Capítulo 4

Servicios Unix

Consideraciones iniciales

Login

Su

Passwd

...

Capítulo 4 Servicios Unix

Entendemos por servicios Unix a aquellas aplicaciones que son comunes a la mayoría de distribuciones Linux y se encuentran por tanto en todas ellas.

Podemos citar algunos ejemplos de los servicios que veremos como son el login, ftp, passwd, halt...

En este capítulo revisaremos todos los servicios Unix que vamos a integrar. Analizaremos la configuración individual de cada uno de ellos, en cuanto a la autenticación se refiere y definiremos nuestro ámbito de actuación.

Por último veremos una serie de scripts, propios de la distribución Red Hat, que también precisan de autenticación. Estos scripts permiten la configuración del sistema en todos los sentidos (red, impresoras, sonido, gráficos...). Algunos de ellos pueden llegar a ser muy útiles ya que automatizan ciertas partes y editan los ficheros de configuración necesarios de forma adecuada. Otros, por el contrario, no incluyen todas las opciones que deberían (o mejor dicho, todas las que existen), sólo incluyen las más usuales. Es preciso, por tanto, una configuración manual, editando los ficheros de configuración necesarios.

Incluso existe un script, authconfig, que permite configurar la autenticación. De hecho fue la primera opción que intentamos, y que se comprobó que no era viable debido a las razones que señalamos antes.

4.1 Consideraciones iniciales

El modo de proceder será el siguiente:

- Nos detendremos en cada servicio mostrando su descripción y comentando brevemente su funcionamiento.

- Analizaremos cada fichero de configuración, en cuanto a la autenticación se refiere.

- Por último propondremos otro fichero de configuración, que sustituirá al anterior y que hará uso de la información pertinente en el directorio OpenLDAP. Este nuevo fichero debe respetar al anterior en cuanto a su funcionalidad se refiere.

Una vez sustituido el fichero de configuración consideraremos al servicio como integrado, puesto que es la única modificación que hay que realizar a cada servicio en concreto. Evidentemente antes de asegurar que todo funciona hay que realizar la fase de pruebas. Esto se verá en un punto posterior.

Una última consideración que debemos tener en cuenta es que el contenido de los ficheros de configuración puede diferir entre distribuciones. Incluso algunos pueden no existir o estar en otra localización. Por ello debemos recalcar que estamos haciendo uso del sistema operativo Red Hat 9.0. Todo lo que estamos realizando se basa en este sistema. En otros no tiene porque funcionar y habría que hacer pequeñas modificaciones.

4.2 Login

Este servicio se encarga de comenzar una sesión en Linux. Normalmente se ejecuta al iniciarse la máquina aunque puede ser invocado de forma directa.

Requiere un nombre de usuario y su correspondiente contraseña. Si el nombre no existe o bien la contraseña para dicho usuario no es válida, login devuelve como salida "login incorrect" y vuelve al principio, a pedir el nombre.

Nótese que no se indica la causa del error evitando así dar información extra a aquellos usuarios no permitidos y que estén intentando acceder.

El fichero de configuración /etc/pam.d/login es el siguiente:

```
#%PAM-1.0
# Fichero /etc/pam.d/login

auth       required     pam_securetty.so
auth       required     pam_stack.so service=system-auth
auth       required     pam_nologin.so


account    required     pam_stack.so service=system-auth


password   required     pam_stack.so service=system-auth


session    required     pam_stack.so service=system-auth
session    optional     pam_console.so
```

La componente auth está compuesta por tres módulos apilados. El primero y el último son consideraciones extras de seguridad para el usuario root y que se pueden consultar en el apéndice.

En cuanto al segundo módulo (y en el resto de componentes) se utiliza pam_stack que a su vez llama a la componente auth del servicio

especificado, en este caso system-auth. Vamos a mostrar dicho fichero ya que lo vamos a referirnos a él en más ocasiones. Es el siguiente:

```
#%PAM-1.0
# Fichero /etc/pam.d/system-auth

auth        required      /lib/security/$ISA/pam_env.so
auth        sufficient    /lib/security/$ISA/pam_unix.so likeauth nullok
auth        required      /lib/security/$ISA/pam_deny.so

account     required      /lib/security/$ISA/pam_unix.so

password    required      /lib/security/$ISA/pam_cracklib.so retry=3 type=
password    sufficient    /lib/security/$ISA/pam_unix.so nullok use_authtok
md5
shadow
password    required      /lib/security/$ISA/pam_deny.so

session     required      /lib/security/$ISA/pam_limits.so
session     required      /lib/security/$ISA/pam_unix.so
```

La componente auth en cuestión utiliza los módulos pam_env que trata de establecer las variables de entorno, que aparecen en su fichero de configuración. Ocurre que ese fichero está vacío por defecto, bueno no exactamente vacío, pero si con todas las líneas comentadas. Así que éste va a cumplirse siempre y no denegará el acceso.

