nfsroot.conf`.

Es importante respetar las mayúsculas y minúsculas en los nombres y los valores, de otro modo los resultados no serán los esperados.

<i>Variable</i>	<i>Descripción</i>
NFSROOT_ETC_HOSTS	En esta variable hay que añadir una línea para el <code>\$mirrorhost</code> y otra para el <code>\$installserver</code> , cuando no hay un servidor de nombres disponible. Estas líneas serán añadidas al <code>\$nfsroot/etc/hosts</code> .
FAI_DEBOOTSTRAP	Se utiliza el comando <code>debootstrap</code> (8) para construir el <code>nfsroot</code> . El cual necesita el nombre de la distribución y la dirección del mirror Debian, que se asignan en esta variable.
FAI_LOCAL_REPOSITORY	Esta variable indica el repositorio local para aquellos paquetes no estándar que deseemos instalar. Al crear el <code>nfsroot</code> se añadirá esta dirección a las fuentes de apt (<code>/etc/apt/sources.list</code>). No hay que olvidar crear el fichero de índice <code>Packages.gz</code> .
FAI_ROOTPW	Esta variable contiene el valor encriptado de la clave de la cuenta de superusuario (<code>root</code>) en el sistema del cliente. Por defecto es <code>fai</code> . Si desea cambiarlo, utilice el comando <code>mcpasswd</code> (1) del siguiente modo: <pre># mcpasswd clave_a_encriptar</pre> La salida del comando es lo que debe asignar a esta variable, no olvide poner las comillas.
KERNELPACKAGE	Es el núcleo que se utilizará para arrancar a los clientes durante el proceso de instalación. Utilice alguno del paquete <code>fai-kernels</code> en <code>/usr/lib/fai/kernel/</code> .

<i>Variable</i>	<i>Descripción</i>
FAICLIENTS	Se encarga de añadir las líneas al <code>/etc/exports</code> para exportar el <code>nfsroot</code> .
NFSROOT_PACKAGES	Paquetes extra que serán instalados en el <code>nfsroot</code> . Deben separarse por espacios y entrecorillarse.
SSH_IDENTITY	Es la ruta donde se encuentra el fichero <code>identity.pub</code> ; este usuario puede iniciar sesión en el <code>cliente</code> como superusuario sin usar ninguna clave. Sólo es útil cuando se el valor <code>sshd</code> en las FAI_FLAGS .
FAI_BOOT	Para que protocolos se han de crear configuraciones. Por defecto se configura para ambos, DHCP y BOOTP.
FAI_DEBOOTSTRAP_OPTS	Son las opciones que se pasan al comando <code>debootstrap</code> para crear el <code>nfsroot</code> . Estas opciones deben adaptarse según la arquitectura que utilizemos.
nfssize	Tamaño del <code>nfsroot</code> . Sólo con propósito de informar.
packages	Son la lista de paquetes que FAI necesita y son instalados en el <code>nfsroot</code> .

Cuadro 6. Referencia de `/etc/fai/make-fai-nfsroot.conf`

D. Referencia de FAI_FLAGS

La variable FAI_FLAGS puede definirse como opción del núcleo al crear un disco flexible de arranque mediante el comando `make-fai-bootfloppy` o mediante el comando `fai-chboot` cuando se arranque el cliente a través de red.

<i>Valor</i>	<i>Descripción</i>
verbose	Muestra más información durante la instalación. Debe ser siempre la primera opción indicada en <code>\$FAI_FLAGS</code> , así las siguientes opciones definidas en <code>\$FAI_FLAGS</code> serán mostradas.
debug	Muestra información para la depuración. Es útil sólo para desarrolladores FAI. Para más información consulte la documentación de FAI.
sshd	Inicia el demonio ssh (secure shell) que permite iniciar sesión remotamente en el sistema.
syslogd	Inicia el demonio de registro (log) del sistema y del núcleo, de manera que los procesos puedan usarlo para registrar información. Esta opción sólo debería usarse si <code>syslogd</code> no está en ejecución en el sistema, por tanto, sólo debería establecerse al inicio de las instalaciones, no en actualizaciones.
createvt	Crea dos terminales virtuales y ejecuta el intérprete de comandos bash si se pulsa <code>Ctrl-C</code> . Puede acceder a las otras terminales mediante <code>Alt-F2</code> o <code>Alt-F3</code> . Si no establece esta opción, al pulsar cualquier tecla se reiniciará el cliente. Esta opción es útil para depurar.
reboot	Reinicia el cliente después de que la instalación haya concluido sin esperar la pulsación de alguna tecla. Esta opción sólo es útil si puede cambiar la imagen de arranque o el dispositivo de arranque de forma automática. Por tanto, no es útil en caso que utilice un disco flexible o un CDROM. Actualmente, sólo debería usarse cuando arranque el cliente desde la tarjeta de red.

