CAPITULO III

CAPITULO III

3. SEGURIDAD EN EL SERVIDOR

3.1 SISTEMA OPERATIVO LINUX

3.1.1 ¿Qué es Linux?

Linux es un sistema operativo gratuito de 32 o 64 bits, similar a Unix, con código abierto optimizado para Internet, el mismo que puede funcionar en distintos tipos de hardware incluyendo procesadores Intel (x86) o RISC. [A 3.01]

El sistema operativo Linux está desarrollado bajo Licencia General Pública GNU (conocida además como GNU GPL), lo cual significa que se pueden usar las herramientas que provee el sistema operativo para desarrollar y vender aplicaciones de Linux sin pagar derechos de comercialización. Si embargo, si se realiza algún cambio en las bibliotecas GPL, también debe realizárselo de forma gratuita en la GPL de turno.

3.1.2 Algunas buenas razones para usar Linux

- No hay derechos de autor9 o pagos de licencias para usar Linux, y el código fuente puede ser modificado para adecuarse a las necesidades de los usuarios.

- Debido a que viene con el código fuente del Kernel, este es bastante portable. Linux corre en más CPUs y platafomras que cualquier otro sistema operativo.

- Las recientes orientaciones de la industria del software y el hardware es empujar a los consumidores a unirse a computadores super rápidos con una gran cantidad de memoria y grandes capacidades de almacenamiento del disco duro. Linux no se ve afectado por estas nuevas orientaciones de la industria este se ejecuta en cualquier clase de computador incluso en un 486 con limitaciones de memoria RAM.

- Linux es un verdadero sistema operativo multi-tarea. Este utiliza sofisticados procesos para el control y administración de la memoria.,esto significa que si un programa falla simplemente se mata este proceso y el resto de programas siguen ejecutándose normalmente.

- Otro beneficio es que Linux es prácticamente inmune a la mayoría de virus existentes que atacan a otros sistemas operativos.

- Cada distribución de Linux viene con más que 12.000 páginas de documentación, además se cuenta con una gran variedad de Web Sites que disponen de información y existen muchas listas de distribución de correo a las cuales se puede acudir en busca de ayuda.

- Una gran cantidad de Linux está escrita en C, el cual es un lenguaje de programación muy común y fácil de aprender.

- Linux admite sesiones multiusuario, lo que implica que varios usuarios pueden acceder simultáneamente y durante estas sesiones, pueden realizar varias tareas.

- Linux cuenta con un gran número de funciones de red, las cuales están asociadas con la mayoría de protocolos y servicios de red existentes.

3.1.3 Linux como servidor de Internet e Intranet

Linux es una excelente plataforma servidora de una Intranet o de Internet, ya que ofrece una óptima potencia a las redes y proporciona clientes y servidores para todos los protocolos esenciales entre ellos, se incluyen: [A 3.02]

FTP.- File Transfer Protocol, Protocolo de Transferencia de Archivos

HTTP.- Hypertext Transfer Protocol, Protocolo de Transferencia de Hipertexto

POP.- Post Office Protocol, Protocolo de Oficina de Correos

SMTP.- Simple Mail Transfer Protocol, Protocolo Simple de Transferencia de Correo

TCP.- Transmission Control Protocol, Protocolo de Control de Transmisión

IP.- Internet Protocol, Protocolo Internet

NNTP.- Network News Transfer Protocol, Protocolo de Transferencia de Noticias en Red

Linux es con toda seguridad el sistema operativo más optimizado para redes, existente en la actualidad. Incluso llega a admitir protocolos de red de otros sistemas operativos, incluyendo

Microsoft Windows y MacOS, de este modo, los servidores Linux se integran perfectamente en cualquier entorno heterogéneo.

3.2 SEGURIDAD FISICA

Un aspecto muy importante de la seguridad de una red y que se suele pasar por alto es que los servidores son más vulnerables a los ataque físicos que a los lógicos, los culpables más frecuentes son:

- Usuarios locales malintencionados
- Vándalos
- Ladrones

De hecho, no sólo es más probable que un servidor sufra un ataque físico, antes que con una utilidad de spoofing, sino que cuando un evento de este tipo sucede, los efectos posteriores pueden ser mucho más devastadores. Si se altera un sistema de forma remota, siempre se pude reinciar, reinstalar o reconfigurar, pero si ha sido dañado o puesto en peligro físicamente, el problema puede ser más serio.

Estos son los motivos por los que la seguridad física debe ser el primer objetivo, pese a que muchas medidas de seguridad física parecen obvias sistemáticamente los usuarios no las aplican.

En cuanto a la seguridad física se deben abordar los siguientes temas:

3.2.1 Ubicación del servidor y acceso físico a él

En cuanto a la ubicación de Servidor y acceso físico al mismo, existen dos aspectos importantes los cuales son: el lugar en que se encuentra ubicado el servidor y las personas que tienen acceso físico al mismo.

Los especialistas en seguridad llevan mucho tiempo sosteniendo que si usuarios malintencionados tienen acceso físico, los controles de seguridad son inútiles y dicha afirmación es totalmente cierta. Salvo raras excepciones, casi todos los sistemas de computación son vulnerables a ataques in situ.

Desde luego, ataque puede significar muchas cosas en este contexto, por ejemplo, imaginemos que algún usuario malintencionado se ha quedado en el servidor por 10 segundos, es muy probable que éste sufra daños importantes en ese intervalo de tiempo, el usuario podría realizar un rudimentario ataque de negación de servicio desconectando cables, desconectando hardware de red o reiniciando el servidor.

La mayor preocupación deben ser los usuarios locales autorizados, aquellos que tienen al menos autorización limitada para acceder al sistema, se estima que el 80% de las intrusiones provienen del personal interno, el motivo es que este personal tiene acceso a información que los agresores remotos a menudo no pueden obtener.

Pero ésta no es la única ventaja que tiene el personal interno, la confianza es otra. En muchas empresas, los empleados de confianza deambulan libremente sin temor a que les hagan preguntas, después de todo, se supone que están en su sitio y a nadie se le ocurre cuestionar su presencia, a menos que entren en un área restringida, así que ¿cómo se puede proteger un sistema frente a los enemigos internos?.

Las agencias gubernamentales y los ISP tienen una amplia experiencia en esta materia y merece la pena seguir su ejemplo. Si el sistema es para toda una empresa, se puede planificar un **centro de operaciones de red (NOC).**

3.2.1.1 El centro de operaciones de red (NOC Network Operation Center)

Un NOC es un área restringida en la que se encuentran los servidores, estos suelen estar asegurados con pernos, fijados a bastidores o asegurados de alguna manera, junto con el hardware de red esencial. [A 3.03]

Idealmente, un NOC debería ser una oficina independiente a la que tuviesen acceso muy pocas personas, aquellas que estén autorizadas deberían tener claves, un buen método es el uso de tarjetas de acceso que incluso restrinja el acceso de los usuarios autorizados a ciertas horas del día. Por último, merece la pena llevar un registro escrito de acceso y ordenar que incluso el personal autorizado firme al entrar y al salir.

Además debe asegurarse que el NOC cumple los siguientes requisitos:

- Debe encontrarse dentro de otro espacio de la oficina y alejado del público; es preferible que no se encuentre en la planta baja.

- La sala y los pasillos que conducen a ella deben ser totalmente opacos; sin puertas de cristal.

- Las puertas de acceso deben tener un blindaje que incluya el cerco de la puerta, esto evita que los intrusos fuercen la cerradura.

- Si se emplea vigilancia (circuito cerrado de TV), se debe dirigir la señal desde la cámara a un VCR remoto, esto garantiza que aunque los ladrones dañen el equipo y se lleven la cinta, se seguirá teniendo pruebas.

- Se debe mantener todos los dispositivos de almacenamiento (backups) en un lugar seguro, o aún mejor, en un lugar distinto del NOC.

Además, hay que promulgar estrictas reglas que prohiban al usuario medio entrar al NOC, y por supuesto se deben de establecer los medios adecuados para que todos los empleados conozcan estas normas y las ejecuten de la manera más adecuada.

3.2.2 Contraseña de BIOS y consola

La mayoría de las arquitecturas (como X86, PPC, Sparc) utilizan contraseñas de BIOS-PROM, contraseñas de consola o de ambos tipo. Los fabricantes de hardware incluyen estos sistemas de contraseñas como una capa extra de seguridad, un obstáculo para disuadir a los usuarios esporádicos de fisgonear.

Las contraseñas de la BIOS o de la PROM evitan que los usuarios malintencionados accedan a la configuración del sistema, mientras que las contraseñas de consola suelen proteger los perfiles de usuario de la estación de trabajo. En cualquier caso, estos sistemas son, al menos, parcialmente efectivos y es conveniente usarlos siempre que sea posible, pero se debe cambiar las contraseñas que vienen por default debido a que son muy conocidas y no estarían brindando la seguridad que se busca.

Hay que asegurarse de que la contraseña no coincida con otras que se utilicen en la red, lo que garantiza que si rompen la contraseña de la BIOS o de la consola, las aplicaciones o máquinas restantes no estarán expuestas a ningún ataque.

Sin embargo, lo más recomendable es no fiarse de las contraseñas de la BIOS y de la consola como una línea de defensa, ya que tienen defectos inherentes, uno de ellos es que los agresores pueden anular las contraseñas de la BIOS con sólo provocar un cortocircuito en la batería de la CMOS; en otros casos, ni siquiera necesitan hacerlo, ya que el fabricante de la placa base incluye un jumper que colocado del modo adecuado borrará la CMOS. Más aún los agresores van armados frecuentemente con barrenadores de BIOS (programas que borran los ajustes de la BIOS) o con utilidades de captura de contraseña de BIOS.

3.2.3 Control biométrico de acceso

Una aproximación más futurista a la seguridad física del hardware consiste en el uso de dispositivos de acceso biométricos. Estas herramientas autentican a los usuarios en base a características biológicas, entre las que se incluyen:

- Olor corporal
- Estructura facial
- Huellas dactilares
- Patrones de retina o de iris
- Trazado de venas
- Voz

El control biométrico de acceso tiene sus pros y sus contras; por un lado, dichos controles ofrecen un alto grado de seguridad, sobre todo los sistemas que utilizan los datos de las huellas dactilares, sin embargo, existen impedimentos prácticos para establecer un enfoque completamente biométrico, por ejemplo, en las exploraciones retinales, se bombardea al ojo con luz infrarroaja, las estructuras fotoreceptoras de la capa exterior responden reflejando dicha luz y la reflexión resultante produce una imagen de los patrones de los vasos sanguíneos de la retina, mediante este tipo de exploraciones se revela información personal médica, a través de los cuales se puede detectar indicios de abuso de drogas, enfermedades hereditarias, SIDA, etc. de ahí que el mantenimiento de una base de datos de patrones de retina puede llevar a un litigio. [A 3.04]

Más allá de los aspectos legales, los sistemas de control de acceso biométrico tiene implicaciones sociales, los empleados pueden ofenderse por estos controles y considerarlos una violación de la intimidad, lo que podría fomentar un ambiente de trabajo hostil, aún cuando no se manifieste de manera abierta.

Finalmente se puede decir que los controles de acceso biométrico son excelentes cuando se utilizan internamente, en lugares cerrado y entre compañeros en los que se confía, se recomienda su uso dentro de las oficinas en las que se encuentran las

máquinas que se usen para el control y la administración de la red.

3.2.4 Hardware de red

La seguridad del hardware de red es otro tema de vital importancia, los errores cometidos a este nivel pueden llevar al desastre, para comprender véase la siguiente figura 3.01:

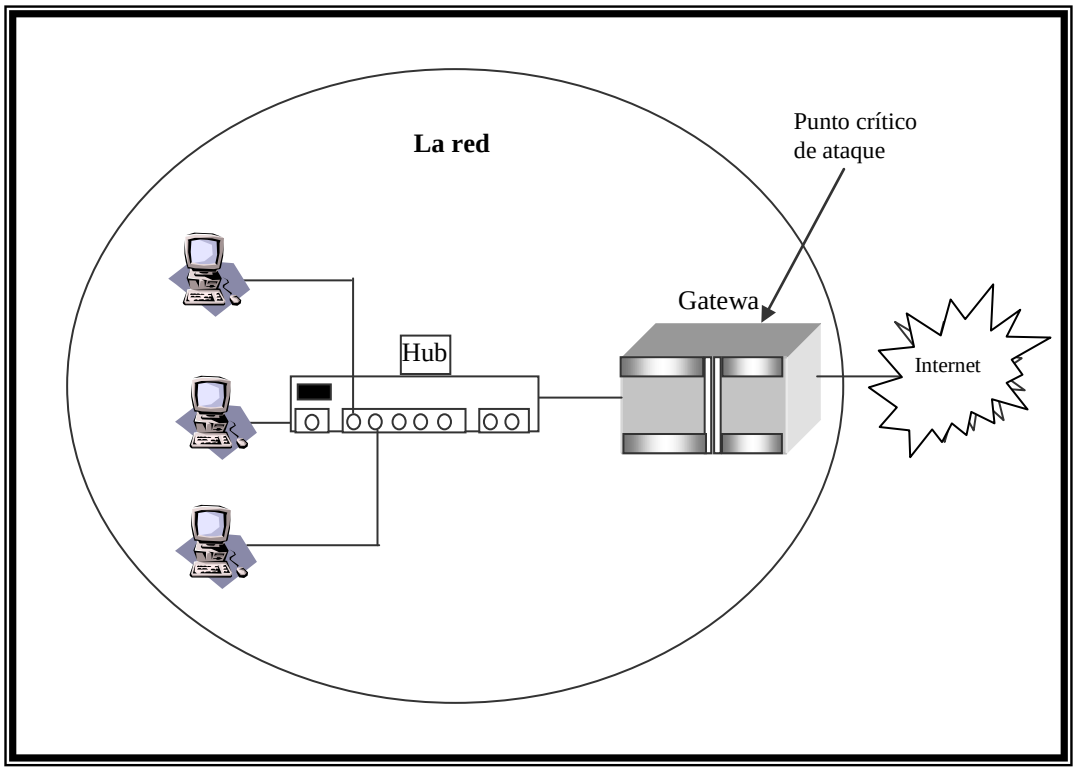

Fig. 3.01 Hardware de red

Como se muestra en la figura 3.01 el router o gateway (en el caso de la UTN) es un punto crítico de ataque, una pasarela a través de la que los usuarios se comunican con el exterior y viceversa, si los agresores consiguen colapsar los routers, switches o hubs, pueden denegar el servicio a mucha gente.

Se puede evitar poner en peligro el hardware de red empleando algunas prácticas de sentido común, en la mayoría de los casos estos pasos serán suficientes, ya que los problemas de los puntos

vulnerables de hardware de red no son habituales comparados con los del software.

Generalmente, el riesgo del hardware de red se produce por errores del operador, muchos usuarios no activan el cifrado o no definen contraseñas de administración, de mantenimiento o de usuarios, lo que deja la configuración del hardware intacta, como salió de fábrica y abre el sistema a ataques.

Además se debe aislar el hardware de red de los usuarios locales en los que no se confíe, muchos routers, bridges y switches proporcionan los medios para realizar recuperación in situ de la contraseña, los usuarios no controlados que tengan acceso físico pueden acometer este procedimiento.

En resumen para asegurar el hardware de red se deben seguir los siguientes pasos:

- Definir contraseñas de administración, de mantenimiento y de usuarios para evitar que los agresores puedan acceder a través de los valores predeterminados. Además, hay que asegurarse que dichas contraseñas no coinciden con otras contraseñas administrativas de la red.

- La mayoría de los routers (y algunos switches) admiten cifrado, pero no lo emplean por default, hay que asegurarse de que se ha activado el cifrado.

- Si no se necesita el control remoto de administración (acceso mediante telnet) es mejor desactivarlo.

- Si el hardware de red tiene puertos sensibles, filtrar y bloquear el acceso a ellos, es una excelente medida de seguridad.

- Si el hardware de red cuenta con opciones de verificación por expiración del tiempo de espera o por sesiones, es conveniente usarlas, ya que evitarán que los agresores puedan tomar el control o burlar las sesiones.

3.2.5 Dispositivos antirrobo

Otra amenaza es el robo, tanto del sistema entero como de componentes individuales, no es necesario que roben el servidor pueden llevarse los dispositivos del disco duro, memoria o tarjetas de expansión y esto ya representaría un gran problema. A continuación se listan unas herramientas que pueden ayudar a proteger el sistema y sus componentes:

- **Laptop Lockup.-** Evita el robo de equipos portátiles utilizando cables de acero resistentes al sabotaje y un candado de cobre que asegura el portátil a la mesa.

- **FlexLock**-50.- Asegura las estaciones de trabajo con un cable de media pulgada resistente a cizallas, cortaalambres y sierras para metal. También existen sistemas de base metálica que aseguran las estaciones de trabajo a las mesas.

- **Computer Guardian**.- Es un sistema antirrobo para PC independiente de la plataforma. Consta de una tarjeta de expansión y software en un disquete externo, cuando se mueve el PC o alguien trata de forzar sus componentes, el sistema hace sonar una sirena para asustar al ladrón y avisar a los demás.

- **Phazer**.- Es un dispositivo de seguridad de fibra óptica que detecta intentos de forzado físico. Este sistema de monitorización descansa sobre un bucle cerrado de fibra óptica, si el bucle se abre, se genera una alarma.

3.2.6 Números únicos, marcado y otras técnicas

También es aconsejable el tomar medidas para identificar el sistema en caso de robo, pues muchas veces los usuarios no conservan recibos o no anotan los números de series, lo cual vuelve imposible el recupera un equipo robado.

Algunas medidas de seguridad habituales que pueden servir como refuerzo legal son las siguientes:

- Llevar un registro meticuloso de todo el hardware, incluyendo los números de modelo y serie, ya que son necesarios si se llama a la policía.

- Marcar de forma permanente los componentes con un número único de identificación utilizando tinta indeleble, pintura fluorescente o pintura-tinta ultravioleta, que es visible sólo con luz negra. En particular se debe marcar la placa madre, las tarjetas de expansión, las unidades de disco, el interior y exterior del CPU y el monitor.

Además existen marcas patentadas o soluciones de identificación que pueden ayudar en esta tarea, en particular se mencionan dos:

- **STOP.-** Es un sistema a dos niveles tanto de prevención de robo como de identificación. En primer lugar se marca todo el hardware con un producto químico indeleble, este tatuaje identifica el equipo como una propiedad robada, se coloca encima una placa de metal especial que se mantendrá adherida bajo 800 libras de presión. Los ladrones sólo pueden vencer a STOP cortando físicamente el chasis plateado tatuado.

- **Accupage.**- Es un sistema de hardware que incrusta un mensaje indeleble en los PC's, que contiene la identidad de su propietario. Accupage se está integrando sobre los nuevos equipos portátiles, pero los sistemas de sobremesa más antiguos también puede asegurarse.

3.2 INSTALACION DEL SISTEMA OPERATIVO LINUX REDHAT 7.1

Antes de proceder con la instalación de Linux en un equipo, es necesario conocer el hardware del mismo, por consiguiente se debe familiarizar con el hardware del computador y responder a las siguiente preguntas:

1. ¿Cuántos discos duros?	2
2. ¿Cuál es la capacidad de cada disco?	18GB
3. ¿Sabe cuál es el disco primario?	SI
4. ¿De qué tipo es el disco duro?	SCSI
5. ¿De cuánta memoria RAM dispone?	512MB
6. ¿Qué tipo de mouse tiene?	PS/2
7. ¿Cuánto botones tiene el mouse?	2
8. ¿Qué clase de monitor tiene?	
9. Si se va a conectar a una red, se debe conocer lo siguiente:	
a) Dirección IP	208.235.198.252
b) Máscara de red	255.255.255.248
c) Dirección del gateway	208.235.198.193
d) Dirección IP del DNS	208.235.198.252

e) Nombre del dominio	utn.edu.ec
f) Nombre del host	ns.utn.edu.ec
g) Tipo de tarjeta de red	PCI
h) Número de tarjetas	2

La pregunta 9 debe contestarse por cada tarjeta de red que se tenga, la datos anteriores se refieren a la tarjeta de red que se conecta a Internet, a continuación se presenta la información de la tarjeta que se conecta a la LAN de la Universidad:

a) La dirección IP	172.20.1.1
b) La máscara de red	255.255.0.0
c) El nombre del host	svrlinuxutn
d) Tipo de tarjeta de red	PCI

Una vez que se conoce esta información básica, se procede con el proceso de instalación, el mismo que se explica detalladamente en el Anexo 1.

3.4 SEGURIDAD Y OPTIMIZACIÓN DEL SISTEMA

Una vez que se ha instalado Linux en el servidor es momento de asegurar los servicios y el software instalado, de esta manera se pueden prevenir muchos ataques antes de que ocurran.

A continuación se presenta ciertas características que pueden ayudar a prevenir ataques tanto internos como externos, entre estos se tiene:

3.4.1 BIOS

Se recomienda deshabilitar el arranque desde disquete y activar las características de password de la BIOS, para mejorar la seguridad del sistema.

De esta manera se bloquea el acceso de personal no autorizado que intente arrancar el equipo con un disco especial o que quiera cambiar las características de la BIOS ya sea el tipo de disco duro, el password, la fecha, etc.

3.4.2 Política de seguridad

Es importante la implementación de la política de seguridad de la institución, pues en ella se establecen las reglas básicas a través de las cuales los usuarios se guiarán para utilizar la red de una manera óptima.

En el capítulo 2 se creó la política de seguridad de la Universidad, en ella se dieron a conocer unas pautas que los usuarios deben de seguir para ayudar a optimizar el uso de la red y del Internet para de esta manera lograr mayor seguridad.

Cabe mencionar que la seguridad no solamente debe estar implementada en el servidor, pues él es solamente un punto dentro de toda red, se necesita la cooperación de todos los usuarios para que la red no se vea comprometida, pues a pesar que de se aseguren todos y cada uno de los servicios que ofrece el servidor, se instale un firewall, nada es suficiente si es que los usuarios no colaboran y se comprometen a cumplir lo establecido en la Política de Seguridad de la Universidad.

3.4.3 La longitud del password

Un punto importante en la seguridad de Linux es la longitud del password, muchas veces las personas usan passwords demasiado pequeños, los cuales son fáciles de descubrir. Es conocido que no existe un password indescifrable, pero mientras de mayor longitud sea este más complicado será descubrirlo.
Para establecer los passwords se recomienda:

- Que tengan un longitud de por lo menos 8 caracteres y que incluya letras, números o caracteres especiales.

- No debe ser trivial; un password trivial es algo fácil de descubrir, como puede ser el nombre de un familiar, la fecha de cumpleaños, una ocupación o alguna característica personal.

- Debe de tener un tiempo de duración determinado, luego del cual se le obligará al usuario a cambiar de password.

3.4.4 La cuenta root

La cuenta root es la que más privilegios tiene dentro de un sistema Linux. El root no tiene restricciones impuestas dentro del sistema, él puede hacer todo lo que desee el sistema no le pregunta nada y le permite realizar cualquier tarea. Cuando se usa esta cuenta es importante ser lo más cuidadoso posible; por ninguna razón se debe iniciar una sesión con este usuario a no ser que se vayan a realizar tareas administrativas que necesariamente deban ser realizadas por el root.

3.4.5 Establecer el tiempo inactividad de la cuenta root

Este parámetro se establece, si por alguna situación especial el administrador sale intempestivamente y por error olvida terminar su sesión, en ese caso estaría dejando a la terminal con la cuenta root activa y cualquier persona podría acercarse y causar algún daño.

Para solucionar este inconveniente, se puede establecer una variable la cual determina el tiempo que la cuenta puede estar sin realizar ninguna actividad, luego del cual esta automáticamente cierra su sesión.

- Editar el archivo **profile** (/etc/profile) y añadir la siguiente línea a continuación de "HISTSIZE=":

```
TMOUT=7200
```

El valor que se ha asignado a "TMOUT" es en segundos y representa 2 horas, lo cual significa que automáticamente se terminará la sesión después de dos horas de inactividad.

3.4.6 LILO y el archivo lilo.conf

Lilo es el encargado de manejar el proceso de arranque en Linux por lo tanto es muy importante protegerlo de la mejor manera.

En este archivo se pueden establecer características de seguridad entre las cuales se pueden mencionar:

timeout=00

Esta opción controla el tiempo que LILO espera para que el usuario ingrese algún modo de arranque especial antes de arrancar con el modo por default. Uno de los requerimientos de seguridad C2 es que este intervalo sea cero a no ser que se tengan dos sistemas operativos en el mismo equipo.

restricted

Esta opción pregunta el password cuando se inicia el sistema en modo "single", si no se emplea esta característica se corre el riesgo de que alguien inicie el sistema con privilegios de root y sin necesidad de conocer el password.

password=<password>

Aquí se escribe el password que se usará cuando el sistema inicie en modo single.

