Mejorando la seguridad de OpenSSH

OpenSSH es la navaja suiza de los programas de acceso remoto: le proporciona un shell en su máquina distante,y transmite datos de forma segura y encriptada - incluyendo comandos, transferencia de archivos, sesiones X11 y VNC, datos rsync, etc.

OpenSSHes tan avanzado que puede considerarse como una VPN simplificada (red privada virtual).

Slackware contiene OpenSSHpor supuesto, y la configuración por defecto ya es bastante segura.Esta página se ha creado para mostrarle algunos ajustes de configuración simples que se pueden aplicar a la configuración predeterminada para mejorar su seguridad.

Necesitará saber cómo usar un editor de texto para poder seguir este HOWTO. Si eres un principiante completo, prueba `nano`.Una vez que esté más avanzado y más cómodo con Linux, siéntase libre de usar algo más poderoso ...

Los archivos de configuración de SSH y encontrar más información

Los archivos de configuración de OpenSSH residen en el directorio`/etc/ssh/`. El más importante es el `/etc/ssh/sshd_config` que vamos a modificar aquí.

La documentación de OpenSSH es muy buena, así que siéntase libre de ingresar el comando: `apropos openssh` or `man -k openssh` y lea las diferentes páginas man, que son mucho más detalladas que esta página wiki.

Modificando el archivo sshd_config

La primera regla, antes de modificar un archivo de configuración importante es hacer una copia de respaldo. Por ejemplo:

```
# cp -v /etc/ssh/sshd_config /etc/ssh/sshd_config.ORIG.20120826
```

El comando mostrado arriba hace una copia del archivo `sshd_config` y añadir la extención `.ORIG` (por original, por supuesto)e incluir la fecha de modificación. Aunque no es perfecto, este sistema asegurar que siempre pueda volver a la versión previa del importante archivo ingresando el siguiente código.

```
# cp -v /etc/ssh/sshd_config.ORIG.20120826 /etc/ssh/sshd_config
```

Tenga en cuenta que esta es una sugerencia, por supuesto, y se recomienda encarecidamente a los administradores de sistemas que tienen que administrar grandes instalaciones, con cientos de servidores, investigar mejores soluciones, como Puppet u otros gestores de configuración de sistemas distribuidos.

Ahora, edite el archivo `/etc/ssh/sshd_config` con su editor favorito y ajusta las siguientes lineas:

Cambiar el puerto predeterminado de SSH

Por defecto, OpenSSH escucha en el puerto 22. A veces se recomienda cambiar este puerto predeterminado a algo diferente, como 2222 o 4242. Esto no es una mala idea, pero debe recordar que al escanear su máquina con un programa como `nmap` obtendrá este nuevo puerto muy rápidamente.Por lo tanto, si bien puede ralentizar algunos ataques en su máquina, no los evitará por completo.

Si desea cambiar el puerto, busque la opción `Port` en `sshd_config`, que generalmente está en la parte superior del archivo, y cambie el valor.

Por ejemplo:

```
Port 22
```

Puede cambiarse por:

```
Port 4242
```

Método alternativo para cambiar el puerto predeterminado de SSH sin cambiarlo

Esta sugerencia es completamente sacada de otro sitio. redcodenetwork.ro La idea aquí es agregar una llamada a este script en `rc.local` para que se ejecute al inicio.
Para usar el ejemplo de descarga en la parte inferior, cámbielo a su gusto y guárdelo en `/etc/rc.d/` (haga que sea ejecutable)
Agregue este ejemplo a continuación a `/etc/rc.d/rc.local`.

```
if [ -x /etc/rc.d/rc.ssh_hide ]; then
 /etc/rc.d/rc.ssh_hide
fi
```

Lo que está haciendo es hacer que parezca que ha cambiado el puerto que está usando ssh y proporcionar cierta seguridad adicional. Lo que sucede es que los escáneres continuarán viendo el puerto 22 abierto e intentarán ir allí mientras su servidor deja caer esos paquetes, porque el encabezado no está dañado. Los paquetes reales vienen en el puerto 8889 y son redirigidos por iptables al puerto 22 con mangle en el encabezado para que no se caigan.