Cuadro 7. Referencia de la variable FAI_FLAGS

Instalación semiautomática de nodos en un cluster

E. Las tareas por defecto

En este apéndice se describe de forma detallada las tareas que ejecuta por defecto FAI, desde que arranca el cliente hasta finalizar su instalación.

Una vez que el núcleo ha arrancado, monta el sistema de ficheros raíz vía NFS desde el servidor y el proceso `init` (8) ejecuta el script `/usr/sbin/fai`. Este script controla la secuencia de instalación. No se ejecuta ningún otro script de `/etc/init.d`.

El script de instalación utiliza muchas subrutinas, que están definidas en `/usr/share/fai/subroutines` y en un fichero específico según el sistema operativo¹². Todas las tareas importantes son ejecutadas a través de la subrutina `task` a la cual se añade el nombre de la tarea como opción (ej. `task instsoft`). La subrutina `task` llama a los `hooks` con el prefijo del nombre de la tarea y luego llama la tarea por defecto (definida como `task_name` en el script `subroutines`). La tarea por defecto y sus `hooks` pueden saltarse bajo demanda usando la subrutina `skiptask()`.

Descripción de las tareas por defecto:

- **confdir** Los demonios `syslog` y el log del núcleo son iniciados. La lista de dispositivos (tarjetas) de red se almacena en `$netdevices`. Entonces, el cliente mediante FAI recoge los parámetros adicionales desde el DHCP o BOOTP del servidor. También se definen algunas variables adicionales. Se crea el fichero de configuración del servidor de DNS. El espacio de configuraciones se monta desde el servidor en al cliente en el directorio `/fai`. Si usa CVS consulte la documentación de FAI. Después de esto, se carga el fichero `/fai/hooks/subroutines` si existe. En él puede definir sus propias subrutinas o redefinir las de FAI.

¹² `/usr/share/fai/subroutines-linux` para Linux, `/usr/share/fai/subroutines-sunos` para Solaris.

- **setup** Esta tarea establece la fecha y la hora del sistema. Se definen todas las **FAI_FLAGS** y se crean dos terminales virtuales adicionales si así se solicita en la variable **FAI_FLAGS**. Se inicia el demonio **ssh** (secure shell), también bajo demanda, para poder iniciar sesión remotamente.
- **defclass** Ejecuta **fai-class (1)** para definir las clases usando los scripts y ficheros en `/fai/class` y clases de `/tmp/fai/additional-classes`.
- **defvar** Carga todos los ficheros `/fai/class/*.var` para cada una de las clases que se hayan definido. También se carga el fichero `/tmp/fai/additional.var` si algún **hook** ha escrito definiciones de variables en él.
- **action** Según el valor de la variable **\$FAI_ACTION**, esta subrutina decide que acción llevar a cabo. Por defecto, están definidas las siguientes acciones: **sysinfo**, **install** y **softupdate**. Si **\$FAI_ACTION** tiene otro valor, la acción definida por el usuario se ejecuta si el fichero `/fai/hooks/$FAI_ACTION` existe. Por tanto, puede definir sus propias acciones fácilmente.
- **sysinfo** Se ejecuta cuando **\$FAI_ACTION=sysinfo**. Muestra información acerca del hardware detectado y monta los discos duros locales en modo de sólo lectura en `/tmp/target/partitionname` o considerando el archivo **fstab** si se encuentra dentro de alguna de las particiones. Los ficheros de registro se almacenan en el servidor. Esta acción no ejecuta ninguna instalación.