El siguiente, pam_unix, es el que realiza realmente la autenticación. El argumento nullok nos permite introducir contraseñas en blanco y el likeauth que realiza ciertas funciones de adaptación.

Por último pam_deny devuelve un código de error cuando la autenticación falla.

La componente account es simplemente una llamada al módulo pam_unix.

La componente passwd está compuesta también por el módulo pam_unix pero previamente se prueba la robustez de la contraseña gracias al módulo

crack_lib. Se establece contraseñas md5 y se utiliza ensombrecimiento de las mismas (shadow).

La última componente, session, tras establecer ciertos límites en los recursos del sistema (gracias a pam_limits) llama a pam_unix para guardar logs sobre el usuario que está accediendo.

Esto es en esencia el esqueleto de la parte común de los ficheros de configuración. El archivo system-auth será usado como base en la mayor parte de estos ficheros.

Volviendo al fichero que comentábamos al principio, /etc/pam.d/login, no hay más que ver la cantidad de veces que aparece system-auth para darnos cuenta de lo importante que es. Al comentar system-auth prácticamente hemos comentado /etc/pam.d/login. Y de hecho, esta fase de la integración consistirá en sustituir los módulos que hagan uso del fichero /etc/passwd por otros que utilicen el directorio OpenLDAP.

Tenemos sólo un módulo que accede a /etc/passwd que es pam_unix (también lo hace pam_pwdb, pero es equivalente en cualquier caso). Así que lo sustituiremos por pam_ldap y realizar de este modo la integración. Además vamos a realizar esta sustitución en el fichero system-auth, con lo cual no variaremos mucho el esqueleto que se usaba hasta entonces. Se debe crear un fichero nuevo, con otro nombre, ya que el fichero system-auth es autogenerado.

Existe una aplicación que permite configurar la autenticación de forma gráfica. Se llama authconfig y cuando se utiliza guarda la configuración en system-auth. No vamos a utilizar dicha aplicación ya que es preferible indicar a mano el esquema propuesto. Por ello vamos a usar otro fichero, que vamos a llamar integrado, y que a continuación mostramos.

```
#%PAM-1.0
# Fichero apilable /etc/pam.d/integrado

auth        required      /lib/security/$ISA/pam_env.so
auth        sufficient    /lib/security/$ISA/pam_ldap.so
auth        required      /lib/security/$ISA/pam_deny.so

account     required      /lib/security/$ISA/pam_ldap.so

password    required      /lib/security/$ISA/pam_cracklib.so retry=3 type=
password    sufficient    /lib/security/$ISA/pam_ldap.so
password    required      /lib/security/$ISA/pam_deny.so

session     required      /lib/security/$ISA/pam_limits.so
session     required      /lib/security/$ISA/pam_ldap.so
```

Ya sólo nos queda utilizar este fichero integrado en el fichero de configuración de login para que las diversas partes de la autenticación se realicen contra el directorio OpenLDAP que tenemos preparado.

Mostramos a continuación dicho fichero /etc/pam.d/login. Vemos como un cambio aparentemente pequeño ha acarreado un cambio cualitativamente grande.

```
#%PAM-1.0
# Fichero /etc/pam.d/login ya integrado

auth       required     pam_securetty.so
auth       required     pam_stack.so service=integrado
auth       required     pam_nologin.so


account    required     pam_stack.so service=integrado


password   required     pam_stack.so service=integrado


session    required     pam_stack.so service=integrado
session    optional     pam_console.so
```

Este será el modo de proceder en el resto de servicios que se proporcionan en la distribución y que necesitan autenticación para su funcionamiento.

4.3 Su

Este servicio cambia la identificación efectiva de usuario y de grupo del usuario, valga la redundancia, que lo invoca. Su uso es el siguiente:

```
su  [opciones] [usuario]
```

Cuando no indicamos usuario, se sobreentiende root. Normalmente tras ejecutar dicho comando pide la clave para este nuevo usuario, excepto si modificamos el módulo PAM de este servicio.

En realidad lo que hace su es ejecutar comandos en nombre del usuario especificado. Lo que ocurre es que este comando suele ser una shell interactiva. Cada usuario usa una shell por defecto, que aparece en el fichero /etc/passwd y es la que se toma para este caso.