Los pasos para asegurar LILO, a través del archivo /etc/lilo.conf son:

Paso 1

El archivo lilo.conf (/etc/lilo.conf) debe quedar de la siguiente manera:

```
boot=/dev/sda
map=/boot/map
install=/boot/boot.b
timeout=00
linear
message=/boot/message
default=linux
restricted
password=<password>
image=/boot/vmlinuz-2.4.2-2
label=linux
initrd=/boot/initrd-2.4.2-2.img
read-only
root=/dev/sda6
```

Paso 2

Ya que este archivo contiene un password no encriptado, es necesario cambiarle los permisos para que solamente el root pueda leerlo.

```
[root@utn /] # chmod 600 /etc/lilo.conf
```

Paso 3

Siempre que se haga un cambio en lilo.conf para activar los cambios se debe ejecutar el siguiente comando:

```
[root@utn /]#  /sbin/lilo -v
```

Paso 4

Una característica más de seguridad es hacer inmutable el archivo lilo.conf.

```
[root@utn /] chattr +i /e/tc/lilo.conf
```

3.4.7 Deshabilitando las teclas Ctrl-Alt-Delete

Un buen control de seguridad consiste en deshabilitar las teclas Ctrl-Alt-Delete debido a que existen muchas personas malintencionadas que pueden acercarse al servidor y apagarle presionando estas teclas con lo cual pueden causar varios daños en el sistema.

Paso 1

Editar el archivo inittab (/etc/inittab) y cambiar la siguiente línea:

```
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

Para leer

```
#ca::ctlaltdelete:/sbin/shutdown -t3 -r now
```

Paso 2

Para que los cambios tengan efecto se ejecuta el siguiente comando:

```
[root@utn /]# /sbin/init q
```

3.4.8 El archivo /etc/services

Los números de los puertos de los servicios estándares se definen en el archivo /etc/services, este archivo habilita para que los programas cliente y servidor conviertan nombres de servicio a estos números de puertos. Solamente el root está autorizado para hacer modificaciones en este archivo, aunque esta es una tarea que no se hará muy a menudo es conveniente tomar medidas de seguridad para este archivo.

- Inmunizar el archivo /etc/services

```
[root@utn /] # chattr +i /etc/services
```

3.4.9 El archivo /etc/securetty

El archivo /etc/securetty permite especificar desde cuales TTY y VC (consola virtual) el usuario root está permitido ingresar al servidor.

- Editar el archivo securetty (/etc/securetty) y comenta las líneas:

```
vc/1                    tty1
#vc/2                   #tty2
#vc/3                   #tty3
#vc/4                   #tty4
#vc/5                   #tty5
#vc/6                   #tty6
#vc/7                   #tty7
#vc/8                   #tty8
#vc/9                   #tty9
```

Lo cual significa que al root se le permite ingresar solamente desde tty1 y vc/1. Es muy recomendable hacer que el root solamente ingrese en una tty o vc y usar el comando su en caso de querer cambiarse de usuario.

3.4.10 Cuentas especiales

Es importante deshabilitar las cuentas que se crean en Linux por default, pero que no se van a utilizar. Linux provee cuentas para varias actividades del sistema, las cuales no son necesarias si los servicios no están instalados en el equipo, por lo tanto lo mejor es removerlas.

Paso 1

Remover los usuarios innecesarios del archivo /etc/passwd, para la operación segura del servidor

```
[root@utn /] # userdel adm
[root@utn /] # userdel lp
[root@utn /] # userdel shutdown
[root@utn /] # userdel halt
[root@utn /] # userdel news
[root@utn /] # userdel uucp
[root@utn /] # userdel operator
[root@utn /] # userdel games
[root@utn /] # userdel gopher
```

Paso 2

Remover los grupos de las cuentas anteriormente eliminadas del archivo /etc/group.

```
[root@utn /] # userdel adm
[root@utn /] # userdel lp
[root@utn /] # userdel news
[root@utn /] # userdel uucp
[root@utn /] # userdel games
[root@utn /] # userdel dip
```

Paso 3

Establecer el bit inmutables a los archivos /etc/passwd, /etc/group, /etc/shadow, /etc/gshadow, para prevenir que sean accidentalmente eliminados o sobrescritos por usuarios no autorizados.

```
[root@utn /] # chattr +i /etc/passwd
[root@utn /] # chattr +i /etc/group
[root@utn /] # chattr +i /etc/gshadow
[root@utn /] # chattr +i /etc/shadow
```

3.4.11 Controlando la forma como se inicia un sistema de archivos

Se debe controlar la manera como se montan el sistema de archivos en cada una de las particiones, se debe establecer algunas opciones como noexec, nodev, nosuid. Esto se puede configurar en el archivo /etc/fstab, el cual contiene información descriptiva acerca de las diferentes opciones para montar el sistema de archivos.

La información relacionada con las opciones de seguridad en el archivo /etc/fstab son:

defaults Permite cualquier cosa en la partición (quota, read-write y suid)

noquota No permite quotas de usuarios en la partición

nosuid No permite accesos SUID/GUID en la partición

nodev No permite configurar caracteres o acceso a dispositivos especiales en la partición.

quota Permitir quotas de usuarios en la partición.

ro Permitir solamente lectura en la partición.

rw Permitir lectura y escritura en la partición.

suid Permitir accesos SUID/GUID en la partición.

Paso 1

Editar el archivo /etc/fstab

```
LABEL=/cache      /cache  ext2    defaults,nodev          1  2
LABEL=/home       /home   ext2    defaults,nosuid         1  2
LABEL=/tmp        /tmp    ext2    defaults,nosuid,nodev   1  2
```

Paso 2

Una vez hechos los cambios necesarios se debe actualizar la información remontando cada una de las particiones afectadas.

```
[root@utn /] # mount /cache -oremount
[root@utn /] # mount /home -oremount
[root@utn /] # mount /tmp -oremount
```

3.4.12 Iniciando el directorio /boot como de solo lectura

El directorio /boot es donde se encuentra el kernel de Linux y algunos archivos relacionados con el mismo. Un aspecto muy importante en la seguridad es hacer que este directorio se monte como de solo lectura para evitar la escritura equivocada o malintencionada de algún usuario.

Paso 1

Editar el archivo /etc/fstab y modificar la línea:

```
LABEL=/boot          /boot    ext2     defaults,ro                    1  2
```

Paso 2

Actualizar los cambios ejecutando el comando:

```
[root@utn /] # mount /boot -oremount
```

Después de realizar estos cambios las particiones quedarán de la siguiente manera:

```
[root@deep /]# cat /proc/mounts

/dev/root      /          ext2     rw                  0     0
/proc          /proc      proc     rw                  0     0
/dev/sda1      /boot      ext2     ro                  0     0
/dev/sda10     /cache     ext2     rw,nodev            0     0
/dev/sda9      /chroot    ext2     rw                  0     0
/dev/sda8      /home      ext2     rw,nosuid           0     0
/dev/sda13     /tmp       ext2     rw,noexec,nosuid    0      0
/dev/sda7      /usr       ext2     rw                  0     0
/dev/sda11     /var       ext2     rw                  0     0
/dev/sda12     /var/lib   ext2     rw                  0     0
none           /dev/pts   devpts   rw                  0     0
```

3.4.13 Shell logging

Todos los comandos que se digitan en el terminal se almacenan en un archivo llamado ~/.bash_history (donde ~/ es el directorio home de cada usuario). Generalmente se almacenan los 500

últimos comandos digitados lo cual implica un riesgo, pues alguien puede acercase al equipo y saber que es lo que se ha hecho e inclusive podría ejecutar algún comando importante y dañar el sistema, por lo tanto reducir el número de comandos que se almacenan en ~/.bash_history es una buena medida de seguridad y ayuda a proteger a los usuarios.

Paso 1

La variable que determina el número de comandos que se almacenan en ~/.bash_history es HISTSIZE la cual se define en /etc/profile para que sea válida para todos los usuarios.

- Editar el archivo /etc/profile y establecer que solamente se almacenen los últimos 10 comandos digitados.:

```
HISTSIZE=10
```

Paso 2

Además se debe añadir la variable HISTFILESIZE=0, la cual determina que el archivo .bash_history debe ser eliminado cada vez que un usuario termina su sesión, de esta manera no se permitirá que nadie examine dicho archivo para saber que estuvo haciendo un usuario determinado.

- Editar el archivo /etc/profile y bajo HISTSIZE añadir la siguiente línea:

```
HISTFILESIZE=0
```

Luego de realizar estos cambio se debe terminar la sesión y volver a ingresar como usuario root para que los cambios se activen.

3.4.14 Proteger los archivos bajo /etc/rc.d/init.d

Establecer correctamente los permisos de los archivos que se encuentran bajo el directorio /etc/rc.d/init.d es muy importante debido que estos son los encargados de iniciar y finalizar la mayoría de servicios que presta el servidor.

- Ejecutar el siguiente comando:

```
[root@utn /] # chmod -R 700 /etc/rc.d/init.d/*
```

Lo cual significa que solamente el super-usario root está permitido para leer, escribir y ejecutar estos scripts.

3.4.15 El archivo /etc/rc.d/rc.local

Por default cuando se inicia un servidor Linux, este muestra el nombre de la distribución de Linux, versión, versión del kernel y nombre del servidor; esto es brindar mucha información a terceras personas y no es conveniente, por tal razón al inicio únicamente debe aparecer la palabra "login" sin ninguna otra información.

Paso 1

Editar el archivo /etc/rc.d/rc.local y colocar el carácter "#" delante de las siguientes líneas:

```
# This will overwrite /etc/issue at every boot. So, make any changes you
# want to make to /etc/issue here or you will lose them when you reboot.
#echo "" > /etc/issue
#echo "$R" >> /etc/issue
#echo "Kernel $(uname -r) on $a $(uname -m)" >> /etc/issue
#
#cp -f /etc/issue /etc/issue.net
#echo >> /etc/issue
```

Paso 2

Remover los archivo issue.net e issue

```
[root@utn /] # rm -f /etc/issue
[root@utn /] # rm -f /etc/issue.net
```

3.4.16 Encontrando archivos .rhosts

La existencia de archivos .rhosts en el servidor podría ser parte de las tareas comunes de administración, pero estos archivos no deben permitirse en el sistema. Se debe recordar que un cracker solamente necesita un punto de inseguridad para ganar acceso e ingresar a la red.

Paso 1

Buscar archivos .rhosts en el sistema

```
[root@utn /] # find /home -name .rhosts
```

Paso 2

Se puede crear un cron job que periódicamente chequee y envíe un reporte en caso de encontrar archivos .rhosts en el directorio /home, para proceder a eliminarlos y de esa manera evitar problemas a futuro.

- Crear el archivo find_rhosts_files en /etc/cron.daily y añadir lo siguiente:

```
#!/bin/sh
  /usr/bin/find /home -name .rhosts | (cat <<EOF
  Este es un reporte automático de la posible exitncia de archivos ".rhosts"
  En el servidor utn.edu.ec.
  EOF
  cat
  ) | /bin/mail -s "Contenido de archivos .rhosts"
```

- Hacer ejecutable el script y establecer que el propietario es el usuario root.

```
[root@utn /] # chmod 755 /etc/cron.daily/find_rhosts_files
[root@utn /] # chown 0.0 /etc/cron.daily/find_rhosts_files
```

3.5 PLUGGABLE AUTHENTICATION MODULES - PAM

Los Pluggable Authentication Modules (PAM) consisten de librerías compartidas, las cuales habilitan al administrador a escoger como se autentican los usuarios con las aplicaciones.

Básicamente PAM habilita la separación de esquemas de autentificación para las aplicaciones. Este se completa debido a que provee una librería de funciones que las aplicaciones pueden emplear para solicitar autenticación de usuarios SSH, POP, IMAP que son aplicaciones PAM-aware, por lo tanto estas aplicaciones pueden ser modificadas para proveer un password cambiando los módulos PAM sin tener que reescribir ningún código en estas aplicaciones.

Los archivos de configuración de los módulos PAM están ubicados en /etc/pam.d y los módulos en sí están ubicados en el directorio /lib/security. El directorio /etc/pam.d contiene una colección de archivos como ssh, pop, imap. Las aplicaciones PAM-aware que no tienen un archivo de configuración automáticamente toman la configuración del archivo other.

A continuación se establecen controles mínimos de seguridad a través del uso de PAM.

3.5.1 La longitud del password

La longitud mínima de un password por default en un sistema Linux es de 5 caracteres. Esto significa que la contraseña de un usuario debe contener como mínimo 5 caracteres entre letras, números, caracteres especiales, etc. Esto no es suficiente y debe ser de por lo menos 8 caracteres. Los argumentos que controlan en Linux la longitud de las contraseñas son: minlen, dcredit, ucredit, lcredit y ocredit.

Paso 1

Para prevenir que se habilite la característica por default de 5 caracteres mínimo para un contraseña, se debe editar el archivo /etc/pam.d/passwd y establecer la longitud mínima en otro valor.

- Editar el archivo /etc/pam.d/passwd y remover la siguiente línea:

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

Paso 2

Una vez que se ha removido la línea indicada anteriormente, es hora de eliminar las siguientes tres líneas de archivo system-auth. Este es un inconveniente del RPM de PAM en Red Hat y es

necesario corregirlo aquí para poder usar la característica que se desea.

- Editar el archivo /etc/pam.d/system-auth y remover las siguientes líneas

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

Paso 3

Ahora se debe añadir las siguientes líneas en /etc/pam.d/passwd. Al usar el módulo pam_cracklib con el argumento minlen se establece la longitud del password.

```
password    required      /lib/security/pam_cracklib.so      retry=3 minlen=12
password    sufficient    /lib/security/pam_unix.so nullok use_authtok md5
shadow   password     required   /lib/security/pam_deny.so
```

Después de realizar estas modificaciones los archivos /etc/pam.d/passwd y /etc/pamd/system-auth deben verse así:

```
#%PAM-1.0
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_cracklib.so retry=3 minlen=12
password sufficient /lib/security/pam_unix.so nullok use_authtok md5
shadow
password required /lib/security/pam_deny.so
```

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth required /lib/security/pam_env.so
auth sufficient /lib/security/pam_unix.so likeauth nullok
auth required /lib/security/pam_deny.so
account required /lib/security/pam_unix.so
session required /lib/security/pam_limits.so
session required /lib/security/pam_unix.so
```

3.5.2 Deshabilitar todos los accesos a la consola

La librería PAM instalada por default permite al administrador del sistema escoger como los usuarios se autentifican en las aplicaciones, así como también los accesos a la consola, programas y archivos. Para deshabilitar todos estos accesos es necesario comentar todas las líneas referentes a pam_console.so dentro del directorio /etc/pam.d. Para esto se creará un script que realizará automáticamente esta tarea.

Paso 1

Crear el archivo disabling.sh y añadir lo siguiente:

```
# !/bin/sh
cd /etc/pam.d
for i in * ; do
sed '/[ ^#].*pam_console.so/s/^/#/' < $i > foo && mv foo
$i
done
```

Paso 2

Hacer ejecutable el script y ejecutarlo

```
[root@utn /] # chmod 700 disabling.sh
[root@utn /] # ./disabling.sh
```

3.5.3 Tablas de control de acceso

El acceso legítimo a un servidor es un aspecto del cual siempre hay que estar pendiente, por lo tanto el uso de un archivo de seguridad que permita tener un control sobre quien y desde donde se conecta al servidor será de mucha ayuda para un mejor el control. Afortunadamente este archivo existe y se llama "access.conf" y se encuentra bajo el directorio /etc/security.

El archivo access.conf el cual se instala en el sistema Linux permite controlar a los usuarios que pueden conectarse al servidor y desde donde lo pueden hacer. Siempre se debe recordar que el acceso al servidor puede ser local o remoto y por tanto es necesario está protegido de estos dos tipos de acceso.

Paso 1

La política de seguridad que se ha establecido es denegar todo, por lo tanto en el archivo access.conf lo primero que se hará será establecer esta política.

- Editar el archivo access.conf y añadir la siguiente línea:

```
-:ALL EXCEPT root prueba :ALL
```

En la línea anterior se está estableciendo que se niega el acceso tanto local como remoto a todos los usuarios, excepto al usuario root y prueba. Cabe anotar que además de especificar los usuarios se puede anotar la direcciones IP si se desea ser más concreto.

Paso 2

Para habilitar el uso de access.conf en Linux es necesario usar añadir unas líneas en el archivo /etc/pam.d/login.

- Editar el archivo /etc/pam.d/login y escribir.

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

3.5.4 Limitar recursos

El archivo limits.conf ubicado bajo /etc/security puede ser usado para controlar y limitar el uso de recursos de los usuarios del sistema. Es importante limitar el uso de los recursos a los usuarios para evitar ataques denegación de servicio (número de procesos, cantidad de memoria, etc).

Paso 1

Editar el archivo /etc/security/limits.com y añadir lo siguiente:

```
*   hard    core    0
*   hard    rss     5000
*   hard    nproc   20
```

Esto dice que se prohibe la creación de archivos core "core 0", se restringe el número de procesos a 20 "nproc 20" y se restringe el uso de memoria a 5M "rss 5000", para todos los usuarios excepto para el root. Con esta clase de quota se tiene un mejor control de los procesos, el uso de la memoria y los archivos core que los usuarios del sistema pueden manejar. El "*" significa que la regla se aplica para todos los usuarios.

Paso 2

Editar el archivo /etc/pam.d/login y añadir la siguiente línea en el final del archivo

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

3.5.5 Bloqueando los usuarios que pueden ejecutar el comando su al root

El comando su (Substitute User) permite que otro usuario del sistema se cambie temporalmente de usuario específicamente en este caso al root y ejecute comandos que solamente él puede hacerlo. Esto no es conveniente para la seguridad del sistema y por tanto se debe restringir los usuarios a los que se les permite ejecutar su al root. Para lograr este objetivo es necesario modificar el archivo /etc/pam.d/su.

Paso 1

Editar el archivo /etc/pam.d/su y descomentar la siguiente línea:

```
authrequired          /lib/security/pam_wheel.so      use_uid
```

Lo cual significa que solamente los miembros del grupo wheel puede hacer su al root; el grupo wheel es una cuenta especial en el sistema y se la puede usar para este propósito.

Paso 2

Anteriormente se definió el grupo wheel dentro del archivo /etc/pam.d/su, entonces ahora es momento de añadir un usuario a este grupo.

- Añadir el usuario prueba1 al grupo wheel.

```
[root@utn /] # usermod -G10 prueba1
```

G representa una lista de grupos suplementarios donde el usuario es también miembro, 10 es el valor numérico del ID de usuarios del grupo whell y prueba1 es el nombre del nuevo usuario.

3.6 ADMINISTRACION DE RED TCP/IP

Linux es uno de los mejores operativos para manejo de red, muchos sitios Internet alrededor del mundo lo usan actualmente. Conociendo el hardware de red y todos los archivos relacionados, se puede tener un gran control sobre el funcionamiento y optimización de la red.

3.6.1 Archivos para manejo de red

En Linux, la red TCP/IP se configura a través de varios archivos de texto, los cuales pueden ser modificados para cambiar la funcionalidad de la red.
A continuación se describen todos y cada uno de los archivos que necesitan ser configurados para que el servidor se conecte tanto a la red interna como a Internet.

3.6.1.1 Los archivos /etc/sysconfig/network-scripts/ifcfg-ethN

Los archivos de configuración de cada una de las tarjetas de red que se instalan en el equipo se cargan en el directorio /etc/sysconfig/network-scripts y se llaman ifcfg-eth0 para la interfaz eth0, ifcfg-eth1 para la interfaz eth1 y así sucesivamente

por cada interfaz añadida. Siempre es necesario revisar estos archivos para verificar que los parámetros son los adecuados.
A continuación se muestra un ejemplo del archivo /etc/sysconfig/network-scripts/ifcfg-eth0

```
DEVICE=eth0
BOOTPROTO=static
BROADCAST=208.164.186.255
IPADDR=208.164.186.1
NETMASK=255.255.255.0
NETWORK=208.164.186.0
ONBOOT=yes
USERCTL=no
```

DEVICE = Nombre de la interfaz.

BOOTPROTO = Protocolo de inicio, que puede ser:

static - Dirección IP estática, es la característica por default de Linux

none - Ningún protocolo debe ser usado al iniciar el equipo

bootp - Usar protocolo bootp

dhcp - Usar protocolo dhcp

BROADCAST = Dirección de difusión

IPADDR = Dirección IP

NETMASK = Máscara de red

NETWORK = Dirección de red

ONBOOT = Determina si se inicializa o no la tarjeta en el momento de encender el equipo.

USERCTL = Determina si otros usuarios a parte del root pueden administrar este dipsositivo.

3.6.1.2 El archivo /etc/resolv.conf

El archivo resolv.conf se emplea para resolver direcciones IP a nombre de hosts.

El siguiente es un ejemplo del archivo /etc/resolv.conf

```
domain dominio.com
search ns1.dominio.com ns2.dominio.com dominio.com
nameserver 208.164.186.1
nameserver 208.164.186.2
nameserver 127.0.0.1
```

3.6.1.3 El archivo /etc/host.conf

El archivo /etc/host.conf especifica como se van a resolver los nombres. Linux usa una librería para obtener direcciones IP que corresponden a nombres de hosts

El siguiente es un ejemplo de archivo /etc/host.conf

```
#Lookup names via /etc/hosts.conf first fall back to DNS resolver
order hosts,bind
#We have machines with multiple addresses
multi on
```

La opción **order** indica el orden de servicios, con las instrucciones anteriores se está indicando que la librería que resuelve nombres primero consulte el archivo /etc/hosts para resolver un nombre y luego chequee el servidor de nombres (DNS).

La opción **multi** determina si un host contenido en /etc/hosts puede tener múltiples direcciones IP. Un host que tiene múltiples direcciones IP se conoce como multihomed, porque la presencia de múltiples direcciones IP implica que el host tiene varias tarjetas de red.

3.6.1.4 El archivo /etc/sysconfig/network

El archivo /etc/sysconfig/network es usado para información específica de la configuración de red del servidor.

El siguiente es un ejemplo del archivo /etc/sysconfig/network

```
NETWORKING=yes
HOSTNAME=nombre_host
GATEWAY=207.35.78.1
GATEWAYDEV=eth0
```

NETWORKING = Establece si el equipo está configurado para trabajar en red o no.

HOSTNAME = Indica el nombre del servidor

GATEWAY = La dirección IP el gateway remoto

GATEWAYDEV = El nombre del dispositivo que se usa para acceder al gateway remoto.

3.6.1.5 El archivo /etc/sysctl.conf

En la versión 7.1 de Linux RedHat todos los parámetros del kernel están disponibles bajo /etc/sys/ los mismos que pueden ser configurados mientras el servidor está funcionando, para lo cual se usa el archivo /etc/sysctl.conf. Este archivo se lee y carga

cada vez que se enciende el equipo o se reinicia el servicio network.

Como se está hablando de la configuración de red, en este momento se va a activar la opción del kernel IPV4 forwarding.

Paso 1

Editar el archivo /etc/systcl.conf y añadir la línea

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

Paso 2

Después de realizar un cambio en /etc/sysctl.conf es necesario reiniciar el servicio network para actualizar los cambios.

- Para reiniciar el servicio network ejecutar el siguiente comando

```
[root@utn /] # /etc/rc.d/init.d/network restart
```

3.6.1.6 El archivo /etc/hosts

Cuando se arranca el servidor, es necesario que este conozca las direcciones IP y nombres de hosts de algunos equipos antes de que el servicio DNS sea referenciado. Este mapa básico de direcciones se almacena en el archivo /etc/hosts. En caso de ausencia de un servidor de dominios, cualquier programa de red puede consultar este archivo para determinar la dirección IP que corresponde a un host.

El siguiente es un ejemplo de /etc/hosts

Dirección IP	Hostname	Alias
127.0.0.1	localhost.localdomain	localhost
208.164.186.1	ns1.midominio.com	ns1
208.164.186.2	ns2.midominio.com	ns2
208.164.186.3	ns3.midominio.com	ns3

La primera columna es la dirección IP a ser resuelta, la segunda columna es el nombre del host y la última columna es un alias del host.

3.6.2 Asegurando la red TCP/IP

En el kernel de RedHat existen muchas opciones relacionadas con la seguridad de la red tales como: la eliminación de paquetes prohibidos que llegan a las interfaces, negación de solicitudes ping, etc., estas se pueden definir en el archivo /etc/sysctl.conf o en archivo /etc/rc.d/rc.local los cuales se cargan cada vez que el equipo se reinicia.

Entre las caraterísticas de seguridad de la red TCP/IP se tienen las siguientes:

3.6.2.1 Prevenir que el servidor responda a solicitudes ping
3.6.2.2 Rechazar solicitudes de broadcast
3.6.2.3 Protocolos de ruteo
3.6.2.4 Habilitar la protección TCP SYN Cookie
3.6.2.5 Deshabilitar ICMP Redirect Acceptance
3.6.2.6 Habilitar la protección de mensajes de error dañinos
3.6.2.7 Habilitar la protección IP spoofing

3.6.2.1 Prevenir que el servidor responda a solicitudes ping

El conjunto de protocolos TCP/IP tiene ciertas debilidades de las cuales los crackers se valen para bloquear sistemas indefensos. Previniendo que el sistema responda a solicitudes ping puede ayudar a minimizar estos problemas. Al no responder a solicitudes ping puede hacer pensar a los crackers que el servidor no esta disponible.

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea

```
net.ipv4.icmp_echo_ignore_all = 1
```

3.6.2.2 Rechazar solicitudes de broadcast

Cuando un paquete es enviado a una dirección de broadcast (ej. 192.168.1.255) desde una máquina en la red local, el paquete es entregado a todas las máquinas de la red. Entonces todas las máquinas de la red responden con mensajes ICMP la solicitud y el resultado puede ser una congestión en la red o pausas en la transmisión de datos (Ataques denegación del servicio).