- **install** Esta tarea controla la secuencia de instalación. Oirá tres pitidos (beeps) del altavoz antes que la instalación empiece. Su principal trabajo es lanzar las otras tareas necesarias y grabar la salida en `/tmp/fai/rcS.log`. Si tiene algún problema durante la instalación, consulte los ficheros que se encuentran en `/tmp/fai/`. Puede encontrar ejemplos en la web de FAI.
- **softupdate** Esta tarea, corriendo dentro de un sistema en producción vía el comando `fai` (8), ejecuta una actualización del software. Para más información, consulte el manual de FAI.
- **partition** Ejecuta la tarea `setup_hddisk` para particionar los discos duros locales. Esta tarea escribe las definiciones de variables para la partición raíz, de arranque y de dispositivo (`$ROOT_PARTITION`, `$BOOT_PARTITION`, `$BOOT_DEVICE`) en `/tmp/fai/disk_var.sh` y crea un fichero `fstab` (5).
- **mountdisks** Monta las particiones creadas de acuerdo al nuevo `/tmp/fai/fstab` de forma relativa a `$FAI_ROOT`.
- **extrbase** Extrae el fichero tar base `base.tgz`, que contiene todos los paquetes necesarios. Éste es una recopilación de los ficheros de sistema básicos de Debian creada por `debootstrap` (8).
- **mirror** Si se accede al mirror Debian vía NFS (esto es, cuando `$FAI_DEBMIRROR` está definida), este directorio será montado en `$MNTPOINT`.
- **debconf** Ejecuta `fai-debconf` (8) para establecer los valores para la base de datos de debconf.

- **prepareapt** Configura `resolv.conf` y algún otro fichero, para la próxima tarea `updatebase`.
- **updatebase** Actualiza la base de datos de paquetes del nuevo sistema y la lista de paquetes disponibles. Además, ejecuta algunos comandos (diversions) en el nuevo sistema usando `dpkg-divert (8)`.
- **instsoft** Instala los paquetes deseados usando los ficheros de clases en `/fai/package_config`.
- **configure** Ejecuta los scripts en `/fai/scripts/` y sus subdirectorios para cada una de las clases definidas.
- **finish** Desmonta todos los sistemas de ficheros en el nuevo sistema y quita las diversions de los ficheros usando el comando `fai-divert`.
- **faiend** Espera la finalización de tareas (jobs) en segundo plano y reinicia la máquina si se definió el flag `reboot` en `$FAI_FLAGS`, en caso contrario espera la pulsación de alguna tecla para reiniciar.
- **chboot** Cambia el enlace simbólico en el `servidor` que indica la imagen del núcleo a cargar en el siguiente arranque desde la tarjeta de red vía TFTP. Sólo se aplica si hemos cargado el núcleo desde la tarjeta de red y no desde disco flexible.
- **savelog** Graba los ficheros de registro (logs) al disco duro local y en la cuenta de usuario `$LOGUSER` en el host `$LOGSERVER` (por defecto es el `servidor`). Actualmente, el fichero `error.log` no es copiado en el `$LOGSERVER`.

F. El concepto de clase

Este apéndice explica de forma detallada como funcionan las clases en FAI.

Las clases determinan que ficheros de configuración se ejecutarán y se utilizan en todas las tareas de instalación. Para determinar que ficheros de configuración usar, un **cliente** mira la lista de clases definidas y utiliza todos los ficheros de configuración que coinciden con el nombre de la clase.

También es posible utilizar únicamente el fichero de configuración de mayor prioridad, ya que el orden de las clases define la prioridad desde el el valor más bajo al más alto.

Hay algunas clases predefinidas (**DEFAULT**, **LAST** y el **nombre del host cliente**), pero las clases también pueden listarse en un fichero o definirse de forma dinámica por los scripts. Así que es fácil definir una clase dependiendo de la información de la subred o del hardware que está disponible en el **cliente**.

Las clases son muy importantes para FAI. Si un cliente pertenece a la clase A, decimos que la clase A está definida. Una clase no tiene ningún valor, simplemente está definida o no. En los scripts, la variable `$classes` tiene una lista, separada por espacios, de las clases definidas.