Por defecto, su no cambia el directorio en el que se encuentra. Establece las variables de entorno SHELL y HOME con los valores que encuentra en el fichero /etc/passwd, como acabamos de comentar y si el usuario no es el root, establece USER y LOGNAME a este usuario.

Las opciones típicas son:

`-l` (o simplemente -) que convierte a la shell en una shell de login, esto es, se borran todas las variables de entorno excepto TERM, HOME y SHELL, que quedan establecidas como acabamos de decir; y USER y LOGNAME que les ocurren lo mismo incluso para el usuario root.

`-c comando` permite ejecutar el comando que se indica como si fuera el usuario especificado.

Pasemos ahora a comentar el fichero PAM de este servicio.

```
#%PAM-1.0
# Fichero /etc/pam.d/su

auth       sufficient   /lib/security/pam_rootok.so
auth       required     /lib/security/pam_stack.so service=system-auth

account    required     /lib/security/pam_stack.so service=system-auth

password   required     /lib/security/pam_stack.so service=system-auth

session    required     /lib/security/pam_stack.so service=system-auth
session    optional     /lib/security/pam_xauth.so
```

Vemos que responde al esquema típico de autenticación que hemos visto hasta ahora. Además podemos comprobar el comportamiento que hemos visto en la descripción. El primer módulo, pam_rootok, permite al root no tener que introducir el password cuando invoque a su.

El resto de módulos llaman a system-auth y el último a pam_xauth.

Pam_auth está diseñado para intercambiar claves xauth, sin las cuales el nuevo usuario no sería capaz de acceder a las X del antiguo.

El fichero de configuración de su que sustituirá al que viene en la distribución será el siguiente.

```
#%PAM-1.0
# Fichero /etc/pam.d/su integrado

auth       sufficient   /lib/security/pam_rootok.so
auth       required     /lib/security/pam_stack.so service=integrado

account    required     /lib/security/pam_stack.so service=integrado

password   required     /lib/security/pam_stack.so service=integrado

session    required     /lib/security/pam_stack.so service=integrado
session    optional     /lib/security/pam_xauth.so
```

4.4 Sudo

Sudo permite a un usuario ejecutar un comando como si fuera root u otro usuario, según especifique el fichero /etc/sudoers. Las identificaciones de usuario y de grupo (uid y gid) reales y efectivas se establecen de forma que coincidan con las que aparece en el fichero /etc/passwd para el usuario objetivo.

Por defecto sudo require que los usuarios se autentiquen mediante el uso de sus respectivos passwords. Una vez autenticados, una marca de tiempo se actualiza y entonces el usuario puede usar sudo sin tener que introducir el password cada vez. Esto es válido por un corto periodo de tiempo, unos 5 minutos, sino se sobreescribe en el fichero /etc/sudoers.

Sudo determina quien está autorizado consultando /etc/sudoers. Podemos actualizar la marca de tiempo sin tener que ejecutar ningún comando, usando para ello sudo -v. El password “caducará” si no hemos introducido durante el periodo en el que es válido.

Si un usuario que no se encuentra listado en el fichero /etc/sudoers intenta ejecutar un comando mediante sudo, se envía un correo a la autoridad apropiada. Por defecto es el root, pero se puede cambiar en el fichero de configuración que tantas veces hemos nombrado.

No se notificará un uso de sudo si se utiliza sudo -v ó sudo -l. El primero de ellos ya lo hemos comentado. El segundo nos permite conocer los comandos que podemos ejecutar en la máquina en la que nos encontramos.

A continuación mostramos el fichero de configuración de la autenticación de sudo. Utiliza el esqueleto que ya hemos comentado con anterioridad.

```
#%PAM-1.0
# Fichero /etc/pam.d/sudo

auth       required     pam_stack.so service=system-auth

account    required     pam_stack.so service=system-auth

password   required     pam_stack.so service=system-auth

session    required     pam_stack.so service=system-auth
```

Por último mostramos el fichero que sustituirá al anterior.

```
#%PAM-1.0
Fichero /etc/pam.d/sudo integrado

auth       required     pam_stack.so service=integrado

account    required     pam_stack.so service=integrado

password   required     pam_stack.so service=integrado

session    required     pam_stack.so service=integrado
```

Debemos hacer un comentario a este servicio, y es que, realmente, no está integrado. Su funcionamiento depende del fichero local /etc/sudoers. En este fichero de cada máquina debe especificarse los usuarios que tienen permitido ejecutar sudo.