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea

```
net.ipv4.icmp_echo_ignore_broadcasts = 1
```

3.6.2.3 Protocolos de ruteo

Ruteo y protocolos de ruteo pueden crear muchos problemas, el IP source routing, donde un paquete IP contiene detalles de la ruta que va a tomar, es peligroso de acuerdo a la RFC 1122 el

host destino debe responder sobre la misma ruta. Si un cracker tiene la posibilidad de enviar un source route packet dentro de la red, entonces él podría interceptar las respuestas y engañar al host haciéndole pensar que se está comunicando con un host verdadero.

Es altamente recomendado deshabilitar el IP source routing de todas las interfaces de red del sistema para así evitar un agujero de seguridad.

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea

```
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.lo.accept_source_route = 0
net.ipv4.conf.eth0.accept_source_route = 0
net.ipv4.conf.eth1.accept_source_route = 0
net.ipv4.conf.default.accept_source_route = 0
```

3.6.2.4 Habilitar la protección TCP SYN Cookie

Un "SYN Attack" es un ataque de negación de servicio que consume todos los recursos del equipo, forzando al administrador a reiniciar el equipo. Este tipo de ataques son fácilmente realizables desde recursos internos o conexiones externas y pueden ocasionar mucha congestión en el servidor.

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea

```
net.ipv4.tcp_syncookies = 1
```

3.6.2.5 Deshabilitar ICMP Redirect Acceptance

Cuando los hosts usan ruteos no óptimos hacia un destino particular, un paquete ICMP redirect es usado por el ruteador para informar al host cual debería ser la ruta correcta. Si un cracker es capaz de falsificar un paquete ICMP redirect , podría alterar las tablas de ruteo en el host y posiblemente arruinará la seguridad del host causando que el tráfico fluya a través de una ruta inadecuada.

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea

```
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.lo.accept_redirects = 0
net.ipv4.conf.eth0.accept_redirects = 0
net.ipv4.conf.eth1.accept_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
```

3.6.2.6 Habilitar la protección de mensajes de error dañinos

Esta opción alerta acerca del mensajes de error dañinos en la red.

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea

```
net.ipv4.icmp_ignore_bogus_error_responses = 1
```

3.6.2.7 Habilitar la protección IP spoofing

La protección spoofing previene a la red de la existencia de fuentes spoofed (fuentes engañosas) las cuales a menudo son usadas por DoS Attacks.

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea

```
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.lo.rp_filter = 1
net.ipv4.conf.eth0.rp_filter = 1
net.ipv4.conf.eth1.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1
```

3.6.3 Optimizando la red TCP/IP

A continuación se procederá a optimizar el funcionamiento de TCP/IP dentro de un sistema Linux, esto se realiza a través de características disponibles con el sistema instalado y permitirá una rendimiento óptimo de la red.

Entre los parámetros de optimización de la red TCP/IP se tienen los siguientes:

3.6.3.1 Recursos TCP/IP

Con las siguiente instrucciones se modifican los valores por default para conexiones TCP/IP, se disminuye la cantidad de tiempo que el servidor Linux tendrá activa una conexión antes de terminarla, este además terminará algunas extensiones IP que son necesarias en la comunicación.

- Editar el archivo /etc/sysctl.conf y añadir las siguientes líneas:

```
net.ipv4.tcp_fin_timeout = 30
net.ipv4.tcp_keepalive_time = 1800
net.ipv4.tcp_window_scaling = 0
net.ipv4.tcp_sack = 0
net.ipv4.tcp_tmestamps = 0
```

3.6.3.2 Recursos buffer-space

Los tres parámetros que se mencionan a continuación se relacionan con las opciones 'total', 'read' y 'write' de TCP buffer-space los cuales pertenecen al protocolo TCP/IP. Al modificar estas variables se incrementa al máximo la cantidad de TCP buffer-space disponibles en el sistema.

- Editar el archivo /etc/sysctl.conf y añadir las siguientes líneas:

```
net.ipv4.tcp_mem = 57344   57344
net.ipv4.tcp_wmem = 114688    458752     3670016
net.ipv4.tcp_rmem = 344064      1376256     11010048
```

3.6.3.3 Recursos buffer-size

Los siguientes parámetros están relacionados con los valores máximos y por default que se van a enviar y recibir a través de los buffers del servidor.

- Editar el archivo /etc/sysctl.conf y añadir las siguientes líneas:

```
net.core.rmem_max = 524280
net.core.rmem_default  =  524280
net.core.wmem_max  =  524280
net.core.wmem_default  =  524280
```

3.6.3.4 El parámetro ip_local_port_range

El parámetro ip_local_port_range define el rango de puertos locales que son usados por el tráfico TCP y UDP. Este está formado por dos números, el primero número es el primer puerto que se permite tráfico TCP y UDP y el segundo número es el último puerto permitido.

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea:

```
net.ipv4.ip_local_port_range = 16384   65536
```

3.6.3.5 Los parámetros ipfrag_high_thresh e ipfrag_low_thresh

Estos dos parámetros se relacionan con la cantidad de memoria usada para reensamblar paquetes IP fragmentados. Cuando los bytes de memoria de ipfrag_high_thresh son designados para este propósito, el manejador de fragmentos IP envía paquetes hasta alcanzar el valor de ipfrag_low_thresh.

- Editar el archivo /etc/sysctl.conf y añadir la siguiente línea:

```
net.ipv4.ipfrag_high_thresh = 512000
net.ipv4.ipfrag_low_thresh  =  446464
```

3.7 FIREWALL DE FILTRADO DE PAQUETES - IPTABLES

Un firewall de filtrado de paquetes se suele implementar dentro del sistema operativo y funciona en las capas de transporte y red. Protege el sistema realizando las decisiones de enrutamiento después de filtrar los paquetes basándose en la información del encabezado del paquete IP.

Un firewall de filtrado de paquetes IPTABLES consta de una lista de reglas de aceptación y denegación. Estas reglas definen explícitamente los paquetes se permiten pasar y los que no a través da interfaz de red. Las reglas usan los campos del encabezado IP, para decidir si enrutar un paquete hacia su destino, eliminar el paquete o bloquear un paquete y devolver una condición de error a la máquina emisora. Esas reglas se basan en una interfaz de red especifica y en la dirección IP del host, las direcciones IP origen y destino del nivel de red, los puertos de servicio TCP y UDP de la capa de transporte, los indicadores de conexión TCP, los tipos de mensaje ICMP del nivel de red y en si el paquete es entrante o saliente.

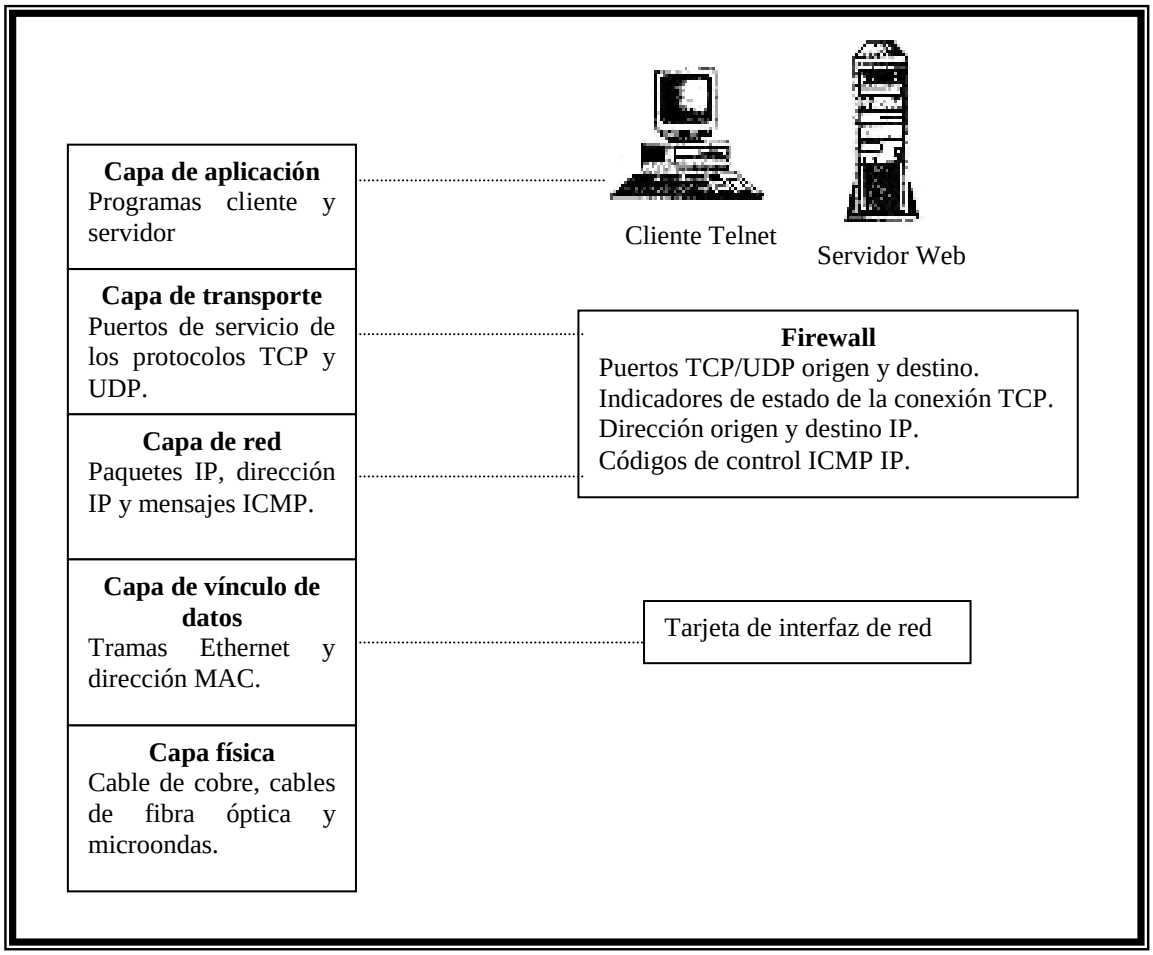

Fig. 3.02 Ubicación del firewall en el modelo de refencia TCP/IP

La idea general es que el administrador controle con mucho cuidado todo lo que sucede entre Internet y la máquina que se ha

conectado directamente a Internet. Sobre la interfaz externa, el administrador filtrará individualmente lo que procede del exterior y lo que sale de la máquina de forma tan precisa y explícita como sea posible, la figura 3.02 muestra la manera como un firewall de filtrado de paquetes funciona en las capas de red y transporte.

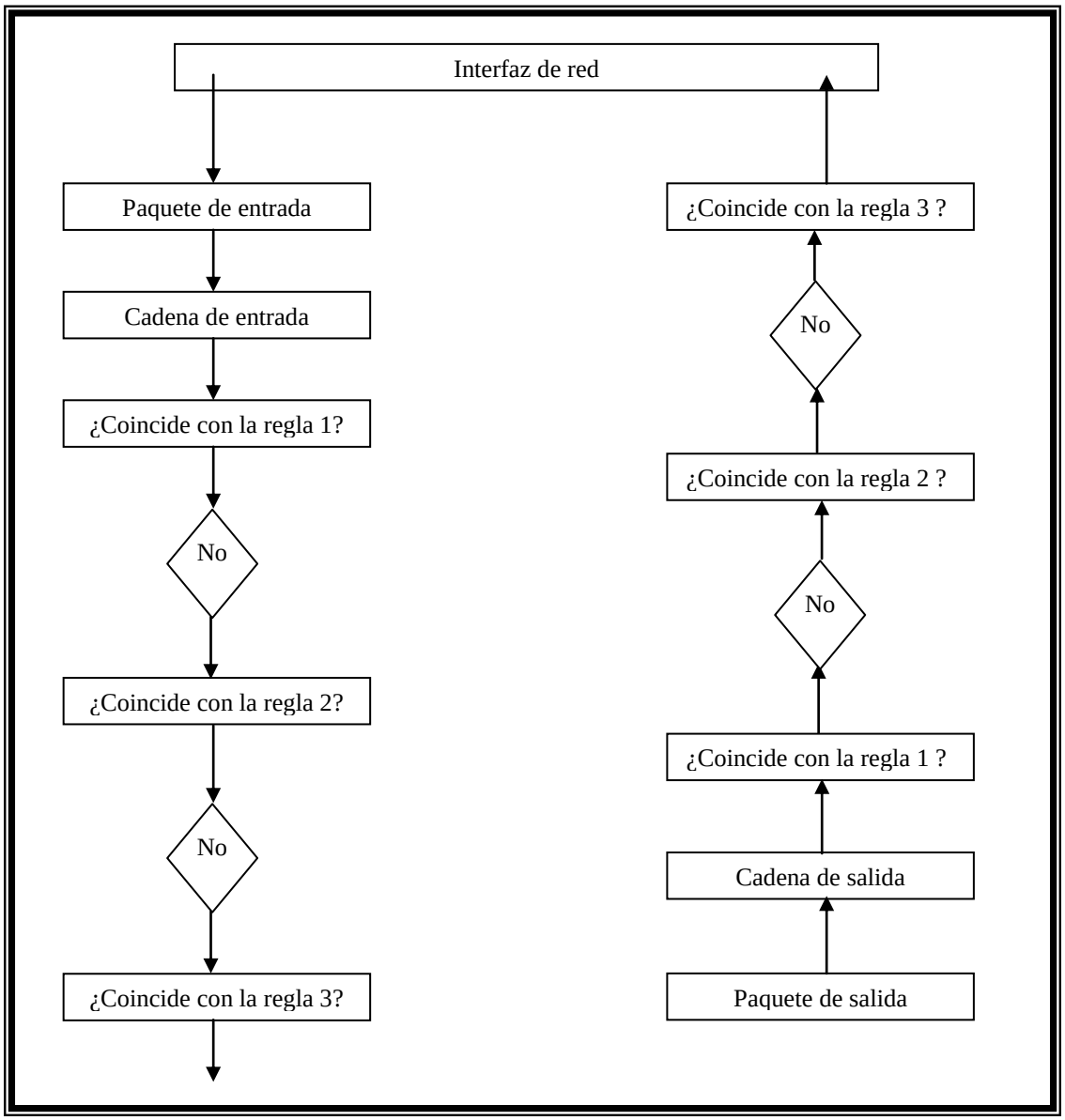

Fig. 3.03 Cadenas de entrada y salida

Para comprender de mejora manera la configuración de una máquina sencilla, puede ayudar pensar en la interfaz como un par E/S. El firewall filtra independientemente lo que entra y lo que sale a través de la interfaz, el filtrado de entrada y el filtrado de salida pueden tener reglas completamente diferentes. Las

listas de reglas que definen lo que puede entrar y lo que puede salir se llaman cadenas; las listas se llaman cadenas porque se compara un paquete con cada regla de la lista, una a una, hasta que se encuentra una coincidencia o la lista termine, como lo indica la figura 3.03.

Este mecanismo de seguridad no es infalible es sólo parte del problema una capa de todo el esquema de seguridad, pues no todos los protocolos de comunicación de aplicación se prestan para el filtrado de paquetes. Este tipo de filtrado pertenece a un nivel demasiado bajo como para permitir autenticación y control de acceso precisos, estos servicios de seguridad deben proporcionarse a niveles más altos, IP no tiene la capacidad de verificar que el emisor es quien o lo que dice ser, la única información de identificación disponible en este nivel es la dirección origen del encabezado del paquete IP, pero lamentablemente ni la capa de red ni la de transporte pueden verificar que los datos de la aplicación son correctos. Sin embargo, el nivel de paquete permite un control más preciso y sencillo sobre el acceso directo a un puerto, el contenido del paquete y los protocolos de comunicación correctos que se pueden establecer fácilmente o de forma adecuada en niveles superiores.

Sin filtrado a nivel de paquete, el filtrado de alto nivel y las medidas de seguridad proxy son inútiles o probablemente ineficaces. Hasta cierto punto deben basarse en la exactitud del protocolo de comunicación subyacente, cada nivel de la pila del protocolo de seguridad agrega otra pieza que los demás niveles no pueden ofrecer fácilmente.

Elección de una directiva predeterminada de filtrado de paquetes

Cada cadena del firewall tiene una directiva predeterminada y una colección de acciones a realizar en respuesta a tipos de mensajes específicos, cada paquete se compara uno a uno, con cada regla de la lista hasta que se encuentra una coincidencia, si el paquete no coincide con ninguna regla fracasa y se aplica la directiva predeterminada al paquete.

Hay dos perspectivas básicas para un firewall

- Denegar todo de forma predeterminada y permitir que pasen paquetes seleccionados de forma explícita.

- Aceptar todo de forma predeterminada y denegar que pasen paquetes seleccionados de forma explícita.

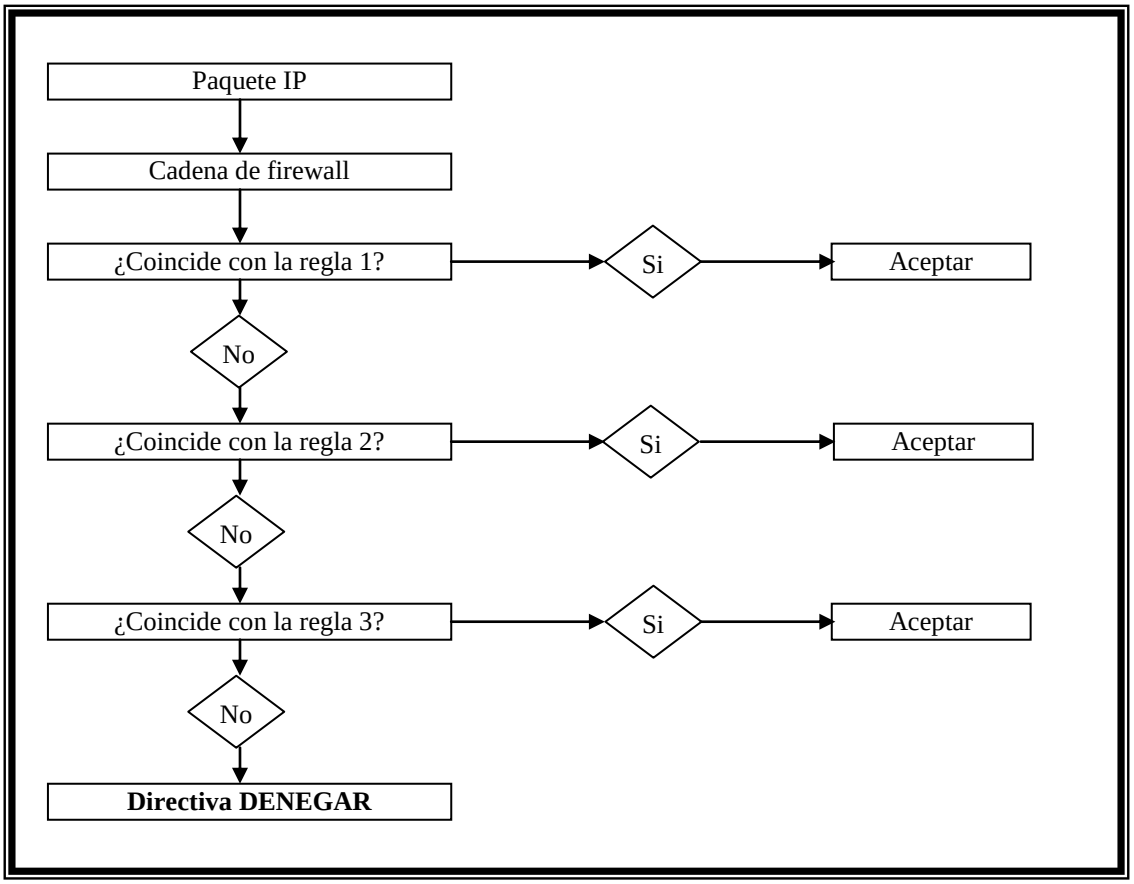

Fig. 3.04 Directiva denegar todo de forma predetermianda

La directiva de denegar todo es la que se ha escogido (figura 3.04), esta aproximación facilita la configuración de un firewall seguro, pero es necesario habilitar explícitamente cada servicio y la transacción de protocolo relacionada que se desea. La política de denegar todo requiere preparar el terreno para habilitar el acceso de Internet.

Rechazar frente a denegar un paquete

El mecanismo del firewall iptables ofrece la opción de rechazar o denegar los paquetes. Como se muestra en la siguiente figura, cuando se rechaza un paquete se descarta y se devuelve un mensaje de error ICMP al remitente. Cuando se deniega un paquete, simplemente se descarta el paquete sin ningún tipo de notificación al remitente.

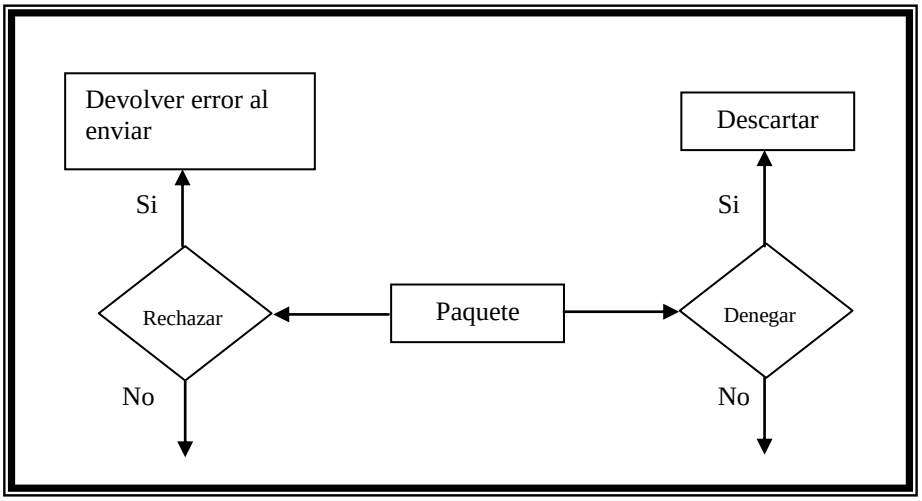

Fig. 3.05 Rechar un paquete frente a denegarlo

La denegación es casi siempre la mejor elección, hay tres razones para esto:

1. Enviar respuesta de error duplica el tráfico de red, la mayoría de paquetes se descartan porque son malévolos, no porque representan un intento inocente por acceder a un servicio que no se está ofreciendo.

2. Cualquier paquete al que responda se puede usar en un ataque denegación de servicio.

3. Cualquier respuesta, incluso un mensaje de error, ofrece información potencialmente útil a quien podría ser un hacker.

Para comprender más claramente como es el funcionamiento de un firewall de filtrado de paquetes, es necesario conocer varios aspectos sobre los cuales estos trabajan para realizar el filtrado, entre estos temas se tienen:

3.7.1 Como filtrar paquetes entrantes
3.7.2 Sondeos y exploraciones
3.7.3 Ataques por denegación de servicio
3.7.4 Como filtrar paquetes salientes

3.7.1 Como filtrar los paquetes entrantes

El lado de entrada del par E/S de la interfaz externa, es el más interesante a la hora de asegurara un sitio. Como se indicó anteriormente se puede filtrar basándose en la dirección origen, la dirección destino, el puerto origen, el puerto destino y el indicador de estado TCP, a continuación se explica en que consiste el filtrado de cada una de esta opciones.

3.7.1.1 Filtrado de dirección origen remota
3.7.1.2 Usurpamiento de dirección origen y direcciones ilegales
3.7.1.3 Bloquear sitios problemáticos
3.7.1.4 Limitar lo paquetes entrantes a aquello procedentes de los hosts remotos seleccionados

3.7.1.1 Filtrado de dirección origen remota

A nivel de paquete, el único medio de identificar el remitente del paquete es la dirección origen del encabezado del paquete, este hecho abre la puerta al spoofing, o usurpamiento de dirección origen, donde el remitente coloca una dirección incorrecta, en vez de la suya propia, en el campo origen. La dirección puede ser una dirección inexistente o puede ser una dirección legítima perteneciente a otra persona. Esto puede permitir varias formas desagradables de romper el sistema y hacerse pasar por el usuario mientras atacan otros sitios, fingiendo ser otra persona cuando están atacando o hacerle creer que es el origen de los mensajes entrantes.

3.7.1.2 Usurpamiento de dirección origen y direcciones ilegales

Hay seis clases principales de direcciones origen que siempre se deben denegar en la interfaz externa, estas direcciones son las de paquetes entrantes que dicen ser una de las siguientes direcciones:

- **Su dirección IP.-** Nunca se verán paquetes entrantes legales que indiquen proceder del mismo equipo. Como la dirección origen es la única información disponible, y ésta se puede modificar, es la única forma de usurpamiento que se puede detectar a nivel de filtrado de paquetes, lo paquetes entrantes que dicen proceder de la misma máquina pertenecen al

usurpamiento de direcciones, no es posible saber con certeza si otros paquetes entrantes proceden de donde dicen proceder.

- **Direcciones IP privadas de clase A, B y C.**- Un conjunto de direcciones en cada uno de los intervalos de las clases A, B y C son reservadas para su uso en LAN privadas, no se pueden usar en Internet. Como tales direcciones las pueden usar cualquier sitio de forma interna sin necesidad de comprar direcciones IP registradas, el servidor nunca debe ver paquetes entrantes procedentes de estas direcciones origen, los rangos de estas direcciones son:

- Las direcciones privadas de clase A, se asignan al intervalo de direcciones de 10.0.0.0 a 10.255.255.255.