Las clases determinan como se ejecuta el proceso de instalación. Por ejemplo, un **cliente** puede configurarse como un servidor FTP sólo añadiendo la clase FTP a él (junto con los scripts de clase que realmente lo convierten en tal). Generalmente, puede crear una configuración sólo cambiando o añadiendo las clases a las que pertenece un **cliente**, permitiendo que la instalación del nuevo cliente sea muy fácil. Así no hay que añadir información adicional a los ficheros de configuración si ya dispone de las clases que necesita. Hay diferentes posibilidades para definir clases:

1. Algunas clases por defecto están definidas en todos los **clientes**: DEFAULT, LAST y el nombre del **cliente**.
2. Las clases pueden listarse en un fichero.
3. Las clases pueden definirse por los scripts.

La última opción es una característica muy potente, ya que estos scripts pueden definir clases automática y dinámicamente. Por ejemplo, varias clases pueden definirse si sólo se detecta cierto hardware en el **cliente** o dependiendo del día de la semana.

Todos los nombres de las clases, excepto el nombre del **cliente**, deben escribirse en mayúsculas. Los nombres no pueden contener guiones, ni almohadillas (#), ni puntos pero puede contener subrayados (guiones bajos). Puede encontrar una descripción de todas las clases en `/usr/share/doc/fai/classes_description.txt`.

Los nombres de los **clientes** raras veces son utilizados para los archivos de configuración. En cambio, puede definir una clase podría y luego añadirla a un **cliente** dado. Esto es porque la mayoría de las veces la configuración no es específica para un único **cliente**, sino compartida entre varios **clientes**.

G. Referencia de comandos

En este apéndice se incluye una referencia, ordenada alfabéticamente, de los comandos Linux, no pertenecientes a FAI, que se han utilizado acompañados de una pequeña explicación y ejemplo de uso. Esta pequeña referencia no pretende ser una guía exhaustiva, simplemente una referencia de ayuda para todo aquel que no sea un experto en Linux/Unix. Para más ayuda puede consultar las páginas del manual (man pages).

<i>Comando</i>	<i>Explicación</i>
apt-cache	Este comando se utiliza, entre otras cosas, para buscar paquetes. Ejemplo: <pre># apt-cache search palabraDeBusqueda</pre> Donde <code>palabraDeBusqueda</code> es la palabra utilizada para la búsqueda en el nombre del paquete y su descripción.
apt-get	Sirve para instalar y desinstalar paquetes, entre otros. <pre># apt-get install paquete</pre> <pre># apt-get remove paquete</pre> También puede usarse para actualizar las fuentes y la distribución completa, respectivamente: <pre># apt-get update</pre> <pre># apt-get dist-upgrade</pre>
cat	Sirve para mostrar el contenido de archivos. <pre># cat /etc/hosts</pre>
chgrp	Cambia el grupo al que pertenece los ficheros. <pre># chgrp nombreGrupo ficheros</pre>
chmod	Cambia los permisos de acceso de ficheros. <pre># chmod g+x *</pre>
chown	Cambia el usuario y el grupo propietarios de ficheros <pre>chown -Rv fai:nogroup /boot/fai</pre>
debootstrap	Este comando se utiliza para crear un sistema básico Debian. No se ha ejecutado directamente; lo utiliza FAI para crear el <code>nfsroot</code> .

<i>Comando</i>	<i>Explicación</i>
df	Informa del espacio usado en los sistemas de ficheros. # df
diff	Muestra las diferencias entre ficheros. # diff /etc/hosts /etc/hosts.old
dpkg	Es el gestor de paquetes de Debian (de nivel medio). # dpkg -l
du	Muestra una estimación de lo que ocupan los ficheros # du -sh
egrep	Se utiliza para filtrar la salida de un comando usando expresiones regulares. # ps aux egrep "rwho named"
file	Dice el tipo de un archivo. # file /usr/sbin/fai
find	Se utiliza para buscar ficheros, entre otros. Es un comando muy potente. # find / -name make-fai-nfsroot
grep	Es similar a egrep, pero no acepta expresiones regulares. # grep h01 /etc/hosts
head	Muestra el principio de un fichero. # head /etc/dhcp3/dhcp.conf
ifconfig	Muestra o configura las tarjetas de red. # ifconfig
ifdown	Se utiliza para desconectar un interfaz de red. # ifdown eth0
ifup	Se utiliza para conectar un interfaz de red. # ifup -a
joe	Es un editor de ficheros de texto.
less	Muestra el contenido de un fichero de forma paginada, permitiendo volver atrás y adelante. # less /etc/exports
ln	Sirve para crear enlaces entre ficheros (links). # ln -s /debmirror /var/www/debmirror