La integración de sudo es posible (aunque no trivial). Se debe recompilar el paquete sudo con diversas opciones activadas, entre ellas, el soporte LDAP. Hay que añadir un nuevo esquema al servidor y modificar los archivos de configuración (ldap.conf, slapd.conf, nsswitch.conf ...).

Si pensamos en el uso que va a tener la herramienta sudo parece que no merece la pena su integración.

En el entorno para el que hemos diseñado el sistema, un centro de cálculo, todas las aplicaciones que puede usar el usuario están controladas. Los permisos para ejecutar ciertas aplicaciones se pueden conceder por grupo y así limitarlos.

De todas formas se puede dar el caso de que sea necesario y por ello, lo propondremos como una posible ampliación.

4.5 Passwd

Este servicio se utiliza para actualizar el password de un usuario. Su funcionamiento es básico. Al ejecutarse pide la contraseña válida hasta entonces y la nueva, que va a sustituir a ésta. Vuelve a pedir la nueva a modo de confirmación y si todo ha ido bien se actualiza convenientemente.

Debido a los motivos ya comentados de vulnerabilidades de los passwords y de lo importante que es elegir un password robusto, disponemos del módulo pam_cracklib. Incluso se ha añadido al esqueleto de autenticación que estamos manejando (se encuentra en system-auth), de forma que se utilice en todos los servicios que requieran actualización de contraseñas.

Al igual que hicimos con los módulos anteriores, presentamos aquí el fichero de configuración.

```
#%PAM-1.0
# Fichero /etc/pam.d/passwd

auth       required     pam_stack.so service=system-auth

account    required     pam_stack.so service=system-auth

password   required     pam_stack.so service=system-auth
```

A continuación mostramos el fichero de configuración para el mismo servicio haciendo uso del directorio OpenLDAP, de igual forma que venimos haciendo hasta ahora.

```
#%PAM-1.0
# Fichero /etc/pam.d/passwd
auth       required     pam_stack.so service=integrado

account    required     pam_stack.so service=integrado

password   required     pam_stack.so service=integrado
```

4.6 Xserver

El entorno gráfico en las máquinas Unix viene proporcionado por el sistema X windows (conocido normalmente como X, o las X). Para Red Hat tenemos XFree86 que es una implementación de X.

El sistema X window presenta una arquitectura cliente – servidor. El servidor de X (Xserver) escucha conexiones de las aplicaciones clientes a través de la red o de la interfaz local. El servidor gestiona la comunicación con el hardware, como puede ser la tarjeta gráfica, un monitor, un ratón, etc.

Normalmente el servidor de las X se inicia mediante xdm u otro gestor de ventanas, aunque también puede arrancarse directamente por un usuario. Esto último está reservado para pruebas y no es recomendable para un uso habitual. Así que el fichero de configuración será el siguiente y no requiere que lo modifiquemos en nuestro sistema con autenticación integrada.

```
#%PAM-1.0
# Fichero /etc/pam.d/xserver

auth       sufficient   /lib/security/pam_rootok.so
auth       required     /lib/security/pam_console.so

account    required     /lib/security/pam_permit.so
```

4.7 Xdm, gdm

Xdm gestiona una colección de displays X (gdm gestiona concretamente el entorno Gnome), que pueden estar en la máquina local o en servidores remotos. Provee de servicios similares a los proporciona init, getty y login en los terminales de caracteres: pide un nombre de usuario y contraseña, autentica a dicho usuario y comienza una 'sesión'.

Una sesión se define por el tiempo de vida de un proceso en particular. En el ámbito del terminal tradicional basado en texto, la sesión sería la shell de login del usuario. En el contexto de xdm sería un gestor de sesión arbitrario. Esto es así ya que en un entorno de ventanas, un proceso de shell de login no tiene asociado ninguna interfaz de tipo terminal con la que conectarse. Cuando un gestor de sesión no se encuentra disponible, se toma como gestor de sesión a un gestor de ventanas o a un emulador de terminal. Entonces la terminación de este proceso provoca la terminación de la sesión.

En cuanto al fichero de configuración PAM, podemos comprobar las similitudes señaladas respecto a login. Es exactamente el mismo salvo que en este caso no se incluye la línea referente a la restricción del acceso directo al sistema como root. Recordemos que esto se lograba mediante el módulo pam_securetty.