- Las direcciones privadas de clase B, se asignan al intervalo de direcciones de 172.16.0.0 a 172.31.255.255.

- Las direcciones privadas de clase A, se asignan al intervalo de direcciones de 192.168.0.0 a 192.168.255.255.

- **Dirección IP multidifusión de clase D.-** Las direcciones IP en el intervalo de las clase D se reservan para su uso como direcciones destino cuando se participa en una difusión en una red multidifusión, como una difusión de sonido o de video. El intervalo comienza con la dirección 224.0.0.0 y termina en 239.255.255.255, el servidor nunca debería ver paquetes procedentes de estas direcciones origen.

- **Direcciones reservadas de la clase E.**- Las direcciones IP en el intervalo de la clase E se ha reservado para usos futuros y experimentales y no se asignan públicamente. El intervalo comienza en la dirección 240.0.0.0 y termina en 247.255.255.255, el servidor nunca debería ver paquetes de

estas direcciones origen y es muy probable que no lo haga, las redes de defensa e inteligencia son lo bastante buenas como para no perder sus paquetes.

- **Direcciones de interfaz de bucle invertido.**- La interfaz de bucle invertido es una interfaz de red privada, que usa el sistema Linux para servicios locales basados en red, en lugar de enviar el tráfico local a través del controlador de la interfaz de red, el sistema operativo toma un atajo a través de la interfaz de bucle invertido como una forma de mejorar el rendimiento. Por definición, el tráfico de bucle invertido apunta al sistema que lo generó, no sale fuera de la red. El intervalo de las direcciones de bucle invertido es 127.0.0.0 a 127.255.255.255. Normalmente se verá como 127.0.0.1, localhost o la interfaz de bucle invertido lo.

- **Direcciones de difusión mal formadas.-** Las direcciones de difusión son direcciones especiales que se aplican a todas las máquinas de una red, la dirección 0.0.0.0 es una dirección origen de difusión especial. Cuando se ve como la dirección origen en un paquete no de difusión regular, la dirección está falsificada.

3.7.1.3 Bloquear sitios problemáticos

Otro esquema común de filtrado de dirección, consiste en bloquear todos los accesos desde una máquina seleccionada, o de forma más normal, desde todo un bloque de direcciones IP de red. Si un sitio gana la reputación de ser un mal vecino de Internet, otros sitios tienden a bloquearlo sin excepciones.

A nivel individual es conveniente bloquear todos los accesos desde redes seleccionadas cuando los individuos de la red remota no hacen más que ocasionar problemas.

3.7.1.4 Limitar los paquetes entrantes a aquellos procedentes de los hosts remotos seleccionados

Puede que se quiera aceptar ciertas clases de paquetes entrantes procedentes sólo de sitios externos específicos o personas concretas, en estos casos las reglas del firewall definirán direcciones IP específicas o un intervalo limitado de direcciones origen IP desde las que se aceptarán estos paquetes.
Las primera clase de paquetes entrantes procede de servidores remotos que responden a las peticiones del usuario, aunque algunos servicios, como los servicios Web o FTP, se puede esperar que procedan de cualquier lugar, otros servicios tendrán una procedencia legítima sólo si proceden del ISP o de los hosts seguros elegidos, un ejemplo de esto son las respuestas del servidor de nombres de dominio DNS.

La segunda clase de paquetes entrantes son las que proceden de clientes remotos que acceden a los servicios que se ofrecen en el servidor. De nuevo, aunque algunas conexiones de servicio entrante, como la conexiones al servidor web, se puede esperar que procedan de cualquier sitio, otro servicios locales sólo se ofrecerán a unos pocos usuarios o amigos de confianza. Algunos ejemplos de servicios locales restringidos pueden ser telnet, ssh, ftp, finger, etc.

3.7.1.5 Filtrado de dirección destino local

El filtrado de paquetes entrantes basándose en la dirección destino no suele ser problemático, la tarjeta de interfaz de red ignora los paquetes habituales que no se dirigen a ella, la excepción son los paquetes de difusión, que se difunden a todos los hosts de la red.

La dirección 255.255.255.255 es la dirección destino de difusión general, esta se puede definir de forma más explícita como la dirección de red seguida por el número 255 en las tuplas restantes de dirección.

La dirección de difusión a destino 0.0.0.0 se parece a la situación de los paquetes punto a punto que decían proceder de la dirección origen de difusión mencionada anteriormente "Usurpamiento de direcciones origen y direcciones ilegales". En este caso, los paquetes de difusión se dirigen a la dirección origen 0.0.0.0 en vez de a la dirección destino 255.255.255.255, este es un intento de identificar el sistema como una máquina Linux.

3.7.1.6 Filtrado de puerto origen remoto

El puerto origen en los paquetes entrantes identifica el programa del host remoto que envía el mensaje, en general todas las peticiones entrantes desde clientes remotos a sus servicios siguen el mismo modelo, y todas las respuestas entrantes de servidores remotos al cliente local siguen un modelo diferente.
Las peticiones y las conexiones de entrada desde clientes remotos a los servidores locales tendrán un puerto origen del intervalo no privilegiado. Para el servidor web todas las conexiones entrantes al servidor web deberán tener un puerto origen entre 1024 y 65535.

Las respuestas entrantes de los servidores remotos con los que se ha conectado tendrán el puerto origen asignado al servicio particular, si se conecta a un sitio web remoto, todos los mensajes entrantes procedentes del servidor remoto tendrán el puerto origen establecido a 80, que es el número de puerto del servicio http.

3.7.1.7 Filtrado de puerto destino local

El puerto destino en los paquetes entrantes identifica el programa o el servicio del equipo al que se dirige el paquete, como ocurre con el puerto origen en general, todas las peticiones entrantes procedentes de clientes remotos a sus servicios siguen el mismo modelo, y todas las respuestas entrantes procedentes de servicios remotos a sus clientes locales siguen un modelo diferente.

Las peticiones entrantes y las conexiones entrantes procedentes de clientes remotos a los servidores locales establecerán el puerto destino al número de servicio asignado al servicio particular. Un paquete entrante dirigido a un servidor web local tendrá el puerto destino establecido a 80, que es el número del puerto del servicio http.

Las respuesta entrantes de los servidores remotos con los que se ha conectado tendrán el puerto destino en el intervalo no privilegiado, si se conecta a un sitio web remoto todos los mensajes entrantes deberán tener un puerto destino entre 1024 y 65535.

3.7.1.8 Filtrado del estado de la conexión TCP entrante

Las reglas de aceptación del paquete TCP entrante pueden hacer uso de los indicadores de estado de la conexión asociados con las conexiones TCP. Todas la conexiones TCP se adhieren al mismo conjunto de estados de conexión, los cuales difieren entre cliente y servidor debido al saludo de tres vías que se realiza durante el establecimiento de la conexión.

Las paquetes TCP entrantes procedentes de clientes remotos tendrán el indicador SYN activado en el primer paquete recibido como parte del saludo de establecimiento de la conexión de tres

vías. La primera petición de conexión tendrá el indicador SYN activado, pero no el indicador ACK. Todos los paquetes entrantes después de la primera petición de conexión tendrán sólo el indicador ACK activado, las reglas del firewall del servidor local permitirán paquetes entrantes, sin tener en cuenta el estado de los indicadores SYN y ACK.

Los paquetes entrantes procedentes de servidores remotos siempre serán respuestas a la petición de conexión inicial que comienza en el programa cliente local, cada paquete recibido desde un servidor remoto tendrá el indicador ACK activado. Las reglas del firewall cliente local solicitarán que todos los paquetes entrantes procedentes de servidores remotos tengan el indicador ACK activado, los servidores legítimos no intentarán iniciar conexiones a programas cliente.

3.7.2 Sondeos y exploraciones

Un sondeo es un intento de conectar o de obtener una respuesta desde un puerto de servidor individual. Una exploración es una serie de sondeos a un conjunto de diferentes puertos de servicio, las exploraciones suelen estar automatizadas.

Por desgracia los sondeos y las exploraciones rara vez son inocentes, es más probable que sea la fase inicial de recopilación de información, que busca debilidades interesantes antes de lanzar un ataque de un hacker. Las herramientas de exploración automatizadas son de uso generalizado y los esfuerzos coordinados por grupos de hackers son habituales.

Existen diversos métodos de exploraciones, entre estos se tienen los siguientes:

3.7.2.1 Exploraciones de puerto generales
3.7.2.2 Exploraciones de puerto dirigidas
3.7.2.3 Destinos comunes en los puertos de servicio

3.7.2.1 Exploraciones de puerto generales

Las exploraciones de puerto generales son sondeos indiscriminados a lo largo de un bloque de puertos de servicio, probablemente todo el intervalo. Existen varias herramientas que hacen este trabajo entre las cuales se puede mencionar mscan, sscan y nscan.

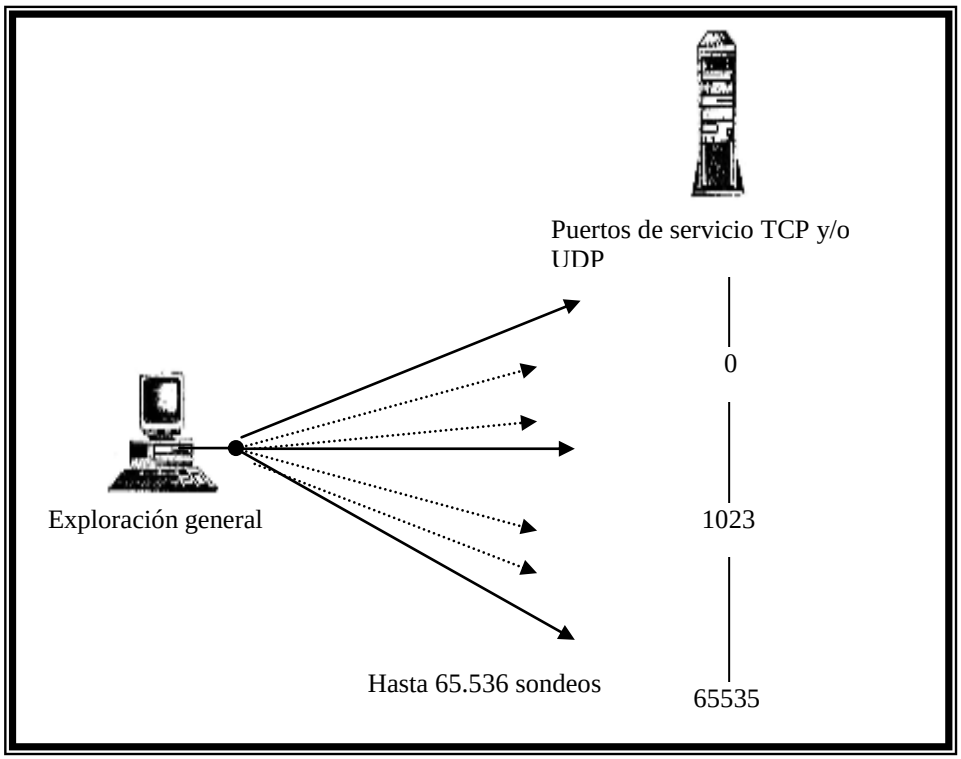

Fig. 3.06 Exploración de puerto general

3.7.2.2 Exploraciones de puerto dirigidas

Las exploraciones de puerto dirigidas buscan debilidades específicas. Las herramientas más nuevas y sofisticadas intentan identificar el hardware, el sistema operativo y las versiones de software, estas herramientas están diseñadas para anular por completo las debilidades conocidas de objetivos específicas.

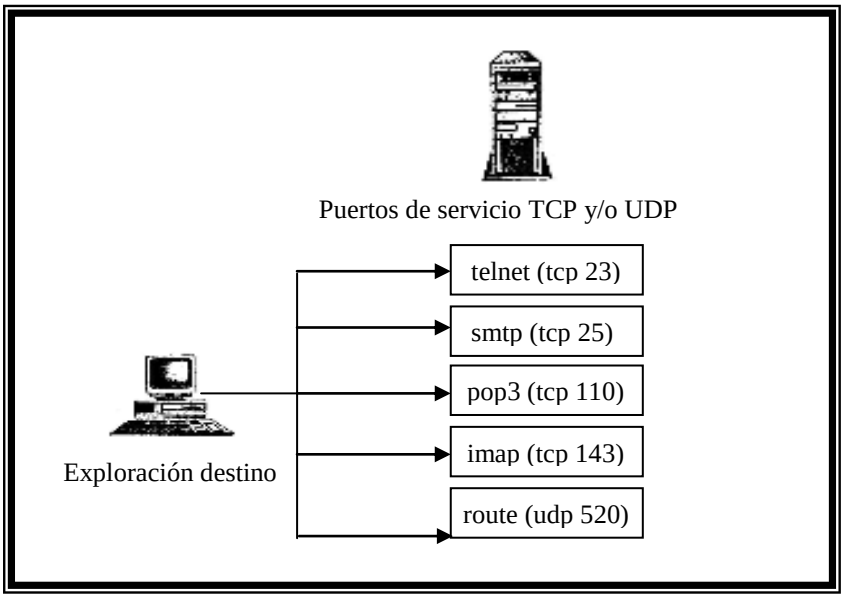

Fig. 3.07 Exploración de puerto dirigida

3.7.2.3 Destinos comunes en los puertos de servicio

Los destinos comunes se suelen sondear y explorar de forma individual, el hacker puede estar buscando una debilidad específica, como un servidor de correo inseguro o un demonio portmap de RPC abierto.

A continuación se mencionan unos pocos puertos comunes:

- Los paquetes entrantes procedentes del puerto 0 reservado son siempre falsos, este puerto no se usa de forma legítima.

- Los sondeos de puertos TCP del 0 al 5 son una firma del programa sscan.

- telnet (23/tcp), smtp (25/tcp), pop3 (110/tcp), sunrpc (111/tcp), imap (143/tcp), snmp (161/tcp), route (520/udp) son los puertos destino favoritos y representan algunas de las debilidades más importantes del sistema.

3.7.3 Ataques por denegacion de servicio

Los ataques por denegación de servicio se basan en la idea de inundar un sistema con paquetes de forma que afecte o degrade seriamente la conexión de Internet, inmovilizando los servidores locales hasta el extremo de no poder atender las peticiones legítimas o en el peor de los casos rompiendo totalmente el sistema. Los dos resultados más comunes son mantener al sistema demasiado ocupado para hacer nada útil e inmovilizar los recursos críticos del sistema.

No es posible protegerse completamente contra los ataques por denegación de servicio, toman tantas formas diferentes como permite la imaginación del hacker. Cualquier cosa que produzca una respuesta del sistema, cualquier cosa que produzca peticiones de recursos en el sistema, cualquier cosa que induzca a un sitio remoto a dejar de comunicarse con el usuario, todo se puede usar en un ataque denegación de servicio.

Sin embargo, estos ataques suelen implicar algunos de los diferentes modelos clásicos, incluyendo inundación SYN TCP, inundación ping, inundación UDP y bombas de redirección de enrutamiento ICMP.

Entre los tipos de ataques por denegación de servicio más comunes, se tiene los siguientes:

3.7.3.1 Ataques masivos SYN TCP

Un ataque de inundación SYN TCP consume los recursos del sistema hasta que no es posible establecer más conexiones TCP entrantes, el ataque hace uso del protocolo de saludo de tres vías durante el establecimiento de la conexión, junto con usurpamiento de la dirección origen IP.

El atacante usurpa la dirección origen e inicia una conexión TCP, el atacante envía un mensaje SYN. La máquina responde enviando una confirmación SYN-ACK; sin embargo en este caso, la dirección a la que contesta el usuario no es la dirección del atacante. La etapa final del establecimiento de conexión TCP, que consiste en recibir un ACK como respuesta nunca tendrá lugar, en consecuencia se consumen los recursos finitos de la conexión de red, la conexión permanece en un estado semiabierto hasta que la conexión alcanza su tiempo de espera. El hacker inunda el puerto con un petición de conexión tras otra, más rápido de lo que los tiempos de espera TCP liberan los recursos. Si esto continúa, todos los recursos están en uso y no se podrán aceptar más peticiones de conexión entrantes, si el objetivo es el puerto smtp, no se podrá recibir correo electrónico.

Existen varias ayudas para usuarios Linux, la primera es el filtrado de direcciones origen descrito anteriormente, pero no hay garantía de que la dirección usurpada pertenezca a las categorías que se pueden anticipar y filtrar. La segunda consiste en compilar

el núcleo del Linux con las cookies SYN habilitadas este es un retardo específico para la inundación SYN.

3.7.3.2 Ataques masivos ping

Cualquier mensaje que provoque una respuesta de la máquina se puede usar para degradar la conexión de red obligando al sistema a gastar la mayor parte del tiempo respondiendo, el mensaje de petición de eco ICMP que se ha enviado mediante ping suele ser un posible culpable.

Ping es una herramienta de redes muy útil, el entorno de Internet actual recomienda deshabilitar el ping entrante o al menos limitar de forma severa de quien puede aceptar peticiones de eco.

3.7.3.3 Ataques masivos UDP

El protocolo UDP es especialmente útil como herramienta denegación de servicio, al contrario que TCP, UDP es sin estado no se incluyen mecanismos de control de flujo, no hay indicadores de estado de conexión, no se usan los números de secuencia del datagrama, no se mantiene información sobre el paquete que se espera a continuación, es relativamente fácil mantener a un sistema tan ocupado respondiendo a sondeos UDP entrantes que no quede ancho de banda para el tráfico de red legítimo.

Como los servicios UDP son inherentemente menos seguros que los servicios TCP, muchas veces es mejor deshabilitar todos los puertos UDP que no son absolutamente necesarios. Cuando se crea el script del firewall se restringe el tráfico UDP sólo a aquellos host que proporcionan servicios UDP necesarios.

3.7.3.4 Bombas de redirección ICMP

El mensaje de redirección de ICMP tipo 5 indica al sistema destino que cambie sus tablas de enrutamiento por una ruta más corta. Si se ejecuta route y puede redireccionar mensajes, es posible para un hacker engañar el sistema para que piense que la máquina del hacker es una de las máquinas locales o una de las máquinas del ISP, o incluso engañar al sistema para que lance todo el tráfico a algún otro host remoto.

3.7.4 Como filtrar paquetes salientes

Si el entorno representa un entorno seguro, filtrar tanto los paquetes salientes es tan importante como filtrar los paquetes entrantes, el sistema no responderá mensajes entrantes si no pasan a través del firewall. Este filtrado también protege a otras personas y al administrador de posibles errores existentes en la máquina.

Filtrar los paquetes salientes también permite ejecutar servicios LAN sin perder paquetes locales en Internet, a donde no pertenecen estos paquetes, no es sólo cuestión de no permitir el acceso externo a los servicios LAN, es también una cuestión de no difundir información del sistema local en Internet.

Al igual que en el caso del filtrado de paquetes entrantes, el filtrado de paquetes salientes trabaja sobre diversos aspectos, entre los cuales se tiene:

3.7.4.1 Filtrado de dirección origen local

Filtrar los paquete salientes basándose en la dirección es fácil, para un pequeño sitio o un equipo independiente conectado a Internet, la dirección origen es siempre la dirección IP del equipo del usuario cuando funciona normalmente. No hay razón para permitir que un paquete saliente tenga otra dirección origen.

3.7.4.2 Filtrado de dirección destino remota

Al igual que los paquetes entrantes, se puede querer permitir que ciertas clases de paquetes salientes sólo se dirijan a redes remotas o máquinas individuales específicas, en estos casos las reglas del firewall definirán direcciones IP concretas o un intervalo restringido de direcciones IP destino donde se permitirán estos paquetes.

La primera clase de paquetes salientes a filtrar mediante la dirección son los paquetes destinados a servidores remotos con los que se ha contactado, aunque puede esperar que algunos paquetes, como los destinados a servidores Web o FTP, se dirijan a cualquier lugar de Internet, otros servicios remotos sólo los ofrecerán de forma legítima el ISP o hosts seguros elegidos de forma concreta.

La segunda clase de paquetes salientes que se deben filtrar por la dirección destino son los paquetes destinados a clientes remotos

que acceden a un servicio que ofrece el servidor. De nuevo, aunque algunas conexiones de servicio salientes pueden ir a cualquier lugar, como las respuestas desde el servidor Web local, otros servicios locales sólo se ofrecerán a unos pocos sitios remotos seguros. Algunos ejemplos de estos servicios son telnet, ssh, etc. las reglas del firewall no solamente denegarán las conexiones entrantes generales a estos servicios, sino que tampoco permitirán respuestas salientes de estos servicios a nadie.

3.7.4.3 Filtrado de puerto origen local

Definir explícitamente los puertos de servicio que se pueden usar en nuestro extremo para conexiones salientes tiene dos propósitos, el primero para los programas cliente y el segundo para los programas servidor. Especificar los puertos origen que se permiten para las conexiones salientes ayuda a asegurar que los programas se comportan correctamente, y protege a otras personas de cualquier tráfico de red local que no pertenezca a Internet.

Las conexiones salientes desde los clientes locales casi siempre se originan desde un puerto origen no privilegiado, restringir los clientes a los puertos no privilegiados en las reglas del firewall ayuda a proteger a otras personas de posibles errores en su extremo, asegurando que los programas cliente se comportan como se espera.

Los paquetes salientes desde los programas del servidor locale se originarán siempre desde el puerto de servicio asignado, si se restringen los servidores a los puertos asignados en el nivel de firewall, se asegura que los programas de servidor funcionan correctamente a nivel de protocolo. Lo más importante es que todo esto sirve de ayuda para proteger todos los servicios privados

de red local que se estén ejecutando desde un acceso exterior, también ayuda a proteger los sitios remotos, para que el tráfico de red que debería permanecer confinado a los sistemas locales, no ocasione molestias.

3.7.4.4 Filtrado de puerto destino remoto

Los programas cliente locales están diseñados para conectarse a servidores de red que ofrecen sus servicios desde los puertos de servicio asignados, desde esta perspectiva restringir los clientes locales para que sólo puedan conectar a los puertos de servicio asociados asegura la exactitud del protocolo. Restringir las conexiones de los clientes a puertos destino específicos también sirve para un par de propósitos, en primer lugar ayuda a vigilar contra los programas de red privados y locales que de forma inadvertida intentan acceder a servidores en Internet; segundo hace todo lo posible para no permitir los errores salientes, las exploraciones de puerto y otras acciones incorrectas que se originan en el sitio.

Los programas servidores locales casi siempre participarán en las conexiones que se originan desde puertos no privilegiados , las reglas del firewall restringen el tráfico saliente de servidores sólo a puertos de destino no privilegiados.

3.7.4.5 Filtrado saliente de estado de la conexión TCP

Las reglas de aceptación de paquetes TCP salientes pueden hacer uso de los indicadores de estado de la conexión asociados con las conexiones TCP, igual que lo hacen las reglas de entrada, todas las conexiones TCP se adhieren al mismo conjunto de estados de conexión, que es diferente para el cliente y el servidor.

Los paquetes TCP salientes procedentes de clientes locales tendrán el indicador SYC definido en el primer paquete que se envíe como parte del saludo de establecimiento de conexión de tres vías. La petición de conexión inicial tendrá definido el indicador SYN, pero no el indicador ACK, todos los paquetes salientes posteriores a l a primera petición de conexión solo tendrán definido el indicador ACK. Las reglas del firewall del cliente local permitirán los paquetes salientes con el indicador SYN o el indicador ACK activado.

Los paquetes salientes procedentes de servidores locales serán siempre respuestas a una petición inicial de conexión iniciada desde un programa cliente remoto, cada paquete que se envía desde sus servidores tendrá activo el indicador ACK, las reglas de firewall de servidor local solicitarán que todos los paquetes salientes de los servidores tengan activo el indicador ACK.

3.7.5 Creación e instalación del firewall de Linux RedHat 7.1 IPTABLES

El nuevo kernel de Linux, soporta un nuevo mecanismo para construir firewalls de filtrado de paquete, este nuevo mecanismo el cual es controlado por una herramienta llamada **iptables** es muy seguro. A través de este firewall se pueden bloquear o detectar muchos ataques que no eran detectables en versiones anteriores de Linux, además bloquea ataques DoS debido a que analiza los paquetes que envían los usuarios y limita el tráfico de los mismos, además almacena cada conexión que pasa a través de él.

Esta nueva tecnología implica que si un paquete extraño prueba ingresar a la red pretendiendo ser parte de una conexión existente, IPTABLES puede consultar su lista de conexiones las

cuales están almacenadas en memoria y si encuentra que el paquete no es igual a ninguno, este borra el paquete y elimina la conexión.

3.7.5.1 ¿Cuál es la Política de Seguridad del Firewall?

La política de seguridad define cuales servicios serán explícitamente permitidos o denegados, cómo estos servicios son usados y cuales son las excepciones para el uso de los mismos, pues el diseño del firewall se basará en la reglas establecidas en la política de seguridad.

Cada regla en la política de seguridad de la red debe ser implementada en el firewall, generalmente un firewall usa uno de los siguientes métodos:

a) *Negación preestablecida:* "lo que no está permitido expresamente, está prohibido"

La negación preestablecida tiene sentido desde el punto de vista de la seguridad porque es una postura de falla segura. Acepta que lo que no se conoce puede hacer daño es la opción más segura, pues por omisión se prohibe todo, para determinar los que se va a permitir .