rc.ssh_hide

```
#!/bin/bash

######First, set SSHD back to the default port 22.
######Next, figure out what port or ports you want to do SSH over.
######Were going to use 99, 88, and 8889 here.
```

```
######Now we take care of the Hypothetical Evil Unprivileged User
######by not accepting anything over those ports in the first place.
######This is only effective for port 8889 but well do all three ports
for the sake of completeness.

/usr/sbin/iptables -t filter -A INPUT -p tcp -m multiport --dports
99,88,8889 -j REJECT --reject-with tcp-reset

######Then, pick a number between 1 and 4294967295 Ill use 0x13F ()
######Were going to tell iptables to reject anything without this mark
coming into port 22.

/usr/sbin/iptables -t filter -A INPUT -p tcp -m tcp --dport 22 -m
connmark ! --mark 0x13F -j REJECT --reject-with tcp-reset

######Now well tell iptables what ports we will accept for ssh.

/usr/sbin/iptables -t filter -A FORWARD -p tcp -m multiport --dports
99,88,8889 -j ACCEPT

######In the mangle table we slap our mark on these packets.

/usr/sbin/iptables -t mangle -A PREROUTING -p tcp -m multiport --dports
99,88,8889 -j CONNMARK --set-mark 0x13F

######Finally in the nat table we tell iptables to send the marked
packets back to port 22

/usr/sbin/iptables -t nat -A PREROUTING -p tcp -m multiport --dport
99,88,8889 -j REDIRECT --to-ports 22

exit 0
```

Tenga en cuenta que no es compatible con ipv6, debido a nat (ip6tables no son compatibles con nat)

Prohibir el acceso root a su máquina

Este es probablemente el cambio más importante que puede hacer para mejorar la seguridad de su máquina: prohibir que el usuario `root` acceda a su máquina. Para hacer esto, busque la siguiente línea en `sshd_config`:

```
PermitRootLogin yes
```

Y cambiarlo a:

```
PermitRootLogin no
```

Si aplica los cambios anteriores, asegúrese de tener al menos un usuario en su máquina que pueda (cambiar de usuario) `su` a `root` o use `sudo` para permisos de administración del sistema más

específicos . La mejor manera de administrar un servidor a través de SSH es tener un usuario en el grupo `wheel`, que puede usar `sudo` y `su` para convertirse en `root` cuando sea necesario.

Ya que la mayoría de las personas que intentan ingresar a su máquina sin ser invitadas usan scripts de fuerza bruta que apuntan al inicio de sesión 'raíz', ya puede descansar mucho más fácilmente, sabiendo que este acceso está bloqueado.

Mejorar la seguridad de inicio de sesión y el tiempo

Arriba y debajo de la opción `PermitRootLogin`, encontrará otros permisos, que detallaremos rápidamente aquí:

- `LoginGraceTime`se utiliza para aumentar, o disminuir, el tiempo que le queda a un usuario para iniciar sesión en la máquina. Esto se puede restringir de forma segura a 5 minutos, con lo siguiente:

```
LoginGraceTime 5m
```

- `MaxAuthTries` se utiliza para aumentar, o disminuir, el número de intentos permitidos a un usuario para autenticar correctamente en la máquina. Esto se puede restringir a 3 intentos usando:

```
MaxAuthTries 3
```

Denegar reenvío X11

A menos que necesite usar X11 sobre SSH, puede desactivarlo de manera segura usando la siguiente opción:

```
X11Forwarding no
```

Tenga en cuenta que deshabilitar X11 no deshabilita VNC a través de SSH, por ejemplo. En la mayoría de los casos, siempre es una buena idea deshabilitar los servicios innecesarios.

Restringir el inicio de sesión SSH a usuarios aprobados

Si está seguro de que solo usuarios específicos necesitan conectarse a través de SSH a su máquina, puede declararlos usando la opción `Permitir usuarios`. Todos los usuarios, y solo los usuarios, mencionados después de la opción podrán conectarse a través de SSH a la máquina.

```
AllowUsers jack backup betty
```

En el ejemplo que se muestra arriba, solo el usuario `jack`, `betty` y `backup` podrá usar SSH para conectarse a la máquina. A todos los demás usuarios se les negará el acceso. Por supuesto, debe usar esta opción con precaución y asegurarse de que el usuario predeterminado (o “a”) esté incluido ...

Reinicie el servidor SSH

Lo último que debe hacer, para asegurarse de que el servidor SSH tenga en cuenta la nueva configuración, es reiniciar el servidor SSH con el comando:

```
# /etc/rc.d/rc.sshd restart
```

Su nueva y más segura configuración de OpenSSH será efectiva ahora. ¡Felicidades!

Ver también

- Páginas de manual de OpenSSH (en línea)

Sources

- Escrito originalmente por User Noryungi
- Traducido al español por — *Antares_alf 2019/03/24 20:32 (UTC)*

howtos, security, ssh, author Noryungi