A continuación mostramos el archivo que estamos comentando.

```
#%PAM-1.0
# Fichero /etc/pam.d/gdm

auth       required     /lib/security/pam_stack.so service=system-auth
auth       required     /lib/security/pam_nologin.so


account    required     /lib/security/pam_stack.so service=system-auth


password   required     /lib/security/pam_stack.so service=system-auth
```

```
session    required     /lib/security/pam_stack.so service=system-auth
session    optional     /lib/security/pam_console.so
```

Y de nuevo presentamos el archivo que sustituirá al anterior en nuestro sistema integrado.

```
#%PAM-1.0
# Fichero /etc/pam.d/gdm integrado

auth       required     /lib/security/pam_stack.so service=integrado
auth       required     /lib/security/pam_nologin.so


account    required     /lib/security/pam_stack.so service=integrado


password   required     /lib/security/pam_stack.so service=integrado


session    required     /lib/security/pam_stack.so service=integrado
session    optional     /lib/security/pam_console.so
```

4.8 Other

El fichero que vamos a presentar a continuación correspondería a un servicio que utilizara PAM como esquema de autenticación, pero que no dispone de archivo de configuración especificado.

Ante el desconocimiento del servicio en cuestión que va a utilizar dicho archivo, no queda más remedio que denegar cualquier intento de autenticación en el sistema. De este modo se previene cualquier acceso dañino o no autorizado que pueda hacer uso de dicho fichero.

El fichero comentado es el siguiente.

```
#%PAM-1.0
# Fichero /etc/pam.d/other

auth     required       /lib/security/pam_deny.so

account  required       /lib/security/pam_deny.so

password required       /lib/security/pam_deny.so

session  required       /lib/security/pam_deny.so
```

Vemos como se ha codificado el comportamiento requerido; denegar el acceso por defecto. No existe autenticación, aunque más bien podríamos decir que siempre va a ser negativa. Tampoco necesitamos otro archivo de configuración, éste se adapta a la situación prevista y funciona adecuadamente.

4.9 Halt, poweroff, reboot

Los comandos halt y poweroff se encargan de parar el sistema y reboot de reiniciarlo, esto es, pasar al nivel de ejecución 0 y 6, respectivamente.

En realidad no son más que una llamada al comando shutdown especificando la opción de parar (-h) o la de reiniciar (-r).

El fichero de configuración en los tres casos es idéntico por lo que lo mostramos sólo una vez.

```
#%PAM-1.0
# Fichero /etc/pam.d/halt, /etc/pam.d/poweroff, /etc/pam.d/reboot

auth       sufficient   pam_rootok.so
auth       required     pam_console.so
#auth       required    pam_stack.so service=system-auth


account    required     pam_permit.so
```

Vemos que el usuario root no tiene que introducir clave alguna y que el resto de usuarios no pueden apagar (o reiniciar) el sistema salvo que fueran ellos los que lo iniciaron. Esto se realiza gracias a pam_console.

Además existe otra opción, que consistiría en tener en cuenta la autenticación que estamos usando para el resto de servicios. Esta opción queda reflejada en la línea comentada y se realizaría, como ya hemos visto otras tantas veces, gracias a pam_stack.

El comportamiento de la opción usada, es decir, dejar al fichero tal como está, corresponde a nuestra política. Es lógico que tanto el root como el usuario que encendió el sistema tengan derecho a apagar el equipo. Hablamos de equipos de trabajos comunes que utilizarán tanto alumnos como profesores por lo que parece que este es el comportamiento más acertado.

Tal como está el fichero de configuración, podemos apreciar como no se hace uso del directorio OpenLDAP, así que no vamos a tener que sustituirlo y de hecho se utilizará el mismo.

4.10 Ssh

Mediante ssh (un cliente de ssh, concretamente) podemos acceder a máquinas remotas y ejecutar comandos en ellas. Pretende reemplazar a los inseguros rlogin y rsh y proveer de una comunicación segura entre dos máquinas sobre una red no segura.

El cliente necesita conocer la dirección de la máquina a la que se va a conectar, o el nombre, si las tenemos en dns.

El usuario debe probar su identidad a la máquina remota usando uno de los diferentes métodos que existen, dependiendo de la versión del protocolo usado. Existen dos versiones, la 1 y la 2, que se diferencian básicamente en que la última soporta más mecanismos de encriptación del tráfico, además de proveer de uno bastante potente para proteger la integridad de la conexión.

Los métodos de autenticación que se utilizan son los siguientes:

Primero se realiza una autenticación por máquina. Si la máquina desde la que el usuario accede se encuentra listada en el fichero /etc/hosts.equiv de la máquina remota, el usuario tiene inmediatamente permiso para acceder. De igual forma si existe cuenta del usuario en la máquina remota, y se encuentra en su directorio 'home' el fichero .shosts o .rhosts puede también acceder. En este caso es necesario que exista una línea en uno de estos dos ficheros que incluya en nombre de usuario y el de la máquina.