Se realiza un análisis de los servicios que se brindan y se hace un balance comparando los efectos que tendrá en la seguridad contra las necesidades de los usuarios y de acuerdo a esta análisis se pondrá en disponibilidad o no dicho servicio.

b) ***Permiso preestablecido:*** **"lo que no está prohibido expresamente, está permitido"**

En el permiso preestablecido por omisión todo está permitido y se irá prohibiendo ciertas acciones o servicios problemáticos específicos conforme sea necesario.

En este caso es necesario especificar que es lo que los usuarios no pueden hacer, lo cual no es una postura de falla segura, pues si el administrador no se da cuenta a tiempo de que un servicio está fallando se pueden tener serios problemas de seguridad.

3.7.5.2 Política de seguridad del firewall de la Universidad Técnica de Norte

La política de seguridad del firewall de la Universidad Técnica del Norte es *n**egación preestablecida:*** **"lo que no está permitido expresamente está prohibido",** pues se ha decido tomar esta medida debido a que existen en la actualidad muchos personas mal intencionadas que se aprovechan de la menor debilidad que pueda tener un sistema operativo para atacar un equipo y ponerlo en contra de otros o simplemente para dejarlo fuera de servicio.

Al tomar esta medida se está protegiendo tanto al servidor como a la red de ataques. Debido a la presencia de servicios que se prestan para ejecutar este tipo de actividades como es el caso del ping de la muerte, inundaciones ICMP, etc, a través del firewall se habilitarán únicamente aquellos servicios necesarios para los usuarios de la Universidad, que son quienes se beneficiarán del uso de Internet y de la red de datos REDUTN.

3.7.5.3 Topología

Todos los servidores deben ser configurados para bloquear al menos los puertos que están sin uso, incluso si estos no están en el servidor firewall, este es un requisito para mejorar la seguridad. Imaginemos que alguien gana acceso al servidor firewall gateway: si el servidor no está configurado para bloquear los puertos sin uso, hay un gran riesgo en la seguridad. Esto es igual para conexiones locales, usuarios sin privilegios ganan acceso desde dentro del red al servidor y también pueden ocasionar problemas.

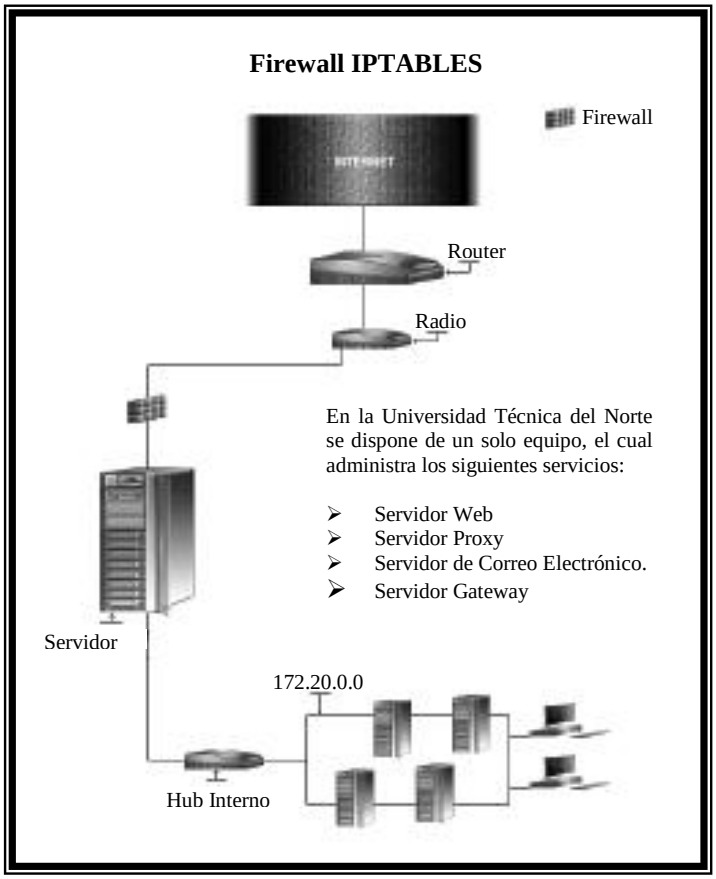

Fig. 3.08 Firewall UTN

La configuración de un servidor se realiza en función de los servicios que va a prestar, en el caso de la Universidad Técnica del Norte, se dispone de un solo Servidor en el cual se encuentran instalados servicios tales como: http, correo

electrónico, DNS, gateway, proxy, entre los más importantes, la figura 3.08 es una representación gráfica del diseño.

Esta implementación no es adecuada, debido a que el servidor está sobrecargado al tener que brindar todos los servicios sin ninguna ayuda adicional, además se pueden generar varios problemas de seguridad al tener unidos en el mismo servidor todos los servicios, pues cada uno de estos tiene debilidades que en algún momento podrían afectar a otros servicios.

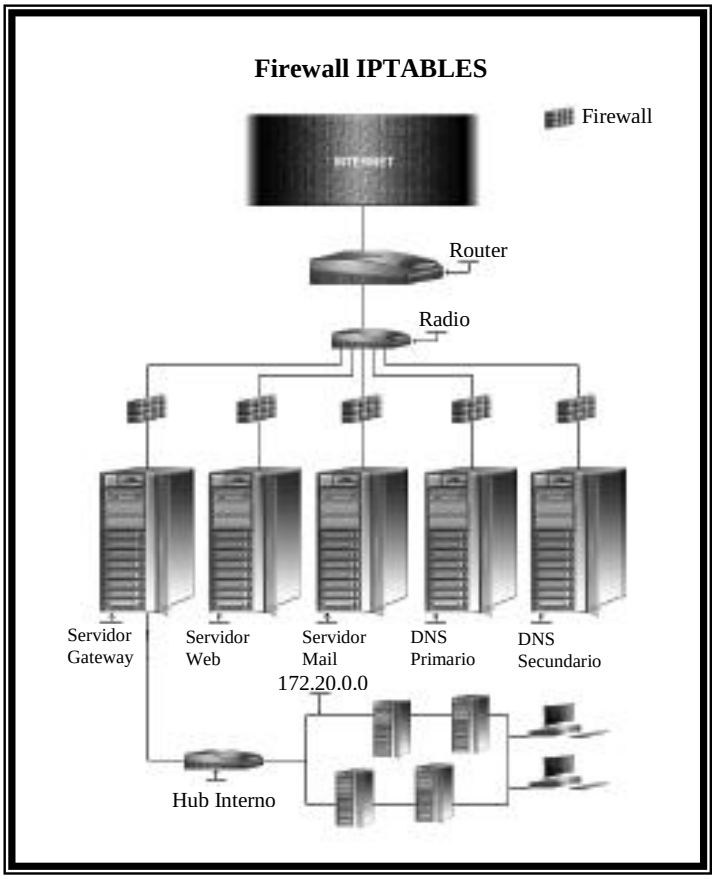

Fig. 3.09 Topología de firewall apropiada

Es recomendable tener un servidor por separado para cada uno de los servicios que se desea brindar, con esto se obtienen ciertas ventajas como: mejorar la velocidad, debido a que cada servidor únicamente se dedicará a realizar el trabajo que se le ha asignado y no tendrá que realizar ninguna otra tarea adicional; mejorar el control ya que se sabe exactamente que servicios están habilitados en cada servidor, con lo cual fácilmente se pueden

detectar problemas en servicios específicos; mejorar la seguridad ya que configurará el firewall apropiado para cada servidor dependiendo del servicio que se desee brindar a través del mismo. Por lo tanto una distribución apropiada de servidores debería similar a la figura 3.09.

Todos los pasos y pre-requisitos necesarios para poner en funcionamiento el firewall de Linux RedHat 7.1 se encuentran en el Anexo 2.

3.8 OPENSSL

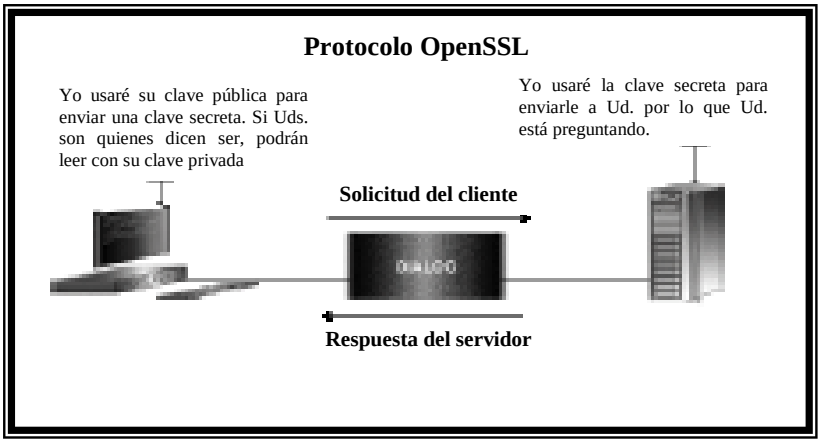

Fig. 3.10 Funcionamiento de OpenSSL

Mucho software como IMAP&POP, SSH, Samba, Sendmail, FTP, Apache y otros que solicitan un nombre de usuario para autenticarlo antes de permitrle el acceso a dicho servicio, por default transmitien el id de usuario y el password en texto plano. Alternativamente, mecanismos de encriptación como SSL permiten tener transacciones seguras. Una vez que OpenSSL ha sido instalado en el servidor Linux se puede usarlo con otras aplicaciones que tenga soporte para esta funcionalidad.

3.8.1 Ventajas de la criptografía

Las principales ventajas que se obtienen al usar teconología de encriptación son:

a) Confidencialidad de Datos

Cuando un mensaje es encriptado, un algoritmo lo convierte en texto cifrado que oculta el contenido del mensaje, el cual puede ser enviado a través de un mecanismo público a otro sitio y luego volverse a convertir el texto entendible por otro usuario. Este proceso invlucra una clave secreta qu es usada para encriptar y luego desencriptar los datos, sin la clave secreta los datos encriptados no son útiles para nada.

b) Integridad de Datos

Un cryptographic checksum, llamado código de autenticación de un mensaje (MAC), puede ser calculado sobre un texto arbitario dado por el usuario para proteger la integridad de los datos. El resultado (texto y MAC) son entonces enviados al receptor quien puede verficar la MAC que viene con el mensaje recalculando la MAC para el mensaje usando la clave secreta apropiada y verificando la igualdad con la MAC enviada.

c) Autenticación

La identificación personal es otro uso de la criptografía, donde el usuario conoce un secreto el cual puede ser usado para identificar su autenticidad.

3.8.2 Configurar OpenSSL

Los archivos de configuración de OpenSSL son los siguientes:

- /usr/share/ssl/openssl.cnf
- /usr/share/ssl/misc/sign.sh

Los archivos de configuración con sus respectivas opciones se encuentran en el Anexo 3.

3.8.3 Algunos usos del software OpenSSL

El software OpenSSL puede ser usado para:

- Crear propios Certifying Authority Server
- Creación de llaves de parámetros RSA, DH y DSA.
- Creación de certificados X.509, CSRs y CRLs.
- Encriptación y Desencriptación de mensajes.
- Tests cliente y servidor SSL/TLS.
- Provee confidencialidad de datos, integridad, autenticación durante las transmisiones de los usuarios.

3.8.3.1 Herramientas Administrativas de OpenSSL

Después de haber configurado el software y una vez que el programa esté ejecutándose apropiadamente, se procederá a realizar un ejemplo del uso de esta herramienta, para lo cual se creará un certificado para el Servidor Web Apache.

3.8.3.1.1 Qué implica la Certificación de un Sitio Web

Un sitio Certificado entrega la confianza y la seguridad a los usuarios de que la información que intercambian con el servidor no será interceptada o mal empleada.

Un Sitio Certificado ofrece para quienes visitan un sitio web la posibilidad de acceder a un perfil completo (legal y comercial) del sitio que están visitando, validando que su presencia en internet está soportada por una empresa legalmente constituida y para quienes son titulares de un sitio web el valor agregado que les da el ser identificados como organizaciones confiables y seguras porque ofrecen a sus clientes y/o usuarios:

- Información general de la empresa (perfil).
- Un canal de comunicación seguro (certificado SSL).
- Políticas de privacidad para el manejo de la información personal (opcional).
- Validación de sus prácticas comerciales (opcional)

3.8.3.1.2 Qué es un Certificado SSL

SSL (Secure Sockets Layer) es el protocolo de comunicación segura más conocido y usado actualmente, es como un túnel que protege a toda la información enviada y recibida. Cuando una comunicación esta asegurada mediante un certificado SSL la información que se protege es la siguiente:

- El URL del sitio
- El contenido del sitio
- El contenido de cualquier forma transmitida

- Los "cookies" enviados del browser al servidor
- Los "cookies" enviados del servidor al browser; y,
- El contenido de las cabeceras de los http.

3.8.3.1.3 Cuál es la vigencia de la certificación de un sitio web

Al igual que los certificados personales, un certificado para servidor se expide por regla general por un período de un año, al final de ese período deberá solicitarse uno nuevo, siguiendo el mismo procedimiento que para su emisión.

Los pasos para la creación de un certificado SSL para Apache se encuentran en el Anexo 4.

3.8.4 Asegurando OpenSSL

A continuación se explican unas medidas de seguridad que se deben tomar para asegurar los principales archivos de configuración de OpenSSL.

3.8.4.1 Cambiando los permisos por default de las Claves de OpenSSL

Hacer las claves "lectura y escritura" solamente para el super-usuario root. Esto es importante para que nadie pueda hacer touch de estos archivos.

```
[root@utn /]# chmod 750 /usr/share/ssl/private/
[root@utn /]# chmod 400 /usr/share/ssl/certs/ca.crt
[root@utn /]# chmod 400 /usr/share/ssl/certs/www.utn.edu.ec.crt
[root@utn /]# chmod 400 /usr/share/ssl/private/ca.key
[root@utn /]# chmod 400 /usr/share/ssl/private/www.utn.edu.ec.key
```

3.9 OPENSSH

Muchos servicios de red como telnet, ftp, rsh, rlogin y rexec son vulnerables a las escuchas electrónicas, ello representa un grave problema, ya que incluso en un entrono de red cerrado, debe existir como mínimo un medio seguro para ver archivos, establecer permisos, ejecutar los scripts del shell, etc.

Para evitar que deteminadas personas capturen el tráfico diario de la red, es conveniente instalar Secure Shell (ssh), el cual es un sistema de inicio de sesión seguro y un buen sustituto de telnet, rlogin, rsh, rcp, etc.

SSH (Secure Shell) es un programa para conectarse a otro equip a través de una red, para ejecutar comandos en una máquina remota y para mover archivos de una máquina a otra, proporciona exhaustiva autenticación y comunicaciones seguras en redes no seguras.

3.9.1 Configurando OpenSSH

Los archivos de configuración de OpenSSH son los siguientes:

- /etc/sssh/ssh_config
- /etc/ssh/sshd_config
- /etc/pam.d/sshd
- /etc/rc.d/init.d/sshd

Los archivos de configuración con sus respectivas opciones se encuentran en el Anexo 5.

3.9.2 Configuración de OpenSSH para un usuario

Después de haber configurado el cliente, el servidor e inicializado el servicio ssh, es momento de crear una nueva clave pública y privada para un usuario establecido, el cual tendrá una conexión segura.

El archivo $HOME/.ssh/authorized_keys2 lista las claves públicas a las cuales se les permite el acceso. Cuando el usuario ingresa, el programa ssh dice al servidor cual es el par de clave que se debe usar para la autenticación. El servidor chequea si la clave está permitida, y si es así, envia al usuario un challenge, un número randómico, encriptado el cual será usado por la clave pública del usuario. El challenge únicamente puede ser desencriptado usando la propia llave privada. Entonces el cliente desencripta el challenge usando la llave privada, probando al servidor que conoce la llave privada sin necesidad de conocer al servidor.

Paso 1

Crear una nueva llave pública y privada para un usuario, para lo cual se ejecuta el siguiente comando en la máquina LOCAL.

```
[root@utn /]# su prueba
[prueba@utn /]$ ssh-keygen -d
Generating DSA parameter and key.
Enter file in which to save the key (/home/prueba/.ssh/id_dsa):
Created directory '/home/prueba/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/prueba/.ssh/id_dsa.
Your public key has been saved in /home/prueba/.ssh/id_dsa.pub.
The key fingerprint is:
1f:af:aa:22:0a:21:85:3c:07:7a:5c:ae:c2:d3:56:64 prueba@utn
```

Paso 2

Copiar la clave pública local id_dsa.pub desde el directorio /home/prueba/.ssh al mismo directorio en el equipo remoto con el nombre de authorized_keys2.

La idea general del funcionamiento de OpenSSH se presenta en la figura 3.10.

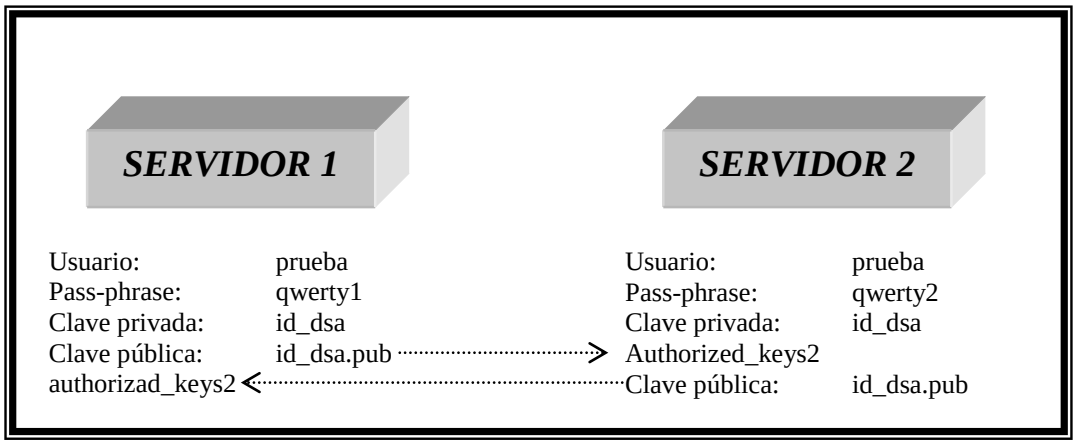

Fig. 3.11 Funcionamiento de OpenSSH

A continuación se presenta un ejemplo de una conexión ssh, la cual provee una comunicación encriptada entre dos hosts en una red insegura. Este programa realiza un ingreso seguro a una máquina remota y ejecuta comandos desde ahí, este reemplaza a telnet, rlogin, rcp, rdist y rsh.

```
[root@utn1 /]# ssh -l prueba utn.edu.ec
prueba@utn.edu.ec's password:
Last login: Tue Oct 19 1999 18:13:00 -0400 from utn1
No mail.
[prueba@utn /]$
```

Donde <prueba> es el nombre de usuario que tiene privilegios de conexión y <utn.edu.ec> es el nombre del equipo, en algunas ocasiones se usa la dirección IP que es similar al nombre del equipo.

3.9.3. Algunos usos de OpenSSH

OpenSSH puede ser usado para:

- Reemplazar programas como: telnet, rlogin, rsh, rdist y rpc.
- Hacer backups seguros sobre la red.
- Ejecutar comandos remotos.
- Accesar a recursos coorporativos sobre Internet con seguridad.
- Transferir archivos remotamente de manera segura.

3.10 LINUX sXid

Los archivos SUID y SGID son especiales, llevan sus permisos de propiedad en lugar de los permisos del usuario que los está ejecutando. Si los ataques pueden introducirse en los archivos SUID o SGID, pueden potencialmente conseguir acceso de raíz y poner en peligro la seguridad del sistema.

Por esta razón para reducir el riesgo que pueden causar este tipo de archivos, previamente se procedió a remover los archivos con bits 's' que sean programas de propiedad del usuario root y que no requieren de este privilegio, pero en el futuro pueden existir archivos que se configuren con el bit 's' habilitado sin recibir notificación alguna de que esto haya sucedido y por consiguiente la seguridad nuevamente estaría comprometida.

sXid es un programa de monitoreo de archivos suid/sgid, diseñado para ejecutarse regularmente por el demonio cron. Básicamente este sigue la pista de cualquier cambio en los archivos y carpetas s[ug]id, si se detecta la existencia de uno o más archivos con estas características entonces se reportarán los cambios en un formato de fácil entendimiento para el

administrador a través de email o en la línea de comandos. sXid automáticamente ejecutará la tarea de encontrar archivos SUID/SGID en el servidor y enviará el reporte al administrador, una vez instalada esta herramienta no hay necesidad de preocuparse por este trabajo.

3.10.1 Compilando y Optimizando sXid

Antes de proceder con la instalación y optimización de sXid se debe tener la precaución de estar trabajando con el usuario root para no tener problemas de privilegios durante este procedimiento.

Paso 1

Una vez que se ha obtenido el código fuente del programa se debe copiar dentro de /var/tmp y ubicarse dentro de este directorio para proceder con la instalación.

```
[root@utn /]# cp sxid_version.tar.gz /var/tmp
[root@utn /]# cd /var/tmp
[root@utn tmp]# tar xzpf sxid_version.tar.gz
```

Paso 2

Ubicarse dentro del nuevo directorio creado por sXid para configurar y optimizar la herramienta.

- Para ubicarse dentro de la nueva carpeta ejecutar el comado:

```
[root@utn tmp]# cd sxid-4.0.0.1/
```

- Para configurar y optimizar sXid:

```
CFLAGS=”-O3 -march=i686 -mcpu=i686 -funroll-loops -fomit-frame-pointer” \
./configure \
--prefix=/usr \
--sysconfdir=/etc \
--mandir=/usr/share/man
```

Paso 3

Ahora se procede a instalar sXid en el servidor, para lo cual debe ejecutar el siguiente comando:

```
[root@utn sxid-4.0.0.1# make install
```

Este comando configura el software para asegurar que el sistema tiene todas las librerías necesarias, compila todos los archivos fuente dentro de binarios ejecutables e instala los binarios y cualquier archivo de soporte adicional dentro de la ubicación apropiada.

Paso 4

Una vez instalado, compilado y optimizado el software la tarea ha finalizado, para liberar un poco de espacio en el equipo se recomienda eliminar los archivos fuente y el archivo tar, debido a que ya no son necesarios.

```
[root@utn /]# cd /var/tmp/
[root@utn tmp]# rm -rf sxid-version/
[root@utn tmp]# rm -f sxid_version_tar.gz
```

3.10.2 Configurando sXid

Después de haber instalado sXid, el siguiente paso configurar los parámetros necesarios en los archivos de configuración, estos archivos son:

- /etc/sxid.conf
- /etc/cron.daily/sxid

3.10.2.1 El archivo de configuración sXid: /etc/sxid.conf

El archivo de configuración de sXid permite la modificación de las opciones de operación del programa, los cambios deben hacerse de acuerdo a las necesidades de la institución y son muy pocos los que en realidad deben hacerse, pues en si este archivo ya viene predeterminado para una operación adecuada y óptima.

Paso 1

A continuación se muestra el archivo el sxid.conf, en el que básicamente se le ha modificado:

- La dirección de correo electrónico a la cual se desea que se envíen los reportes.

- El parámetro ENFORCE para remover los archivos con bits 's' en caso de que se encuentren.

- El parámetro MAIL_PROG para indicar donde se encuentra instalado el programa de correo electrónico.

```
# Configuration file for sXid
# Note that all directories must be absolute with no trailing /'s
# Where to begin our file search
SEARCH = "/"
# Which subdirectories to exclude from searching
EXCLUDE = "/proc /mnt /cdrom /floppy"
# Who to send reports to
EMAIL = "root@utn.edu.ec"
# Always send reports, even when there are no changes?
ALWAYS_NOTIFY = "no"
# Where to keep interim logs. This will rotate 'x' number of
# times based on KEEP_LOGS below
LOG_FILE = "/var/log/sxid.log"
```

```
# How many logs to keep.sXid 1
KEEP_LOGS = "5"
# Rotate the logs even when there are no changes?
ALWAYS_ROTATE = "no"
# Directories where +s is forbidden (these are searched
# even if not explicitly in SEARCH), EXCLUDE rules apply
FORBIDDEN = "/home /tmp"
# Remove (-s) files found in forbidden directories?
ENFORCE = "yes"
# This implies ALWAYS_NOTIFY. It will send a full list of
# entries along with the changes
LISTALL = "no"
# Ignore entries for directories in these paths
# (this means that only files will be recorded, you
# can effectively ignore all directory entries by
# setting this to "/"). The default is /home since
# some systems have /home g+s.
IGNORE_DIRS = "/home"
# File that contains a list of (each on it's own line)
# of other files that sxid should monitor. This is useful
# for files that aren't +s, but relate to system
# integrity (tcpd, inetd, apache...).
# EXTRA_LIST = "/etc/sxid.list"
# Mail program. This changes the default compiled in
# mailer for reports. You only need this if you have changed
# it's location and don't want to recompile sxid.
MAIL_PROG = "/bin/mail"
```

Paso 2

Por seguridad se deben cambiar los privilegios del archivo sxid.conf

```
[root@utn /]# chmod 400 /etc/sxid.conf
```

3.10.2.2 El archivo Cron de sXid: /etc/cron.daily/sxid

El archivo sXid es una pequeño script que es ejecutado diariamente por el programa crond del servidor y sigue las pistas o cualquier cambio de los archivos y carpetas s[ug]id y si detecta alguna variación enviará un e-mail al administrador del sistema.