<i>Comando</i>	<i>Explicación</i>
locate	Localiza ficheros en la base de datos que siguen un patrón. # locate make-fai-nfsroot.conf
ls	Lista el contenido de los directorios. # ls -last
more	Muestra el contenido de un fichero de forma paginada. Similar a less pero menos potente. # more /etc/fai/make-nfsroot.conf
nano	Es otro editor de ficheros de texto.
netstat	Muestra informaciones acerca de las conexiones de red. # netstat
ps	Muestra los procesos que están en ejecución. # ps -edaf
ruptime	Muestra el estado de los equipos de la red local. # ruptime
rwho	Muestra quien ha iniciado sesión en los equipos de la red local. # rwho -a
sed	Se utiliza para transformar cadenas de texto.
tail	Muestra el final de un fichero. Se utiliza normalmente para hacer seguimientos de ficheros de registro en tiempo real. # tail -f /var/log/syslog
tee	Lee de la entrada estándar y escribe a la vez en la salida estándar y en ficheros. # ps -edaf tee ps_now
top	Muestra las tareas en ejecución. # top
vi	Es un editor muy potente, pero de difícil manejo -al menos al principio-. Es el editor usado durante este proyecto. # vi nombreFichero
whereis	Localiza ficheros binarios, de fuentes y de páginas de manual. # whereis fai
zless	Igual que less pero permite ver ficheros de texto comprimidos mediante gzip, con sufijo .gz.

<i>Comando</i>	<i>Explicación</i>
	# zless /usr/share/doc/fai/fai-guide.txt.gz
zmore	Igual que less pero permite ver ficheros de texto comprimidos mediante gzip, con sufijo .gz. # zmore /usr/share/doc/fai/fai-guide.txt.gz

Cuadro 8. Referencia de comandos

Bibliografía

Thomas Lange. 2005. FAI (Fully Automatic Installation).

<http://www.informatik.uni-koeln.de/fai/fai-guide.pdf> [consulta: Agosto 2005].

F. actualización: 19/05/2005.

Listas de correo de FAI.

<http://www.mail-archive.com/linux-fai@rrz.uni-koeln.de/>. [consulta: Julio, Agosto y Septiembre del 2005].

Canal de IRC de FAI, #fai en <irc://irc.freenode.net>. [consulta: Agosto y Septiembre del 2005]

Páginas de Manual de Linux (man pages).

Wikipedia. En sus versiones inglesa (<http://en.wikipedia.org/>) y castellana (<http://es.wikipedia.org/>). [consulta: Julio, Agosto y Septiembre del 2005].

Manuales de Debian.org. Consultados los siguientes entre Julio y Septiembre del 2005.

Instalación de Debian GNU/Linux:

<http://www.us.debian.org/releases/woody/i386/install.es.pdf>

Manual de referencia de Debian GNU/Linux:

<http://www.us.debian.org/doc/manuals/reference/reference.es.pdf>

APT-HOWTO:

<http://www.us.debian.org/doc/manuals/apt-howto/apt-howto.es.pdf>

The Linux Documentation Project. Consultados varios HOWTOs.

<http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/>. [consulta:

Julio, Agosto y Septiembre del 2005].

DHCP HOWTO:

Instalación semiautomática de nodos en un cluster

<http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/DHCP.pdf>

DNS HOWTO:

<http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/DNS-HOWTO.pdf>

Influg.org. Consultados varios COMOs. <http://insflug.org/comos/comos.php3>.
[consulta: Julio, Agosto y Septiembre del 2005].

DNS Como - <http://insflug.org/COMOs/DNS-Como/DNS-Como.html>

DHCP Mini-COMO:

<http://insflug.org/COMOs/DHCPd-Mini-Como/DHCPd-Mini-Como.html>

Cómo pasar de DOS/Windows a Linux:

<http://insflug.org/COMOs/Dos-Win-Linux-Como/Dos-Win-Linux-Como.html>

Redes en Linux Como:

<http://insflug.org/COMOs/Dos-Win-Linux-Como/Dos-Win-Linux-Como.html>

Servidor de Raíces NFS Mini-COMO:

<http://insflug.org/COMOs/Dos-Win-Linux-Como/Dos-Win-Linux-Como.html>