Usar sólo este método es desaconsejable, por la inseguridad que conlleva.

El segundo método consiste en usar autenticación de clave pública (es muy típico usar RSA). El tercero sería pedir la contraseña de usuario, tal como se hace clásicamente.

Pues bien, la autenticación se realiza de forma que se intenta el primer método, si falla se intenta el segundo, y si falla éste se intenta el último. Si falla el último se deniega el acceso.

Gracias al último método podemos integrar la autenticación en el directorio. Para ello modificamos el fichero PAM, como hemos ido haciendo hasta ahora.

El fichero de configuración que usa ssh por defecto es éste:

```
#%PAM-1.0
# Fichero /etc/pam.d/sshd

auth       required     pam_stack.so service=system-auth
auth       required     pam_nologin.so


account    required     pam_stack.so service=system-auth


password   required     pam_stack.so service=system-auth


session    required     pam_stack.so service=system-auth
session    required     pam_limits.so
session    optional     pam_console.so
```

El fichero que lo sustituirá será el siguiente.

```
#%PAM-1.0
# Fichero /etc/pam.d/sshd

auth       required     pam_stack.so service=integrado
auth       required     pam_nologin.so


account    required     pam_stack.so service=integrado


password   required     pam_stack.so service=integrado


session    required     pam_stack.so service=integrado
session    required     pam_limits.so
session    optional     pam_console.so
```

4.11 Ftp (vsftpd)

Ftp es el protocolo que permite la transferencia de archivos en una red TCP/IP. Se basa en el modelo cliente – servidor. Cuando un usuario desea intercambiar (ya sea subir o descargar) archivos con una máquina de su red, ejecuta el cliente indicando la máquina en cuestión. Entonces el servidor debe comprobar las credenciales: autenticar al usuario y determinar si tiene permiso para acceder al máquina por ftp.

La autenticación del usuario se realiza contra /etc/passwd como siempre, y de igual forma vamos a cambiarlo. Además tenemos la opción de no permitir que algunos usuarios, que aunque están registrados en nuestro sistema, no tienen permiso para hacer ftp. Para ello tenemos un archivo donde indicamos los usuarios permitidos, o los no permitidos, según nuestra política. En nuestro sistema este archivo es /etc/vsftpd.users y lo gestiona el módulo pam_listfile.

El servicio de ftp se proporciona en nuestro sistema mediante el demonio vsftpd (very secure file transfer protocol daemon). También se incluye el servidor proftpd pero es el primero el que se instala por defecto. Además cubre todas nuestras necesidades, por lo que va a ser el servidor elegido.

El fichero de configuración del servidor es /etc/vsftpd.conf por defecto, aunque se puede especificar otro al arrancar el servidor. De hecho es esto lo que ocurre. Se utiliza el fichero /etc/vsftd/vsftpd.conf y en él se pueden indicar distintas opciones. Las que no se indican conservan el valor que traen por defecto, algo normal en los archivos de configuración.

El fichero de configuración que vamos a utilizar va a indicar que el servidor se ejecuta como demonio, es decir, sin ser gestionado por ningún superservidor, como puede ser xinetd. También añadiremos la opción user_list que permitirá tener una lista de usuarios no permitidos. Ya hemos

hablado de un fichero que hacía esto mismo. La diferencia es que a los usuarios en esta lista no se les pedirá ni siquiera la contraseña y directamente se rechazará. El objetivo de esto es evitar el envío de información sensible en claro, aunque, como vamos a ver a continuación, se va a intentar resolver el problema de raiz.

En cuanto a los clientes podemos usar el típico comando ftp, pero debido a la falta de seguridad con la que maneja tanto la información de usuario como los datos que transmite, vamos a desecharlo. Usaremos entonces sftp, que básicamente realiza la misma función que ftp, pero lo hace sobre una conexión de transporte encriptada ssh.

Sftp no es un cliente ftp, sino que es un programa que se incluye dentro del paquete openSSH. Este paquete tiene un subsistema servidor, llamado sftp-server, que es el que se encarga de dar el servicio al cual sftp accede. La autenticación que usa sftp queda recogida dentro de ssh, por lo que basta su configuración para que ambos autentiquen de forma deseada.

En cuanto a ftp habrá que decidir que hacer con él. Podemos eliminarlo, dejarlo tal como está o mostrar un mensaje de advertencia cuando se invoque. Por el momento vamos a dejarlo disponible, remarcando que no debe ser usado.