Paso 1

Crear el archivo sxid y añadir lo siguiente:

```
#!/bin/sh
SXID_OPTS=
if [ -x /usr/bin/sxid ]; then
/usr/bin/sxid ${SXID_OPTS}
fi
```

Paso 2

Ahora se debe hacer ejecutable el archivo y cambiarle los persimos a modo 0700

```
[root@utn /]# chmod 700 /etc/cron.daily/sxid
```

3.11 LINUX LOGCHECK

Auditar y almacenar los eventos que ocurren en el sistema es algo muy importante, debido a que esto puede ayudar al administrador a prevenir grandes problemas a los que se ve expuesto el servidor debido la existencia de una conexión a Internet y de que este es accesible a todo el mundo, lo cual lo hace vulnerable de sufrir algún ataque en cualquier momento.

Una de las tareas más importantes en el mundo de la seguridad es chequear regularmente los archivos de log. A menudo las actividades diarias de un administrador no le permiten tener el tiempo libre necesario para dedicarse a esta tarea y esto puede traer problemas en el futuro.

Aquí es donde Logcheck puede ayudar al administrador. Logcheck es un software que está diseñado para ejecutarse

automáticamente y chequear los logs del sistema para verificar violaciones de seguridad o actividades inusuales. Logcheck utiliza un programa llamado logtail que recuerda la última posición de lectura desde un archivo de log y usa esta posición para informar únicamente de los sucesos actuales.

Obviamente el reporte será a través de email para de esta forma facilitarle al administrador la tarea, para que este no tenga que esté revisando todos los días los archivos, pués él únicamente revisará su e-mail y encontrará las novedades del día anterior y de acuerdo a los sucesos podrá tomar decisiones que mejoren la funcionalidad del servidor.

3.11.1 Compilando y Optimizando Logcheck

Antes de proceder con la instalación y optimización de Logchech se debe tener la precaución de estar trabajando con el usuario root para no tener problemas de privilegios durante este procedimiento.

Paso 1

Una vez que se ha obtenido el código fuente del programa se debe copiar dentro de /var/tmp y ubicarse dentro de este directorio para proceder con la instalación.

```
[root@utn /]# cp logcheck_version.tar.gz /var/tmp
[root@utn /]# cd /var/tmp
[root@utn tmp]# tar xzpf logcheck_version.tar.gz
```

Paso 2

Ubicarse dentro del nuevo directorio creado por Logcheck para configurar y optimizar la herramienta.

- Para ubicarse dentro de la nueva carpeta ejecutar el comado:

```
[root@utn tmp]# cd logcheck-1.1.1/
```

- Para optimizar dentro del archivo Makefile establecer la variable CFLAGS en:

```
CFLAGS=”-O3 -march=i686 -mcpu=i686 -funroll-loops -fomit-frame-pointer”
```

Paso 3

Ahora se procede a instalar Logcheck en el servidor, para lo cual se deben ejecutar los siguientes comandos:

```
[root@utn logcheck-1.1.1]# mkdir +m700 /etc/logcheck
[root@utn logcheck-1.1.1]# make linux
```

Estos comandos configuran el software para asegurar que el sistema tiene todas las librerías necesarias, compila todos los archivos fuente dentro de binarios ejecutables e instala los binarios y cualquier archivo de soporte adicional dentro de la ubicación apropiada.

Paso 4

Una vez instalado, compilado y optimizado el software la tarea ha finalizado, para liberar un poco de espacio en el equipo se recomienda eliminar los archivos fuente y el archivo tar, debido a que ya no son necesarios.

```
[root@utn /]# cd /var/tmp/
[root@utn tmp]# rm -rf logcheck-version/
[root@utn tmp]# rm -f logcheck_version_tar.gz
```

3.11.2 Configurando Logcheck

Una vez instalado Logcheck se procederá a verificar o cambiar los archivos de configuración solamente si es necesario caso contrario no. Estos archivos son:

- /etc/logcheck/logcheck.hacking
- /etc/logcheck/logcheck.ignore
- /etc/logcheck/logcheck.violations
- /etc/logcheck/logcheck.violations.ignore

Desde la instalación por default no es necesario modificar los archivos de configuración de Logcheck, las entradas por default son apropiadas y se recomienda no alterarlas a menos que se absolutamente necesario.

Paso 1

Aunque en realidad, no hay necesidad de cambiar los archivos de configuración de Logcheck, una acción necesaria consiste en crear el archivo logcheck dentro /etc/cron.daily/logcheck para que el demonio crond lo ejecute una vez por día.

```
cat <<EOF > /etc/cron.daily/logcheck
# !/bin/sh
# Daily check Log files for security violations and unusual activity
/usr/sbin/logcheck
EOF
```

Paso 2

Ahora se deben cambiar los permisos del archivo y hacerlo ejecutable.

```
[root@utn /]# chmod 700 /etc/cron.daily/logckeck
```

3.12 LINUX PORTSENTRY

Los firewalls ayudan a proteger la red de intrusos, con estos podemos escoger que puertos se quiere que esten abiertos y cuales no, esta información es privada solamente para la organización, nadie de afuera conoce esta información, pero los atacantes, así como los spammers conocen algunos tipos de ataques que pueden usar programas especiales para examinar todos los puertos del servidor hasta encontrar información útil que le indique cuales puertos son accesibles y cuales no.

Un port scan es un indicador de que un gran problema viene en camino, este es a menudo el precursor de un ataque y es una pieza crítica de información para defender adecuadamente los recursos.

Portsentry es un programa diseñado para detectar y responder en contra de port scans a hosts destino en tiempo real y tiene un número de opciones para detectar port scans. Cuando este, encuentra un problema puede reaccionar en uno de los siguiente caminos:

- Un log indicando el incidente sucedido vía syslog().
- El host destino es automáticamente eliminado.
- El host local es automáticamente re-configurado para rutear todo el tráfico al destino para elimiar al host con el fin de desaparecer el sistema destino.
- El host local es automáticamente re-configurado para elimiar todos los paquetes desde el destino vía un filtrado de paquetes local.

El propósito de este es dar al administrador una advertencia de que su host está siendo probado por intrusos.

3.12.1 Compilando y Optimizando PortSentry

Antes de proceder con la instalación y optimización de Portsentry se debe tener la precaución de estar trabajando con el usuario root para no tener problemas de privilegios durante este procedimiento.

Paso 1

Una vez que ha se obtenido el código fuente del programa se debe copiar dentro de /var/tmp y ubicarse dentro de este directorio para proceder con la instalación.

```
[root@utn /]# cp portsentry_version.tar.gz /var/tmp
[root@utn /]# cd /var/tmp
[root@utn tmp]# tar xzpf portsentry_version.tar.gz
```

Paso 2

Ubicarse dentro del nuevo directorio creado por Portsentry para configurar y optimizar la herramienta.

- Para ubicarse dentro de la nueva carpeta ejecutar el comado:

```
[root@utn tmp]# cd portsentry-1.0/
```

- Para optimizar, dentro del archivo Makefile establecer la variable CFLAGS en:

```
CFLAGS=-O3 -march=i686 -mcpu=i686 -funroll-loops -fomit-frame-pointer -Wall
```

Paso 3

Ahora se procede a instalar Portsentry dentro del servidor, para lo cual se deben ejecutar los siguientes comandos:

```
[root@utn portsentry-1.0]# make linux
[root@utn portsentry-1.0]# install -m700 -s portsentry /usr/sbin/
[root@utn portsentry-1.0]# mkdir -p -m700 /etc/portsentry
[root@utn portsentry-1.0]# install -m600 portsentry.conf /etc/portsentry/
[root@utn portsentry-1.0]# install -m600 portsentry.ignore /etc/portsentry/
[root@utn portsentry-1.0]# touch /etc/portsentry/portsentry.modes
[root@utn portsentry-1.0]# chmod 600 /etc/portsentry/portsentry.modes
[root@utn portsentry-1.0]# mkdir -p -m700 /var/log/portsentry
[root@utn portsentry-1.0]# touch /var/log/portsentry/portsentry.blocked
[root@utn portsentry-1.0]# touch /var/log/portsentry/portsentry.history
```

Con los comandos anteriores se configura el software para el sistema operativo Linux, se compilan todos los archivos fuente dentro de binarios ejecutables e instalan los archivo y binarios relacionados a Portsentry en el directorio que se ha establecido por default.

Paso 4

Una vez instalado, compilado y optimizado el software la tarea ha finalizado, para liberar un poco de espacio en el equipo se recomienda eliminar los archivos fuente y el archivo tar, debido a que ya no son necesarios.

```
[root@utn /]# cd /var/tmp/
[root@utn tmp]# rm -rf portsentry-version/
[root@utn tmp]# rm -f portsentry_version_tar.gz
```

3.12.2 Configurando Portsentry

Luego de haber instalado Porsentry, es necesario verificar o cambiar, si es necesario, las opciones de los archivos de configuración, estos archivos son:

- /etc/portsentry/portsentry.conf
- /etc/portsentry/portsentry.ignore
- /etc/portsentry/portsentry.modes
- /etc/rc.d/init.d/portsentry
- /etc/logrotate.d/portsentry

3.12.2.1 El Archivo de Configuración: /etc/portsentry/portsentry.conf

El archivo porsentry.conf es el archivo principal de configuración de Portsentry, el cual permite configurar las opciones que modifican la operación del programa. Los cambios que se realicen a este archivo se harán de acuerdo a los requerimientos y el sistema operativo.

Desde este archivo de configuración se pueden especificar cuales puertos se desea que Portsentry escuche, cuales direcciones IP están denegadas, monitorear, ignorar, deshabilitar automáticamente respuestas, etc.

A continuación se muestra el contenido del archivo portsentry.conf.

```
# PortSentry Configuration
# $Id: portsentry.conf,v 1.13 1999/11/09 02:45:42 crowland Exp crowland $
# IMPORTANT NOTE: You CAN NOT put spaces between your port arguments.
# The default ports will catch a large number of common probes
# All entries must be in quotes.
########################
# Configuración de Puertos #
########################
TCP_PORTS="1,11,15,79,111,119,143,540,635,1080,1524,2000,5742,6667
,12345,
12346,20034,31337,32771,32772,32773,32774,40421,49724,54320"
UDP_PORTS="1,7,9,69,161,162,513,635,640,641,700,32770,32771,32772,
32773,32774,31337,54321"

##############################################
# Opciones de Detección de Escaneo en modo  Avanzado
```

```
################################################
ADVANCED_PORTS_TCP="1023"
ADVANCED_PORTS_UDP="1023"
ADVANCED_EXCLUDE_TCP="113,139"
ADVANCED_EXCLUDE_UDP="520,138,137,67"

########################
# Archivos de Configuración
########################
# Hosts a ignorar
IGNORE_FILE="/etc/portsentry/portsentry.ignore"
# Hosts que han sido denegados
HISTORY_FILE="/var/log/portsentry/portsentry.history"
# Hosts que han sido denegados temporalmente
BLOCKED_FILE="/var/log/portsentry/portsentry.blocked"

###################
# Ignore Options #
###################
# 0 = No bloquear scans UDP/TCP
# 1 = Bloquear scans UDP/TCP
# 2 = Corre comando externo solamente (KILL_RUN_CMD)
BLOCK_UDP="1"
BLOCK_TCP="1"
###################
# Borrar Ruteos#
###################
KILL_ROUTE="/sbin/route add -host $TARGET$ reject"
#####################
# Valor Scan trigger       #
#####################
SCAN_TRIGGER="0"
#####################
# Banner                        #
#####################
PORT_BANNER="** ACCESO NO AUTORIZADO **."
# EOF
```

3.12.2.2 El Archivo /etc/porsentry/portsentry.ignore

El archivo porsentry.ignore es donde se añaden hosts que se desea sean ignorados si se conectan a un puerto que está siendo examinado por Portsentry. Este debe contener el localhost

(127.0.0.1) y la dirección IP de la interfaz local (lo), no se recomienda poner la dirección IP de ningún equipo de la red local.

```
# Put hosts in here you never want blocked. This includes the IP
addresses of all local interfaces on the protected host (i.e virtual
host, mult-home) Keep 127.0.0.1 and 0.0.0.0 to keep people from playing
games.

127.0.0.1
0.0.0.0
```

3.12.2.3 El Archivo de Modos de Portsentry: /etc/ portsentry/ portsentry.modes

El programa Porsentry puede ser configurado en seis modos de operación diferentes, pero solamente uno de ellos puede ser inicializado. Para ser más seguros se puede iniciar una modo TCP y un modo UDP.

Los modos en los cuales trabaja Portsentry son:

- portsentry -tcp (Basic port-bound TCP mode)
- portsentry -udp (Basic port-bound UDP mode)
- portsentry -stcp (Stealth TCP scan detection mode)
- portsentry -sudp (Stealth UDP scan detection mode)
- portsentry -atcp (Advanced "Stealth" TCP scan detection mode)
- portsentry -atcp (Advanced "Stealth" UDP scan detection mode)

Para optimizar el uso del software es preferible usar Portsentry en **Advanced Stealth TCP scan detection mode y Stealth UDP scan detection mode.**

Con Advanced Stealth TCP scan detection mode "-atcp", Portsentry primero cheque para determinar cuales puertos están

corriendo en el servidor, entonces remueve estos puertos para monitorearlos y empieza a ver que sucede con los mismos. Este es muy poderoso y reacciona extremadamente rápido ante un escaneo de puertos, además consume muy poco tiempo de CPU. Este modo es el más sensitivo y efectivo de todas las opciones de protección. Con Stealth UDP scan detection mode "-sudp", los puertos UDP serán escuchados y monitoreados correctamente.

En el archivo /etc/portsentry/portsentry.modes se establecen los modos de operación bajo los cuales Portsentry ejecutará su trabajo. Dentro del archivo pueden haber los seis modos pero solamente dos de ellos pueden estar habilitados.

```
# Place whitespace dilineated modes below.
# Blank lines and pound deliniated comments are ignored.
# tcp
# udp
# stcp
atcp
sudp
# audp
```

3.12.2.4 El Archivo de Inicialización de Portsentry: /etc/ rc.d/ init.d/portsentry

El script /etc/rc.d/init.d/portsentry es el responsable de inicializar y detener automáticamente el demonio Portsentry en el servidor, el contenido de este archivo es:

```
#! /bin/sh
# portsentry Start the portsentry Port Scan Detector
# chkconfig: 345 98 05
# description: PortSentry Port Scan Detector is part of the Abacus Project \
# suite of tools. The Abacus Project is an initiative to release \
# low-maintenance, generic, and reliable host based intrusion \
# detection software to the Internet community.
# processname: portsentry
# configfile: /etc/portsentry/portsentry.conf
```

```
# pidfile: /var/run/portsentry.pid
# Source function library.
. /etc/rc.d/init.d/functions
# Get config.
. /etc/sysconfig/network
# Check that networking is up.
if [ ${NETWORKING} = "no" ]
then
exit 0
fi
[ -f /usr/sbin/portsentry ] || exit 0
# See how we were called.
case "$1" in
start)
echo -n "Starting Port Scan Detector: "
if [ -s /etc/portsentry/portsentry.modes ] ; then
modes=`cut -d "#" -f 1 /etc/portsentry/portsentry.modes`
else
modes="tcp udp"
fi
for i in $modes ; do
portsentry -$i
echo -n "$i "
done
echo
touch /var/lock/subsys/portsentry
;;
stop)
echo -n "Stopping Port Scan Detector: "
killproc portsentry
echo
rm -f /var/lock/subsys/portsentry
;;
status)
status portsentry
;;
restart|reload)
$0 stop
$0 start
;;
*)
echo "Usage: portsentry {start|stop|status|restart|reload}"
exit 1
esac
exit 0
```

3.12.2.5 El Archivo de Rotación de Portsentry: /etc/ logrotate.d/ portsentry

El archivo /etc/logrotate.d/portsentry es el responsable de rotar automáticamente los archivos de log relacionados con Portsentry, el contenido de este archivo es:

```
/var/log/portsentry/portsentry.blocked {
postrotate
/usr/bin/killall -HUP portsentry
endscript
}
/var/log/portsentry/portsentry.blocked.atcp {
postrotate
/usr/bin/killall -HUP portsentry
endscript
}
/var/log/portsentry/portsentry.blocked.sudp {
postrotate
/usr/bin/killall -HUP portsentry
endscript
}
/var/log/portsentry/portsentry.history {
postrotate
/usr/bin/killall -HUP portsentry
endscript
}
```

3.13 LINUX TRIPWIRE

Con el creciente avance de la ciencia y la facilidad de los hackers para encontrar nueva formas de dañar sistemas, era necesario encontrar una herramienta que ayude a la detección de modificaciones no autorizadas en los archivos del sistema.

Tripwire es una herramienta que ayuda a los administadores de sistemas y a los usuarios a monitorear un conjunto designado de archivos para notificar cualquier cambio que pueda sucitarse.

Tripwire puede notificar al administrador la existencia de archivos corruptos, así como de medidas de control que deberían tomarse antes de que los daños ocurran irremediablemente.

Tripwire es un chequeador de integridad de archivos y directorios, una utilidad que compara un conjunto designado de archivos y directorios con información almacenada previamente en una base de datos. Cualquier diferencia que se encuentre es señalada y almacenda, incluyendo la inclusión o eliminación de entradas. Con Tripwire el administador del sistema puede lograr un alto grado de certeza de que un conjunto dado de archivos estará libre de modificaciones no autorizadas si Tripwire no reporta cambios.

3.13.1 Compilando y Optimizando Tripwire

Antes de proceder con la instalación y optimización de Tripwire se debe tener la precaución de estar trabajando con el usuario root para no tener problemas de privilegios durante este procedimiento.

Paso 1

Una vez que se ha obtenido el código fuente del programa se debe copiar dentro de /var/tmp y ubicarse dentro de este directorio para proceder con la instalación.

```
[root@utn /]# cp tripwire_version.tar.gz /var/tmp
[root@utn /]# cd /var/tmp
[root@utn tmp]# tar xzpf tripwire_version.tar.gz
```

Paso 2

Ubicarse dentro del nuevo directorio creado por Tripwire para configurar y optimizar la herramienta.

- Para ubicarse dentro de la nueva carpeta ejecutar el comado:

```
[root@utn tmp]# cd tw_ASR_1.3.1_src/
```

- Para optimizar dentro del archivo Makefile establecer la variable CFLAGS en:

```
CFLAGS=-O3 -march=i686 -mcpu=i686 -funroll-loops -fomit-frame-pointer
```

Paso 3

Ahora se procede a instalar Tripwire en el servidor, para lo cual deben ejecutarse los siguientes comandos:

```
[root@utn tw_ASR_1.3.1_src]# make
[root@utn tw_ASR_1.3.1_src]# make install
[root@utn tw_ASR_1.3.1_src]# chmod 700 /var/spool/tripwire/
[root@utn tw_ASR_1.3.1_src]# chmod 500 /usr/sbin/tripwire/
[root@utn tw_ASR_1.3.1_src]# chmod 500 /usr/sbin/siggen
[root@utn tw_ASR_1.3.1_src]# mv /usr/sbin/tw.config /etc/
[root@utn tw_ASR_1.3.1_src]# strip /usr/sbin/tripwire
[root@utn tw_ASR_1.3.1_src]# strip /usr/sbin/siggen
```

Con los comandos anteriores se configura el software para el sistema operativo Linux, se compilan todos los archivos fuente dentro de binarios ejecutables e instalan los archivos y binarios relacionados a Tripwire en el directorio que se ha establecido por default.

Paso 4

Una vez instalado, compilado y optimizado el software la tarea ha finalizado, para liberar un poco de espacio en el equipo se recomienda eliminar los archivos fuente y el archivo tar, debido a que ya no son necesarios.

```
[root@utn /]# cd /var/tmp/
[root@utn tmp]# rm -rf tw_ASR_version/
[root@utn tmp]# rm -f tripwire_version_tar.gz
```

3.13.2 Configurando Tripwire

Una vez instalado Tripwire, el siguiente paso es verificar los archivos de configuración para determinar si es o no necesario realizar alguna modificación, estos archivos son:

- /etc/tw.config
- /etc/cron.daily/Tripwire

3.13.2.1 El Archivo de Configuración de Tripwire: /etc/tw.config

El archivo tw.config es el archivo de configuración de Tripwire donde se decide cuales archivos y directorios del sistema se desea sean monitoreados.

Paso 1

El formato del archivo /etc/tw.config es el siguiente:

```
# First, root's "home"
/root                                   R
!/root/.bash_history
/                                       R
# OS itself and critical boot resources
/boot                                   R
# Critical directories and configuration files
/bin                                    R
/chroot                                 R
/etc                                    R
/lib                                    R
/sbin                                   R
# Critical devices
/dev/kmem                               R
/dev/mem                                R
/dev/null                               R
/dev/zero                               R
/proc/devices                           R
/proc/net                               R
/proc/tty                               R
/proc/sys                               R
```

```
/proc/cpuinfo                R
/proc/mounts                 R
/proc/dma                    R
/proc/filesystems            R
/proc/ide                    R
/proc/interrupts             R
/proc/ioports                R
/proc/scsi                   R
/proc/kcore                  R
/proc/self                   R
/proc/kmsg                   R
/proc/stat                   R
/proc/fs                     R
/proc/bus                    R
/proc/loadavg                R
/proc/uptime                 R
/proc/locks                  R
/proc/version                R
/proc/meminfo                R
/proc/cmdline                R
/proc/misc                   R

# Other popular filesystems
/usr                         R
/dev                         L-am

# Truncate home
=/home                       R
# var tree
=/var/spool                  L
/var/db                      L
/var/lib                     L
/var/local                   L
!/var/lock
/var/log                     L
/var/preserve                L
/var/spool/cron              L
/var/spool/mqueue            L
/var/spool/mail              L
/var/spool/tripwire          L
# Unusual directories
=/proc                       E
=/tmp
=/mnt/cdrom
```

Paso 2

Por razones de seguridad se deben cambiar los permisos del archivo.

```
[root@utn tmp]# chmod 400 /etc/tw.config
```

3.13.2.2 El Archivo Cron de Tripwire: /etc/cron.daily/tripwire

El archivo tripwire es un pequeño script que es ejecutado automáticamente todos los días por el programa crond del servidor, este examinará los posibles cambios que pudieron haber tenido los archivos o directorios y enviará los resultados vía mail al administrador.

Paso 1

El archivo /etc/cron.daily/tripwire tendrá las siguientes líneas:

```
#!/bin/sh
/usr/sbin/tripwire -loosedir -q | (cat <<EOF
This is an automated report of possible file integrity changes, generated by
the Tripwire integrity checker. To tell Tripwire that a file or entire
directory tree is valid, as root run:
/usr/sbin/tripwire -update [pathname|entry]
If you wish to enter an interactive integrity checking and verification
session, as root run:
/usr/sbin/tripwire -interactive
Changed files/directories include:
EOF
cat) | /bin/mail -s "File integrity report" root
```

Paso 2

Hacer ejecutable archivo y establecer los permisos adecuados

```
[root@utn tmp]# chmod 700 /etc/cron.daily/tripwire
```

3.14 LINUX XINETD

Xinetd es una herramienta que puede controlar ataques denegación de servicio debido a que provee mecanismos de control de acceso para todos los servicios, basándose en direcciones IP de clientes remotos que desean conectarse al servidor, así como también haciendo disponibles los servicios de acuerdo a tiempos de acceso, almacena un registro de accesos y tiene la capacidad de atar los servicios a una interfaz específica.

Pero Xinetd no es eficiente o adecuado para todos los servicios, y especialmente para servicios como FTP y SSH, para estos es mejor ejecutar los servicios independientemente como demonios aparte. Cargando demonios FTP o SSH como demonios independientes se elimina el tiempo de carga y incluso se reduce el swapping de librerías compartidas, además FTP y SSH tienen por sí solos excelentes mecanismos de control.

Unas pocas caracterísitcas de Xinet son:

- Provee mecanismos de control de acceso.
- Previene ataques de negación de servicio.
- Gran capacidad de almacenamiento de accesos al servidor.
- Permite que los servicios estén disponibles en base a tiempo.
- Limita el número de servidores que pueden ser inicializados.
- Soporta Ipv6.
- Interacción con el usuario.

3.14.1 Configurando Xinetd

Los archivos de configuración de Xinetd son los siguientes:

- /etc/xinetd.conf
- /etc/rc.d/init.d/xinetd

Los archivos de configuración con sus respectivas opciones, se encuentran en el Anexo 6.

3.15 SERVIDOR DE NOMBRES DE DOMINIO - DNS

Una vez que se ha instalado el software de seguridad en el servidor Linux, es momento de mejorar y afinar la ejecución de la red en el servidor. El Servidor de Nombres de Dominio es uno de los más importantes servicios de red para la comunicación IP, y por esta razón debe ser instalado en el servidor.