Se muestra a continuación el fichero de configuración PAM.

```
#%PAM-1.0
# Fichero /etc/pam.d/vsftpd

auth       required     pam_listfile.so item=user sense=deny
                        file=/etc/vsftpd.ftpusers onerr=succeed
auth       required     pam_stack.so service=system-auth
auth       required     pam_shells.so


account    required     pam_stack.so service=system-auth


session    required     pam_stack.so service=system-auth
```

Vamos a sustituirlo por el siguiente.

```
#%PAM-1.0
#Fichero /etc/pam.d/vsftpd integrado

auth       required     pam_listfile.so item=user sense=deny
                        file=/etc/vsftpd.ftpusers onerr=succeed
auth       required     pam_stack.so service=integrado
auth       required     pam_shells.so


account    required     pam_stack.so service=integrado


session    required     pam_stack.so service=integrado
```

4.12 Xscreensaver

Este programa espera hasta que el ratón y el teclado lleven cierto tiempo parados, entonces ejecuta una demostración gráfica escogida al azar. La demostración desaparece y el escritorio vuelve al estado normal en cuanto existe alguna actividad por parte del ratón o del teclado.

Este programa nos permite además bloquear el terminal para prevenir que otros lo usen cuando el usuario no está. Cuando éste vuelve y quiere usar de nuevo su terminal, debe introducir su clave para volver a tenerlo disponible. La autenticación, como el resto de servicios, la gestiona el fichero de configuración PAM. Vamos a mostrarlo aquí.

```
#%PAM-1.0
#Fichero /etc/pam.d/xscreensaver

# Red Hat says this is right for them, as of 7.3:
auth       required     pam_stack.so service=system-auth

# This is what we were using before:
# auth       required   pam_pwdb.so shadow nullok
```

El fichero en sí es un tanto particular. Incluye varios comentarios indicando lo que utilizaban anteriormente. Usaban el módulo pam_pwdb, que posee una interfaz genérica para las bases de datos de passwords, y ahora usan pam_unix (dentro de system-auth), que está más especializado y extendido.

El fichero que vamos a usar es simplemente el siguiente:

```
#%PAM-1.0
#Fichero  /etc/pam.d/xscreensaver integrado

auth       required     pam_stack.so service=integrado
```

4.13 Cups

Cups es el sistema de impresión de Unix. Se basa en el protocolo IPP (internet printing protocol) para la gestión de trabajos y de colas. Permite configurar impresoras tanto locales como de red bajo la misma interfaz. Para ello necesita autenticación y usa el fichero de configuración PAM siguiente.

```
#%PAM-1.0
#Fichero /etc/pam.d/cups

auth    required        pam_stack.so service=system-auth


account required        pam_stack.so service=system-auth
```

Para integrar la autenticación vamos a sustituir el anterior fichero por este otro, que hace uso del directorio OpenLDAP.

```
#%PAM-1.0
#Fichero /etc/pam.d/cups integrado

auth    required        pam_stack.so service=integrado


account required        pam_stack.so service=integrado
```

Por defecto cups sólo permite realizar labores administrativas, como pueden ser añadir impresoras o borrar trabajos de la cola, al root. Será este entonces el usuario, con su correspondiente clave, el que tengamos que usar para realizar dichas tareas.

4.14 Hwbrowser

Hwbrowser es una herramienta que muestra todo el hardware detectado en el sistema. Se puede usar interactivamente para navegar por los detalles de la configuración hardware del sistema. No tiene demasiado interés.

Mostramos a continuación el fichero de configuración PAM.

```
#%PAM-1.0
#Fichero /etc/pam.d/hwbrowser

auth       sufficient   pam_rootok.so
auth       sufficient   pam_timestamp.so
auth       required     pam_stack.so service=system-auth


session    required     pam_permit.so
session    optional     pam_xauth.so


account    required     pam_permit.so
```

Y el fichero que lo sustituye.

```
#%PAM-1.0
# Fichero /etc/pam.d/hwbrowser integrado

auth       sufficient   pam_rootok.so
auth       sufficient   pam_timestamp.so
auth       required     pam_stack.so service=integrado


session    required     pam_permit.so
session    optional     pam_xauth.so


account    required     pam_permit.so
```

4.15 Dateconfig

Este programa permite establecer la franja horaria en la que se encuentra la máquina, así como la fecha y hora exacta. No tiene mayor interés.