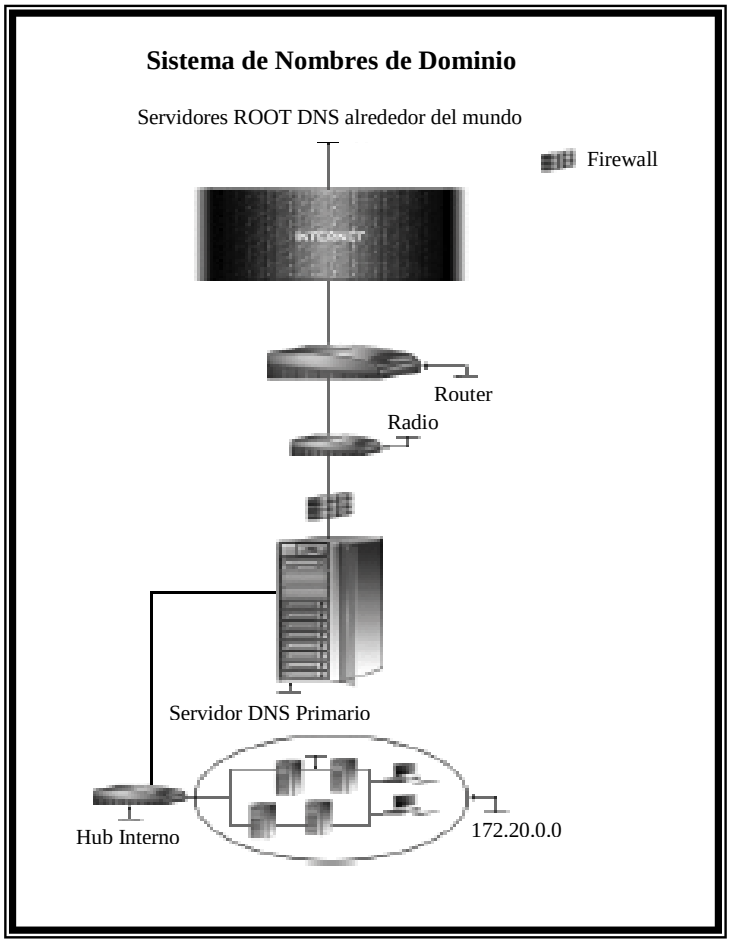

Fig. 3.12 Servidor DNS

Un servidor de nombres es un programa que almacena información acerca de nombres y responde a consultas de programas llamados resolvers, los cuales actúan como procesos cliente. La función básica de un NS es proveer información de los objetos de red sobre los cuales le consultan.

BIND (Berkely Internet Name Domain) es ampliamente usado, es una implementación gratuita de un Servidor de Nombres de Dominio para Linux y NT. Este provee un servidor, una librería cliente y varios programas de utilidad; se considera que un 90% de hosts en el Internet usan esta herramienta como Servidor de Nombres de Dominio.

DNS es un sistema de base de datos distriubida que traduce los nombres de anfitrión a direcciones IP, y direcciones IP a nombres de anfitrión. DNS también es el mecanismo estádar de Internet para almacenar y acceder a varios tipos de datos sobre anfitriones; proporciona información sobre un afitrión determinado al mundo en general.

Los clientes DNS, incluyen cualquier programa que necesita hacer cualquiera de lo siguiente:

- Traducir una nombre de anfitrión a una dirección IP.
- Traducir una dirección IP a un nombre de anfitrión.
- Obtener otra información publicada sobre un afitrión (como su registro MX).

Fundamentalmente cualquier programa que utiliza nombres de anfitrión puede ser un cliente DNS, lo cual inlcuye, en esencia, cualquier programa que tenga que ver algo con redes, incluyendo programas cliente y servidor para Telnet, SMTP, FTP y casi

cualquier otro servdior de red. DNS es por lo tanto un servicio de red fundamental, del cual dependen otros servicios de red.

En el servicio DNS de Linux, en el extremo cliente está el programa resolver, una biblioteca de rutinas invocadas por los procesos de la red. En el extremo servidor está un daemon llamado named.

DNS está diseñado para enviar las solicitudes y respuestas entre los clientes y servidores, a fin de que estos últimos puedan actuar como parte de los clientes u otros servidores.

3.15.1 Cómo funciona DNS

En escencia, cuando un cliente necesita determinada información (por ejemplo la dirección IP del anfitrión ftp.algunlugar.net), se la pide a su servidor DNS local. El servidor DNS local primero examina su propio caché para ver si conoce la respuesta; si no, le pregunta a otros servidores DNS para obtenerla. Cuando el servidor DNS local obtiene la respuesta, la recupera y responde al cliente. Por ejemplo, para encontrar la dirección ftp.algunlugar.net, el servidor DNS local primero le pregunta a uno de los servidores de nombres raíz públicos cuáles máquinas son servidores de nombres para el dominio net, entonces le pregunta a uno de esos servidores de nombres del dominio net cuáles máquinas son servidores de nombres para el dominio algunlugar.net y luego le pregunta a uno de esos servidores de nombres la dirección IP de ftp.algunlugar.net.

Estas preguntas y respuestas son transparentes al cliente: en cuanto a él se refiere, se ha comunicado sólo con el servidor local, no sabe, tampoco le importa, que el servidor local se haya

comunicado con varios otros servidores para responder a la pregunta original.

3.15.2 Configurando ISC BIND & DNS

La configuración del Servidor de Nombres de Dominio (DNS) dependará del tipo de servidor que se desee configurar, en este caso se trata de un Servidor de Nombres de Dominio Primario cuyos archivos de configuración son:

- /etc/named.conf
- /var/named/db.cache
- /var/named/utn.hosts
- /var/named/utn.rev
- /var/named/utn.record
- /var/named/AdmCentral.hosts
- /var/named/AdmCentral.rev
- /var/named/AdmCentral.record
- /var/named/local.hosts
- /var/named/local.rev
- /var/named/local.record
- /etc/sysconfig/named
- /etc/rc./init.d/named

Los archivos de configuración con sus respectivas opciones se encuentran en el Anexo 7.

3.15.3 Ejecutando ISC BIND & DNS en una cárcel chroot

Una cárcel chroot se refiere a un ambiente simulado, bajo el cual se ejecutan ciertos servicios, pero que en caso de que un intruso consiga acceder al equipo y controlar dicho servicio este

únicamente tendrá acceso a la estructura de la cárcel más no al resto de la estructura del sistema.
El ambiente simulado presenta datos falsos, pero el sistema está configurado de tal modo que se registran las actividades del intruso.

Al ejecutar DNS en una cárcel chroot se está previniendo de que este sea usado como un punto de ruptura dentro de la red. En realidad existen muchos sitios que han sufrido ataques remotos luego de que han obtenido acceso root a los hosts corriendo ISC BIND & DNS.

Para minimizar el riesgo ISC BIND & DNS puede ejecutarse como un usuario no root, lo cual limita cualquier daño que puede ser hecho con un usuario normal dentro de su propio shell. Para aumentar la seguridad además se ejecuta DNS en una cárcel chroot.

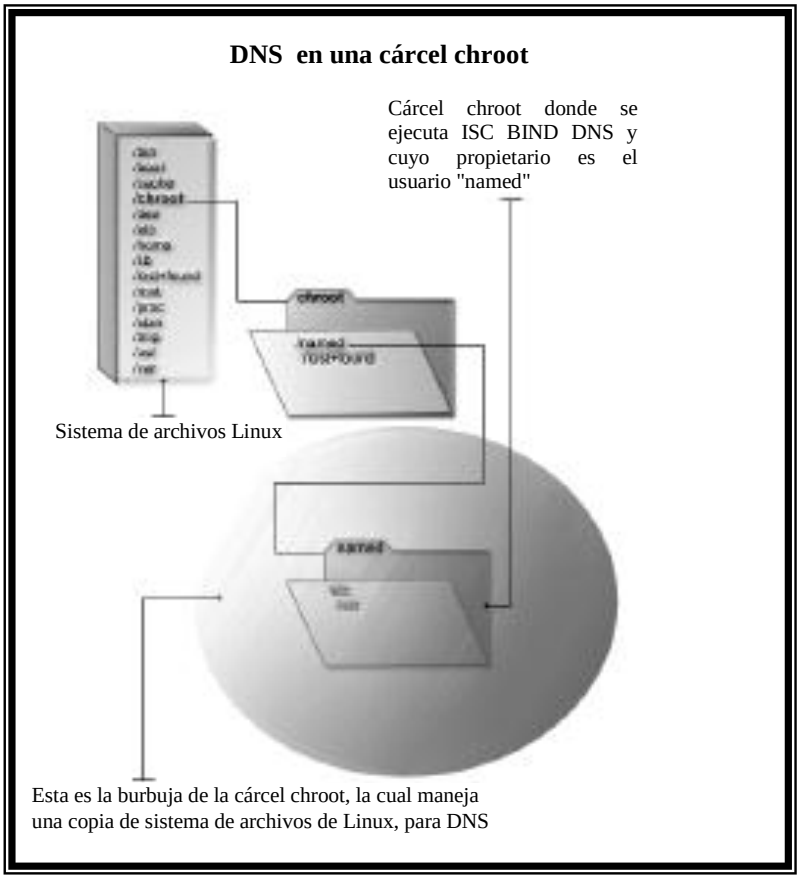

Fig. 3.13 DNS en una cárcel chroot

El principal beneficio de una cárcel chroot es limitar la porción de sistema de archivos que el demonio DNS puede mirar en el directorio de la cárcel. Adicionalmente, desde la cárcel solamente se necesita soporte para DNS, los programas relacionados a ISC BIND & DNS disponibles en la cárcel son extremadamente limitados. Algo muy importante es que no se necesitan programas setuid-root, los cuales pueden ser utilizados para ganar acceso como root y romper la cárcel, la figura 3.12 muestra como es el ambiente de una cárcel chroot.

Los archivos para configurar named dentro de una cárcel chroot, se encuentran en el Anexo 8.

3.16 SERVIDOR DE CORREO ELECTRONICO - SMTP

En Internet el intercambio de correo electrónico entre servidores se maneja con SMTP. Un servidor SMTP en el anfitrión acepta correo y examina la dirección destino para decidir si entrega el correo localmente o lo transmite a otra máquina; si decide entregarlo localmente, vuelve a codificar los encabezados del correo y la dirección de entrega a la forma apropiada para el programa local de entrega, y luego le entrega el correo a ese programa. Si decide transmitir el correo a otra máquina, modifica los encabezados, contacta a esa máquina y envía el correo.

En Linux el software que permite realizar este tipo de trabajo es Sendmail, el cual es un uno de los programas más ampliamente usados en el Internet para transferencia de correo (MTAs - Mail Transfer Agents).

Un Servidor Central de Correo es el encargado de tareas como enviar, recibir y transmitir todo el correo que proceda de la red

local o que tenga como destino la misma, además se encarga de enviar cualquier correo de la red interna que tegan como destino una red externa. En conclusión todos los correos que lleguen desde el Internet se almacenan en el Servidor Central y cualquier correo que desee enviarse pasa a través del Servidor Central de Correo, con esta arquitectura se limita las tareas de administración en el servidor y se mejora la seguridad.

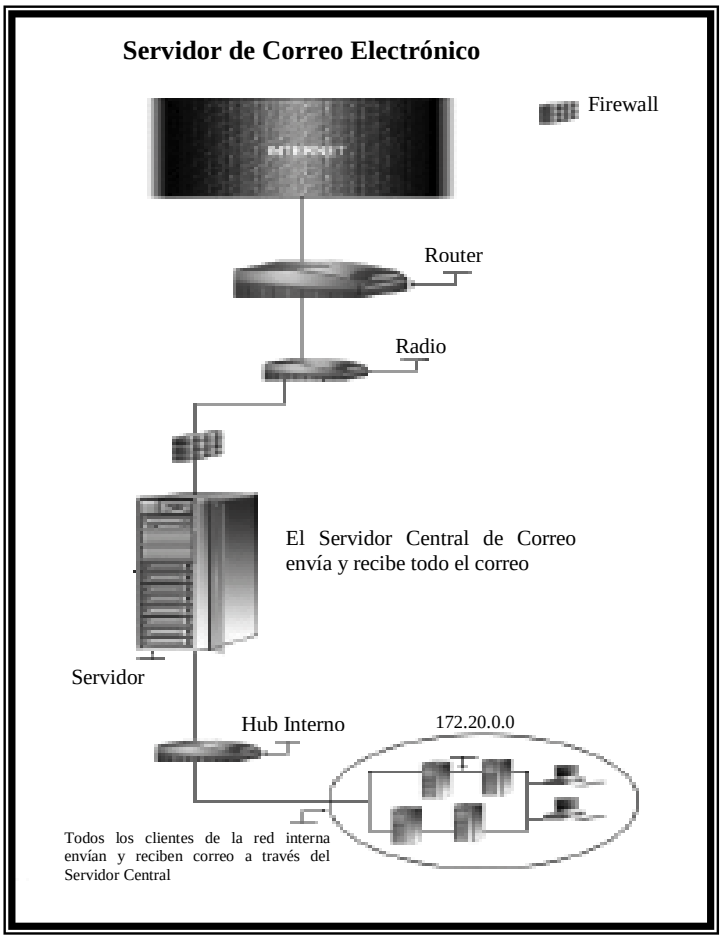

Fig. 3.14 Servidor de Correo Electrónico Sendmail

Luego de instalar un Servidor Central de Correo es necesario instalar un Protocolo de Acceso a Mensajes de Internet, debido a que sendmail es un programa que solamente sirve para enviar y recibir mensajes pero que no los puede leer. Por lo tanto un Servidor Central de Correo necesita un programa que permita a los usuarios conectarse al Servidor Central (sendmail) para obtener y leer sus correos. Un programa de este tipo puede ser

IMAP o POP los cuales son requeridos y deben ser instalados junto con Sendmail.

3.16.1 Configurando Sendmail

Para configurar un Servidor Central de Correo son necesarios los siguientes archivos:

Archivo	Detalle
➢ /etc/mail/access	Archivo de config. de accesos
➢ /etc/mail/access.db	Tabla Hash DB de aliases
➢ /etc/mail/relay-domains	Establece los dominios a enviarse
➢ /etc/mail/aliases	Archivo de alias de usuarios
➢ /etc/mail/aliases.db	Tabla Hash DB de aliases
➢ /etc/mail/virtusertable	Archivo de config. virtusertable
➢ /etc/mail/virtusertable.db	Tabla Hash DB de virtusertable
➢ /etc/mail/domaintable	Archivo de config. domaintable
➢ /etc/mail/domaintable.db	Tabla Hash DB de doamintable
➢ /etc/mail/mailertable	Archivo de config. mailertable
➢ /etc/mail/mailertable.db	Tabla Hash DB de mailertable
➢ /etc/mail/local-host-names	Nombres del Servidor de Correo
➢ /etc/sysconfig/sendmail	Configuraciones del sendmail
➢ /etc/rc.d/init.d/sendmail	Inicia sendmail

Los archivos de configuración de Sendmail con sus respectivas opciones se encuentran en el Anexo 9.

3.16.2 Asegurando Sendmail

A continuación se presentan ciertas medidas que se deben tomar, para mejorar la seguridad de sendmail. Lo interesante de esto es que se trabajará solamente sobre los componentes ya instalados y no se necesitará añadir ningún software especial.

3.16.2.1 Restringir el shell "smrsh"

Los usuarios pueden hacer uso de sendmail para que ejecute comandos en su nombre. El programa vacation es un ejemplo de esto, genera automáticamente respuesta de correo para el coreo entrante. El resultado es que los usuarios pueden ejecutar cualquier programa en el sistema, pero con privilegios del nivel de sistema en vez de con los suyos propios.

Smrsh es un shell restringido para sendmail, si se reemplaza el shell /bin/sh con el shell /etc/smrsh en el archivo de configuración de sendmail. /etc/sendmail.cf se permite que los usuarios solamente utilicen los programas que puede ejecutar sendmail. Smrsh ejecuta sólo los programamas que ha instalado el usuario o a los que se han creado vínculos en el directorio /etc/smrsh/.

Para mayor exactitud, si alguien obtiene sendmail para correr un programa sin ir a través de un aliases o un archivo de reenvio, smrsh limita la cantidad de programas que este intruso puede ejecutar. Cuando es usado en conjunción con Sendamil, smrsh efectivamente limita los alcances de sendmail para ejecutar programas a solamente los especificados en el directorio de smrsh.

Paso 1

Primeramente se debe determinar la lista de comandos que smrsh debería permitir correr a Sendmail, por default estos se le limitan a:

```
"/bin/mail"
"/usr/bin/procmail"
```

Paso 2

El siguiente paso es poblar el directorio /etc/smrsh con los programas para ejecutar Sendmail. Para prevenir duplicación de programas, y hacer que todo trabaje correctamente, lo mejor es establecer enlaces hacia los programas desde /etc/smrsh antes que copiar los programas:

```
[root@utn /]# cd /etc/smrsh
[root@utn smrsh]# ln -s /bin/mail mail
[root@utn /]# cd /etc/smrsh
[root@utn smrsh]# ln -s /usr/bin/procmail procmail
```

Paso 3

Ahora se debe configurar Sendmail para que use el shell restringido, para lo cual se debe modificar la definición de Mprog en el archivo sendmail.cf, reemplazando /bin/sh con /usr/sbin/smrsh.

```
Mprog, P=/usr/sbin/smrsh, F=lsDFMoqeu9, S=10/30, R=20/40, D=$z:/, T=X-
Unix,A=sh -c $u
```

Paso 4

Ahora se debe reiniciar el servidor para que los cambios surtan el efecto deseado.

```
[root@utn /]# /etc/rc.d/init.d/sendmail      restart
Shutting down sendmail:                      [OK]
Starting sendmail:                           [OK]
```

3.16.2.2 El mensaje saludo de SMTP

Cuando Sendmail acepta una conexión SMTP entrante envía una mensaje de saludo al otro host, este mensaje identifica a la

máquina local y es lo primero que envía para decir que ya está listo.

- Para modificar el mensaje de saludo, editar el archivo /etc/sendmail.cf y cambiar la siguiente línea:

```
O SmtpGreetingMessage=$j Sendmail $v/$Z; $b
```

Para leer:

```
O SmtpGreetingMessage=$j
```

3.16.2.3 Cambiar los permisos para los archivos del diretorio /etc/mail

Por seguridad es conveniente cambiar los permisos de los archivos del directorio /etc/mail para que sean solamente de lectura y escritura del super usuario root.

```
[root@utn /]# chmod 600 /etc/mail/*
```

3.16.2.4 Hacer inmutables los archivos del directorio /etc/mail

Para mejorar la seguridad se debe hacer inmutables a todos los archivos que se encuentran bajo el directorio /etc/mail.

```
[root@utn /]# chattr +i /etc/mail/sendmail.cf
[root@utn /]# chattr +i /etc/mail/local-host-names
[root@utn /]# chattr +i /etc/mail/relay-domains
[root@utn /]# chattr +i /etc/mail/aliases
[root@utn /]# chattr +i /etc/mail/access
[root@utn /]# chattr +i /etc/mail/virtusertable
[root@utn /]# chattr +i /etc/mail/domaintable
[root@utn /]# chattr +i /etc/mail/mailertable
```

3.17 PROTOCOLO DE ACCESO A MENSAJES DE INTERNET - UW IMAP

SMTP se utiliza para intercambiar correo entre servidores. Comúnmente los usuarios acceden a su correo como un archivo en la máquina donde fue entregado; sin embargo, a veces hay razones para utilizar un protocolo independiente para distribuir el correo de un servidor a un usuario individual.

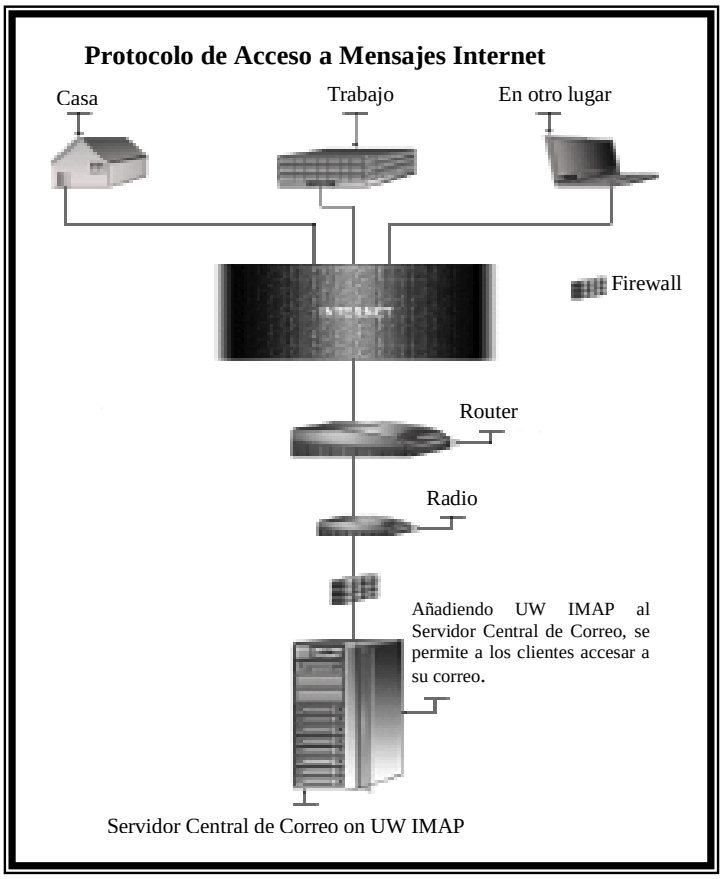

Fig. 3.15 Protocolo de Acceso a Mensajes de Internet

IMAP es un protocolo cliente/servidor que sirve para manejar buzones electrónicos de usuario. Con IMAP, el buzón de un usuario (el verdadero archivo donde el correo electrónico del usuario se guarda para que lo vea después) está en un servidor, no en la máquina personal del usuario, cuando el usuario quiere su correo electrónico, accede a su buzón utilizando un programa

cliente que se encuentra instalado en su máquina y emplea el protocolo IMAP.

3.17.1 Configurando UW IMAP

Los archivos necesarios para configurar WU IMAP son:

- /etc/pam.d/imap
- /etc/pam.d/pop

Los archivos de configuración con sus respectivas opciones se encuentran en el Anexo 10.

3.18 SERVIDOR PROXY - SQUID (Plataforma Linux)

Un sistema proxy proporciona aceso a Internet a un solo anfitrión,o a un número muy pequeño de anfitriones, aunque parece que lo proporciona a todos. Los anfitriones que sí tienen acceso actúan como proxies para las máquinas que no lo tienen, haciendo lo que estas últimas quieren que se haga.

Los servidores proxy actúan de la siguiente manera: algún anfitrión con el que pueda establecer comunicación el usuario, que puede a su vez comunicarse con el mundo exterior; el programa cliente del usuario se comunica con el servidor proxy en lugar de hacerlo directamente con el servidor real que está en Internet. El servidor proxy evalúa solicitudes del cliente y decide cuales pasar y cuales no, si una petición es aprobada el servidor proxy habla con el servidor real en nombre del cliente y procede a transmitir las solicitudes del cliente al verdedaro servidor y a transmitir las respuestas de éste de nuevo al cliente.

Un sistema proxy no requiere de un hardware especial, aunque si de un software especial para la mayoría de servicios, por lo cual existen en el mercado algunos programas de este tipo que optimizan el ancho de banda, mejoran la seguridad e incrementan la velocidad de navegación en el web. Pero estos servidores proxy comerciales tiene dos inconvenientes: son comerciales y no soportan ICP. Squid es la mejor elección en cuanto a servidores proxy de cache por que es robusto, gratuito y puede usar características ICP.

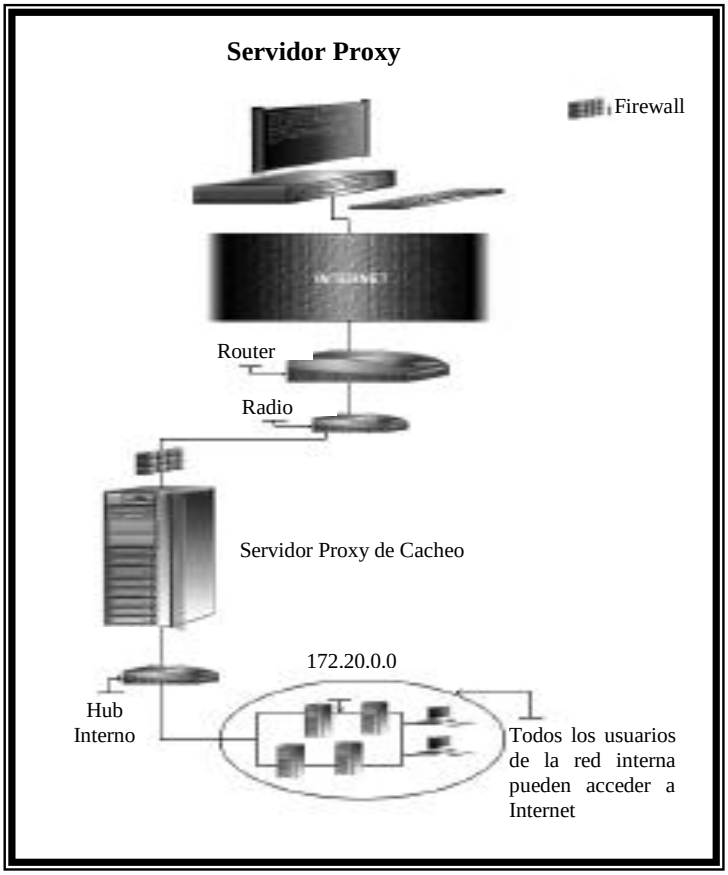

Fig. 3.16 Servidor Proxy Squid

Squid ofrece amplias características en la ejecución de caché para clientes web, además soporta objetos de datos FTP, Gopher, HTTP y HTTPS. Este almacena objetos calientes en RAM, mantiene una robusta base de datos en disco, tiene un complejo mecanismo de control de acceso y soporta protocolo SSL para conexiones proxificadas seguras. Además puede ser enlazado

jerárquicamente a otros servidores proxy para optimizar el caché de páginas.

Squid se configurará como un servidor proxy de caché para permitir a todos los usuarios de la red interna acceder a Internet a través del proxy. Al utilizar el servidor proxy para permitir el acceso de los clientes al exterior se mejora la seguridad y la velocidad de acceso gracias a la utilización del caché que proporciona el servidor proxy.

Con Squid como un servidor proxy de caché se pueden bloquear sitios restringidos, sitios no convenientes para el público y controlar el acceso de las personas de la red interna al Internet.

3.18.1 Configurando Squid

Los archivos necesarios para la configuración y optimización de squid son:

- /etc/squid/squid.conf
- /etc/sysconfig/squid
- /etc/logrotate.d/squid
- /etc/rc.d/init.d/squid

Los archivos de configuración con sus respectivas opciones se encuentran en el Anexo 11.

3.18.2 Asegurando Squid

A continuación se explican unas medidas de seguridad que se deben tomar para asegurar los principales archivos de configuración de Squid.

3.18.2.1 Inmunizando el archivo de configuración de Squid

Con el objetivo de mejorar la seguridad y prevenir cualquier inconveniente que pueda presentarse con el archivo /etc/squid/squid.conf es conveniente inmunizarlo con el siguiente comando:

```
[root@utn /]# chattr +i /etc/squid/squid.conf
```

3.18.2.2 Memoria física

El recurso más importante de Squid es la memoria física, no es necesario tener un procesador de mucha capacidad, en el caso de Squid lo más conveniente es tener la mayor cantidad de memoria posible y a la vez disponer de un disco de SCSI, para de esta manera aprovechar las características de velocidad que el hardware pueda brindar.

3.19 SERVIDOR FTP - Wu-ftpd (Plataforma Linux)

FTP se utiliza para transferir archivos de una máquina a otra. Se puede utilizar para transferir cualquier tipo de archivo,incluyendo binarios, ejecutables, imágenes, texto ASCII, PostScript, archivos de sonido, video y más. Hay dos tipos de acceso FTP: FTP de usuario y FTP anónimo. FTP de usuario requiere de una cuenta en el servidor y permite a los usuarios obtener cualquier archivo como si iniciaran una sesión local. FTP anónimo es para las personas que no tienen una cuenta y se utiliza para proporcionar archivos específicos al mundo en general.

El uso de FTP anónimo es más común en el Internet, los servidores FTP anónimos son el mecanismo estándar para

distribuir programas, información y otros archivos que los sitios desean poner disponibles vía Internet. Si se proporciona un servidor de FTP anónimo, cualquier persona en Internet puede iniciar una conexión FTP con servidor y acceder a cualquier archivo que los dueños del servidor hayan seleccionado para ponerlo disponible en un área no restringida.

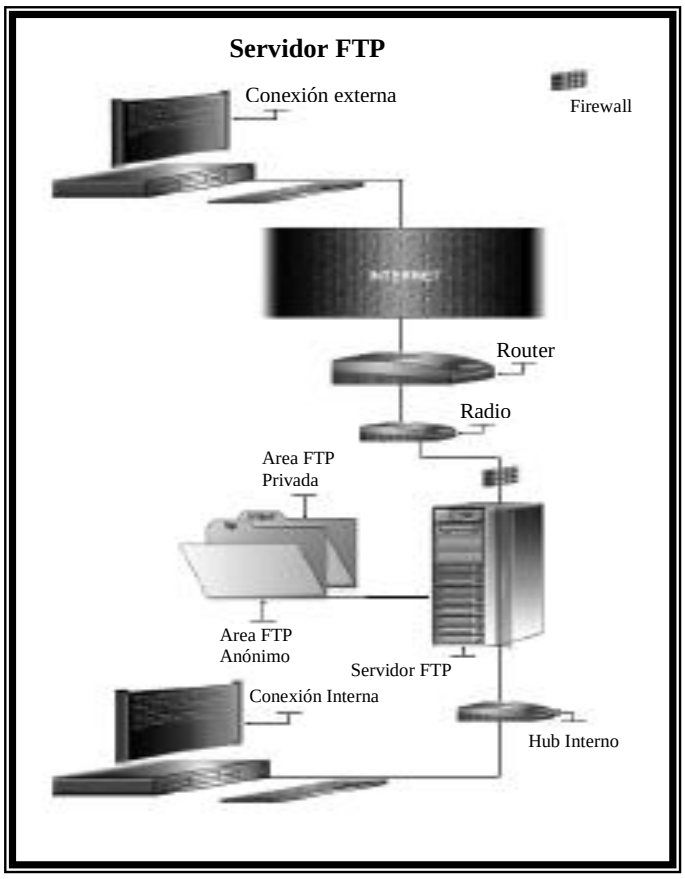

Fig. 3.17 Servidor FTP

3.19.1 Ejecutando Wu-ftpd en una cárcel chroot

Esta parte se enfoca a prevenir que Wu-ftpd sea usado como un punto débil dentro del sistema. El principal beneficio de utilizar la cárcel chroot es limitar la parte del sistema de archivos que el demonio puede mirar dentro del directorio principal de la cárcel. Adicionalmente, dado que la cárcel solamente necesita soportar Wu-ftpd, los programas disponibles en la cárcel son extremadamente limitados.

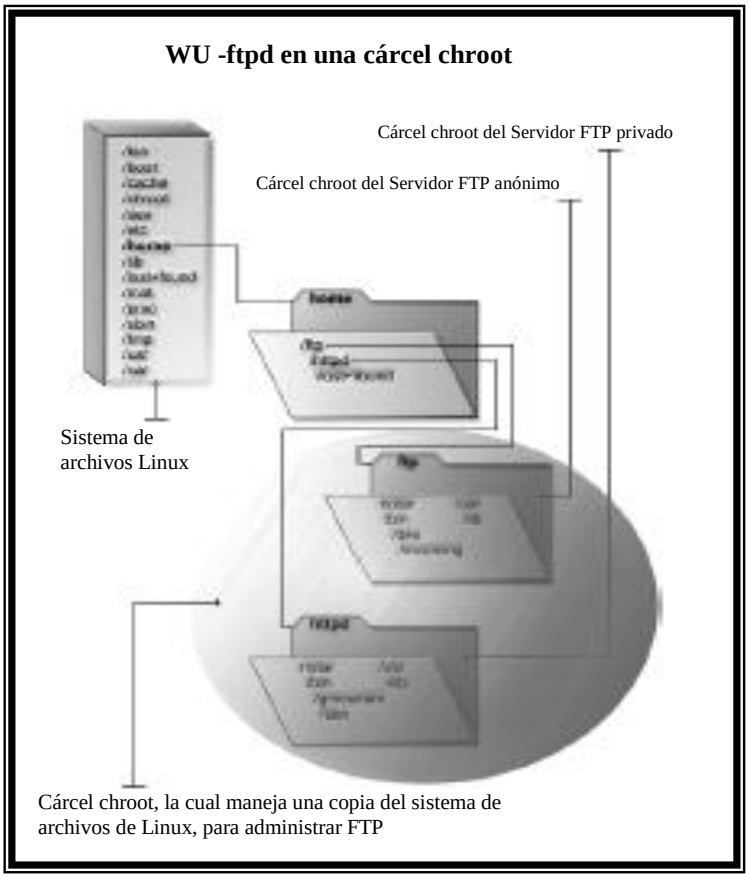

Fig. 3.18 Wu-ftpd en una cárcel chroot

Algo mucho más importante, es que no hay necesidad de tener programas setuid-root, los cuales pueden ser usados para ganar acceso root y romper la cárcel. Ejecutando Wu-ftpd en una cárcel chroot mejora significativamente la seguridad del sistema.

Los pasos necesarios y los archivos de configuración de FTP dentro de una cárcel chroot, se encuentran en el Anexo 12

3.19.3 Asegurando Wu-ftpd

A continuación se detallan ciertas medidas que se deben tomar para mejorar y optimizar la seguridad de Wu-ftpd. Todo lo que se detalla a continuación se basa en el programa instalado y no se requiere de ningún software adicional.

3.19.3.1 El comando upload

Por default el servidor Wu-ftpd da privilegios de upload a todos los usuarios. El parámetro upload permite a clientes remotos cargar y colocar archivos en el Servidor FTP. Para óptima seguridad no se debe permitir que los usuarios tengan posibilidad de cargar dentro de los directorios "/", "/bin", "/dev", "/lib", "/usr" y "/usr/bin" del directorio chroot /home/httpd.

En el archivo /etc/ftpaccess se ha utilizado esta característica con el fin de mejorar la seguridad, esto se establece en las siguientes líneas:

```
upload /home/httpd * no
upload /home/httpd * /dev no
upload /home/httpd * /bin no
upload /home/httpd * /lib no
upload /home/httpd * /usr no
upload /home/httpd * /usr/bin no
# Areas where upload clauses are allowed.
upload /home/httpd /ftputn yes ftputn ftputn 0644 dirs 0755
upload /home/httpd /ftputn/* yes ftputn ftputn 0644 dirs 0755
```

En las dos últimas líneas se permite cargar dentro del directorio y de un subdirectorio de /ftputn con permisos de archivos en 644 y la creación de nuevos directorios con permisos 755 para usuarios guest y el grupo ftputn.

4.19.3.2 El Archivo especial .notar

Si se está protegiendo los directorios para que se no se realicen cargas indebidas, es además conveniente no permitir el uso del comando tar en áreas donde el comando upload no está permitido.

Paso 1

Crear el archivo especial .notar en cada directorio y en el directorio FTP. No se debe usar el comando touch para crear un .notar, se debe usar echo:

```
[root@utn /]# echo “Tarring is denied” > /home/httpd/.notar
[root@utn /]# echo “Tarring is denied” > /home/httpd/dev/.notar
[root@utn /]# echo “Tarring is denied” > /home/httpd/bin/.notar
[root@utn /]# echo “Tarring is denied” > /home/httpd/lib/.notar
[root@utn /]# echo “Tarring is denied” > /home/httpd/usr/.notar
[root@utn /]# echo “Tarring is denied” > /home/httpd/usr/bin/.notar
```

Paso 2

Cambiar los permisos de los archivos .notar por razones de seguridad:

```
[root@utn /]# chmod 0444 /home/httpd/.notar
[root@utn /]# chmod 0444 /home/httpd/dev/.notar
[root@utn /]# chmod 0444 /home/httpd/bin/.notar
[root@utn /]# chmod 0444 /home/httpd/lib/.notar
[root@utn /]# chmod 0444 /home/httpd/usr/.notar
[root@utn /]# chmod 0444 /home/httpd/usr/bin/.notar
```

3.19.3.3 El comando noretrieve

El parámetro noretrieve del servidor Wu-ftp permite negar la transferencia de archivos o directorios seleccionados. Es conveniente prevenir downloads de los subdirectorios /dev, /bin, /lib, /usr y /usr/bin en el directorio /home/http.

En el archivo /etc/ftpaccess se ha utilizado esta característica con el fin de mejorar la seguridad, eto se establece en las siguientes líneas:

```
# We'll prevent downloads with noretrieve.
noretrieve /home/httpd/dev/
noretrieve /home/httpd/bin/
noretrieve /home/httpd/lib/
noretrieve /home/httpd/usr/
noretrieve /home/httpd/usr/bin/
```

3.20 SERVIDOR WEB - APACHE

Apache es el servidor HTTP más ampliamente usado en estos días, este es mejor que cualquier servidor Web gratuito o comercial, y provee una gran cantidad de características y ventajas que lo hacen fácil de usar. La función básica de un servidor Web como Apache, es desplegar y servir páginas HTML almacenadas en el servidor para un cliente que entiende código HTML. En conjunción con otros módulos o programas puede llegar a convertirse en un poderoso software, el cual podrá proveer servicios útiles a un cliente.

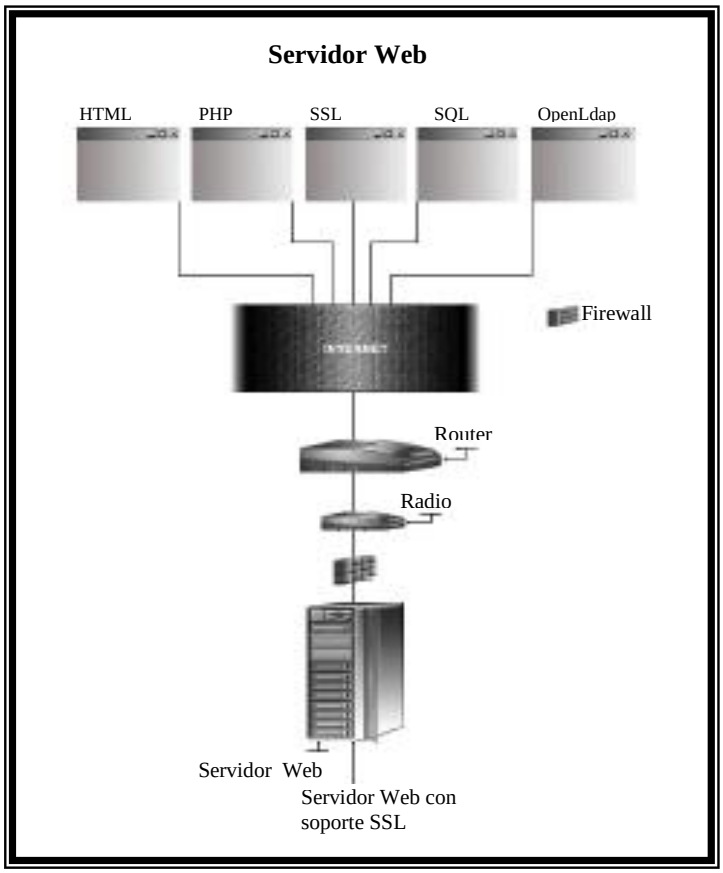

Fig. 3.19 Servidor Web Apache

Existen una gran variedad de módulos que se pueden agregar a Apache con el fin de que brinde un mejor servicio, pero en este caso se van a explicar los más importantes y que son necesarios para el funcionamiento de otros servicios que se desea preste el Servidor Web, estos son: mod_ssl, PHP4, SQL database.

3.20.1 Configurando Apache

Los archivos de configuración para los diferentes servicios son muy específicos dependiendo de las necesidades que se tenga. Algunas veces se puede instalar Apache para que solamente sirva páginas web, otras veces se puede instalar con conectividad a una base de datos y soporte SSL, etc. En este caso se ha instalado Apache para que sirva páginas Web, soporte PHP4, SSL y además se ha añadido autentificación con password para ciertas páginas.

Los archivos de configuración de Apache son:

- /etc/httpd/conf/httpd.conf
- /etc/logrotate.d/apache
- /etc/rc.d/init.d/httpd

Los archivos de configuración con sus respectivas opciones se encuentran en el Anexo 14.

3.20.2 Asegurando Apache

A continuación se describen algunas características que mejoran y optimizan la seguridad de Apache. Lo interesante es que se hace referencia únicamente a características propias del software y no hay necesidad de añadir ningún programa especial.

3.20.2.1 Cambiando los permisos de algunos archivo y directorios del Servidor Web

Cuando se instala Apache, existen algunos archivos y directorios que tiene demasiados privilegios por default. El programa binario httpd puede ser establecido como de solo lectura para el super-usuario root, ejectuable por el propietario, el grupo y otros, por razones de seguridad. Los directorios /etc/httpd/conf y /var/log/httpd no necesitan ser de lectura, escritura o ejecutables para otras personas.

```
[root@utn /]# chmod 511 /usr/sbin/httpd
[root@utn /]# chmod 700 /etc/httpd/conf/
[root@utn /]# chmod 700 /var/log/httpd/
```

3.20.2.2 Inmunizar el archivo de configuración /etc/httpd/conf/ httpd.conf

Al hacer inmutable el archivo de configuración /etc/httpd/conf/httpd.conf se previene de borrados, sobre-escritura o creación de enlaces simbólicos. Para inmunizar el archivo se utiliza el siguiente comando:

```
[root@utn /]# chattr +i /etc/httpd/conf/httpd.conf
```

3.20.2.3 Crear el archivo .dmpasswd para autenticación de usuarios

El archivo .dbmpasswd se utiliza cuando se desea usar autenticación de usuarios para el acceso a las páginas Web y de esta manera lograr la protección de áreas restringidas.

Paso 1

El programa dbmmanage, el cual viene por default con Apache, se emplea para crear y actualizar nombres de usuario y passwords para usuarios HTTP. Este método usa un formato de archivo DBM este es el mecanismo más rápido cuando se tiene que manejar cientos de usuarios en un archivo de password. Primero que todo es importante cambiar los permisos del programa para que tenga permisos de escritura, lectura y ejecución solamente por el super-usuario root

```
[root@utn /]# chmod 750 /usr/bin/dbmmanage
```

Paso 2

Una vez que se ha establecido los permisos apropiados, se debe crear el archivo de formato DBM con un nombre de usuario y su password.

```
[root@deep /]# /usr/bin/dbmmanage /etc/httpd/dbmpasswd adduser prueba
New password:
Re-type new password:
User prueba added with password encrypted to dtkTL83yvMbFQ using crypt
```

Paso 3

Ahora es momento de añadir en el httpd.conf la parte de Sitio Web que estará protegida con nombres de usuario y contraseña, para lo cual se añaden las siguientes líneas en el archivo de configuración:

```
<Directory "/var/www/html/private">
Options None
AllowOverride AuthConfig
AuthName "Area Restringida"
AuthType Basic
AuthDBUserFile /etc/httpd/dbmpasswd
require valid-user
</Directory>
```

Paso 4

Reninicar el servicio para que los cambios surtan efecto.

```
[root@utn /]# /etc/rc.d/init.d/httpd restart
Shutting down http:                [OK]
Starting httpd:                    [OK]
```

Paso 5

Finalmente, para probar el área protegida se debe acceder desde el navegar web y escribir la siguiente dirección http://www.utn.edu.ec/private/ y deberá aparecer una pantalla similar a la siguiente:

Fig. 3.20 Area private de Apache

3.21 SERVIDOR PARA COMPARTIR ARCHIVOS - SAMBA

Actualmente en las organizaciones se manejan diferentes tipos de sistemas operativos y se tiene la necesidad de compartir archivos e impresoras entre todos los equipos de la red, independientemente del sistema operativo que se este usando.

Samba es un gran servicio de red que permite compartir archivos e impresoras y funciona sobre la mayoría de sistemas operativos que existen hoy en día.

Samba es un protocolo a través del cual una gran cantidad de PCs pueden compartir archivos e impresoras y otro tipo de información si fuere necesarios. Los sistemas operativos que soportan nativamente este protocolo son Windows 95/98/2000/NT y añadiendo ciertos paquetes también está disponible para DOS, VMS, ciertos UNIX y más.

Apple Macs y algunos Web Browsers pueden igualmente entenderse con este protocolo. Alternativas para SMB incluye Netware, NFS, Apple Talk, Banyan, Decnet, etc. muchos de estos tienen ventajas, ninguno tiene especificaciones públicas y amplias implementaciones en equipos por default.

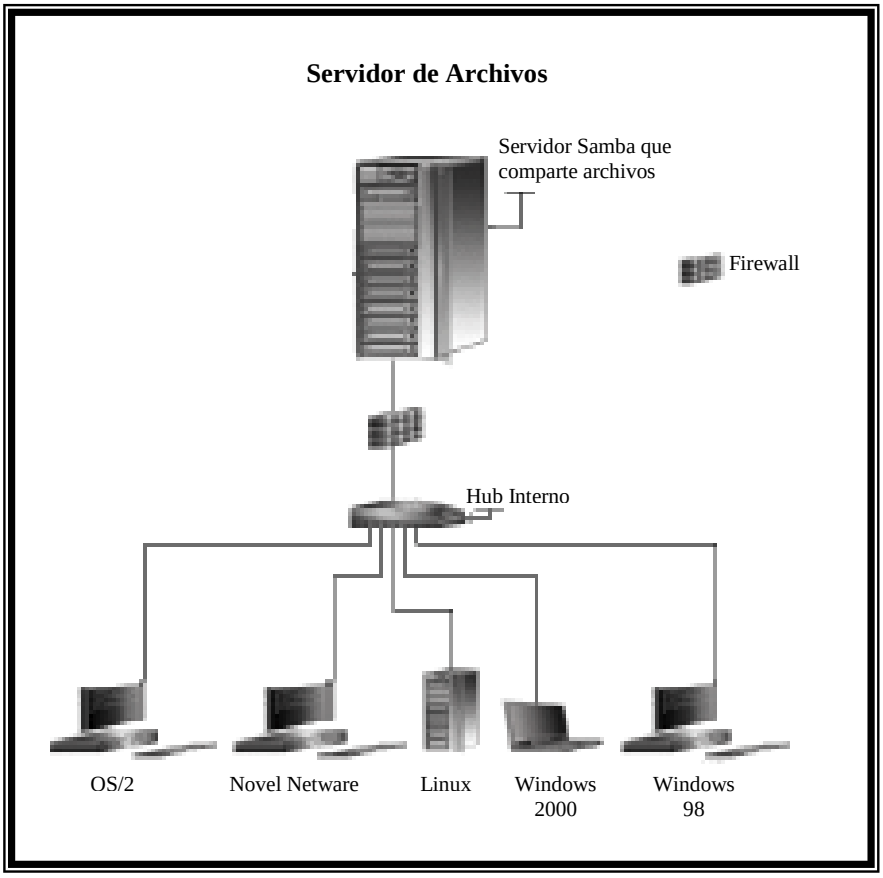

Fig. 3.21 Servidor para comparit archivos Samba

El Software de Samba incluye un servidor SMB, para proveer archivos para Window NT y LAN Manager-style y servicios de impresión para clientes SMB como Windows 200, Warp Server, smbfs y otros, un servidor de nombres NetBIOS, el cual entre

otras cosas da soporte browsing, un ftp como cliente SMB así que puede accesar a los recursos del PC desde Unix, Netware y otros sistemas operativos, finalmente, una extensión tar a un cliente para backup de PCs.

3.21.1 Configurando Samba

Los archivos necesarios para configurar Samba son:

➢ /etc/samba/smb.conf	Archivo de configuración
➢ /etc/samba/lmhosts.conf	Mapea archivos Net BIOS
➢ /etc/sysconfig/samba	Configuración de samba
➢ /etc/pam.d/samba	Soporte PAM para samba
➢ /etc/logrotate.d/samba	Rotación de logs
➢ /etc/rc.d/init.d/smb	Inicia samba

Los archivos de configuración con sus respectivas opciones, se encuentran en el Anexo 15

3.21.2 Asegurando Samba

A continuación se explican unas medidas de seguridad que se deben tomar para asegurar los principales archivos de configuración de Samba.

3.21.2.1 Crear el archivo de password encriptado para conexión de clientes

El archivo /etc/samba/smbpasswd es donde se almacenan los passwords encriptados de los clientes que pueden conectarse al Servidor Samba.

Es importante crear este archivo de password e incluir los nombres de usuarios que tienen privilegios de conexión antes de

que los clientes intenten conectarse con el servidor. Sin este paso no será posible la conexión hacia el Servidor Samba.

Paso 1

Para crear nuevas cuentas de usuarios Samba en el sistema, es necesario primero crear una cuenta válida en Linux antes de habilitarla para el servicio smb.

- Usar el siguiente comando para crear un nuevo usuario en el archivo /etc/passwd.

```
[root@utn /]# useradd -s /bin/false smbadmin 2>/dev/null || :
[root@utn /]# passwd smbadmin
Changing password for user smbadmin
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully
```

Paso 2

Una vez que se ha añadido el cliente a /etc/passwd en el servidor Linux, se puede generar el archivo smbpasswd, para lo cual se ejecuta el siguiente comando.

```
[root@utn /]# cat /etc/passwd | mksmbpasswd.sh > /etc/ samba/
                smbpasswd
```

Paso 3

Finalmente, el último paso es crear el mismo nombre de usuario como una cuenta Samba y añadirla al archivo /etc/samba/smbpasswd.

```
[root@utn /]# smbpasswd -a smbadmin
New SMB password:
Retype new SMB password:
Added user smbadmin.
Password changed for user smbadmin.
```

Paso 4

Cambiar los permisos del archivo smbpasswd para que solamente el super-usuario root pueda leerlo y escribirlo. Esta es una característica de seguridad.

```
[root@utn /]# chmod 600 /etc/samba/smbpasswd
```

3.21.2.2 Inmunizando el archivo de configuración de Samba

Con el objetivo de mejorar la seguridad y prevenir cualquier inconveniente que pueda presentarse con los archivos /etc/samba/smb.conf y /etc/samba/lmhosts es conveniente inmunizarlos con el siguiente comando:

```
[root@utn /]# chattr +i /etc/samba/smb.conf
[root@utn /]# chattr +i /etc/samba/lmhosts
```