Su fichero de configuración PAM es el siguiente.

```
#%PAM-1.0
# Fichero /etc/pam.d/dateconfig

auth       sufficient   pam_rootok.so
auth       sufficient   pam_timestamp.so
auth       required     pam_stack.so service=system-auth


session    required     pam_permit.so
session    optional     pam_xauth.so
session    optional     pam_timestamp.so


account    required     pam_permit.so
```

Se sustituirá por este:

```
#%PAM-1.0
# Fichero /etc/pam.d/dateconfig integrado

auth       sufficient   pam_rootok.so
auth       sufficient   pam_timestamp.so
auth       required     pam_stack.so service=integrado


session    required     pam_permit.so
session    optional     pam_xauth.so
session    optional     pam_timestamp.so


account    required     pam_permit.so
```

4.16 Scripts de configuración propios de Red Hat

A continuación vamos a enumerar los scripts que permiten configurar ciertos aspectos del sistema. No entraremos en detalles, sólo se trata de una interfaz gráfica. Tan sólo comentaremos sus funciones. Ni siquiera vamos a mostrar los ficheros de configuración PAM, ya que son similares a los usados en los servicios que hemos visto anteriormente. Hemos realizado simplemente el cambio de system-auth por integrado, del módulo pam_stack. No tienen mayor interés.

Se muestran ordenados alfabéticamente.

4.16.1 Redhat-cdinstall-helper

Esta aplicación se encarga de lanzar el instalador de la distribución, cuando se introduce un cdrom de instalación.

4.16.2 Redhat-config-date

Esta aplicación permite configurar la fecha y la hora, así como la franja horaria en la que se encuentra el equipo en cuestión. Es equivalente al comando dateconfig.

4.16.3 Redhat-config-keyboard

Mediante esta utilidad podemos cambiar el idioma, que usa por defecto el teclado del sistema.

4.16.4 Redhat-config-language

Permite cambiar el idioma por defecto que se usa en el sistema, entre aquellos que han sido previamente instalados.

4.16.5 Redhat-config-mouse

Esta utilidad permite cambiar la configuración que trae por defecto el ratón del sistema.

4.16.6 Redhat-config-network

Esta es una herramienta tipo asistente que permite configurar el sistema, para que se conecte a una red. Soporta Ethernet, Wireless, TokenRing, ADSL, RDSI y PPP.

4.16.7 Redhat-config-network-druid

Similar al anterior.

4.16.8 Redhat-config-packages

Comprueba el estado de los paquetes del sistema y permite instalar los paquetes correspondientes a una aplicación concreta.

4.16.9 Redhat-config-printer

Mediante esta interfaz podemos configurar las impresoras conectadas.

4.16.10 Redhat-config-proc

Esta utilidad permite cambiar parámetros del kernel.

4.16.11 Redhat-config-rootpassword

Permite cambiar la contraseña de root de forma gráfica.

4.16.12 Redhat-config-securitylevel

Permite configurar el nivel de seguridad que posee el sistema. Se han definido tres niveles, bajo, medio y alto. Estar en un nivel o en otro lo determina el número de puertos que dejamos abiertos para ofrecer servicios, tales como ftp o ssh.

4.16.13 Redhat-config-services

Esta es una herramienta gráfica que permite habilitar (o deshabilitar) servicios, de forma que se inicien cuando lo haga la máquina. También permite iniciarlos, pararlos o reiniciarlos.

4.16.14 Redhat-config-soundcard

Este script es una pequeña interfaz gráfica que permite detectar y configurar la tarjeta de sonido que posee el sistema.

4.16.15 Redhat-config-time

Similar a redhat-config-date.

4.16.16 Redhat-config-users

Utilidad gráfica que permite administrar usuarios y grupos.

4.16.17 Redhat-config-xfree86

Interfaz gráfica que permite al usuario configurar su servidor XFree86.

4.16.18 Redhat-install-packages

Esta herramienta se usa para instalar paquetes aislados. Toma como argumento el paquete en cuestión y resuelve dependencias, si se necesita, de la distribución principal.

4.16.19 Redhat-logviewer

Interfaz gráfica para la consulta de archivos de logs del sistema.

4.16.20 Setup

Esta es una herramienta que integra prácticamente toda la configuración del sistema. Cuando se ejecuta muestra un menú que permite seleccionar el aspecto del sistema que se desea configurar. La selección de uno de ellos no es más que una llamada a uno de los programas que acabamos de ver. Así por ejemplo si seleccionamos 'mouse configuration', setup llamará a la utilidad redhat-config-mouse, que ya hemos comentado.