

# Hardening básico en Linux



24/11/23

---

Jose Almirón Lopez

## Índice

|   |           |
|---|-----------|
| <b>¿Qué es Hardening?</b>                     | <b>2</b>  |
| <b>Configuración de arranque</b>              | <b>2</b>  |
| Contraseña de arranque                        | 2         |
| Acceso Single User (usuario único) y permisos | 3         |
| <b>Configuración de usuarios y grupos</b>     | <b>4</b>  |
| Configuración de contraseñas                  | 4         |
| Configuración del entorno                     | 6         |
| <b>Configuración de acceso</b>                | <b>8</b>  |
| Configuración de sesión                       | 8         |
| Configuración del cifrado                     | 10        |
| <b>Configuración de sudo</b>                  | <b>11</b> |
| <b>Configuración de servicios</b>             | <b>12</b> |
| Configuración de sincronización de tiempo     | 12        |
| Servicios a desinstalar                       | 12        |
| <b>Configuración del sistema de ficheros</b>  | <b>13</b> |
| Definición de las particiones                 | 13        |
| Configuración de FSTAB                        | 14        |
| <b>Configuración de red</b>                   | <b>15</b> |
| Redirecciones y enrutado                      | 15        |
| Protocolos no habituales                      | 16        |
| <b>Configuración del FireWall</b>             | <b>16</b> |
| Configuración de UFW (Uncomplicated FireWall) | 17        |
| <b>Actualizaciones de software</b>            | <b>18</b> |
| Configuración de Advanced package tool        | 18        |
| <b>Registros</b>                              | <b>18</b> |
| Configurar "System Auditing"                  | 19        |
| Eventos a auditar                             | 19        |
| Configuración de Rsyslog                      | 19        |

## ¿Qué es Hardening?

El **Hardening**, o **bastionado**, es un proceso fundamental para fortalecer la seguridad de un sistema, adaptándolo al nivel necesario según su propósito y los datos que maneja. Este abarca todos los aspectos del sistema, desde el sistema de archivos hasta los usuarios y servicios. Dada su importancia, organismos internacionales han desarrollado guías periódicas de Hardening para los productos más comunes, proporcionando directrices esenciales para mantener sistemas seguros y protegidos contra posibles amenazas.

## Configuración de arranque

El proceso de arranque se destaca como uno de los aspectos más críticos, ya que una configuración insuficientemente segura podría permitir que un atacante con acceso al mismo comprometa el servidor, otorgándole la capacidad de ejecutar comandos como usuario con privilegios. A continuación, se detallan los pasos recomendados:

- Implementar una contraseña de arranque.
- Establecer una contraseña para el usuario root es fundamental para prevenir el acceso no autorizado durante el modo de recuperación.
- Restringir el acceso al archivo de configuración de inicio, ubicado en `"/boot/grub/grub.cfg"`, es una medida crucial para fortalecer la seguridad del sistema.

## Contraseña de arranque

Para establecer una contraseña de arranque del sistema, es necesario generar una contraseña cifrada utilizando el siguiente comando: **`"grub-mkpasswd-pbkdf2"`**.

```
jose@jose-almiron:~$ grub-mkpasswd-pbkdf2
Introduzca la contraseña:
Reintroduzca la contraseña:
El hash PBKDF2 de su contraseña es grub.pbkdf2.sha512.10000.0E471F51F435669632E35C962008A29407AC7610EBEB8CBC2477F6AD1F7939
04FD5C1DFF87FAE17807E792684D2A6F24D11230B7191AC5CCE8BB27FFB5C594AA.B01DD3BF48F11AD84669CE87F67E5D815A8B4C3539407A055D4730A
93E3F37E8019D12508D16DA831EC0407ADC857922DF0ABE1BB3B756296C2230D62142F075
```

Al ejecutar este comando, se te pedirá que ingreses una contraseña. Es importante destacar que la contraseña debe ser robusta, es decir, debe tener al menos 14 caracteres e incluir al menos una letra mayúscula, una minúscula, un número y un símbolo.

Después de ingresar la contraseña, el comando mostrará una salida con el hash correspondiente el cual debemos copiar. A continuación, es necesario crear un archivo en el directorio ***"/etc/grub.d/"*** que contenga lo siguiente.

```

jose@jose-almiron: /etc/grub.d
GNU nano 6.2                               init-pwd *
cat <<EOF
set superuser="root"
password_pbkdf2 root grub.pbkdf2.sha512.10000.0E471F51F435669632E35C962008A29407AC7610EBEB8CBC2477F6AD1F793904FD5C1DFF87F
EOF

```

Una vez creado, lo guardamos y le asignamos permisos de ejecución en el siguiente comando: ***"chmod +x nombre\_fichero"***.

```

jose@jose-almiron: /etc/grub.d$ sudo chmod +x init-pwd

```

A continuación, actualizamos GRUB con el comando ***"update-grub"***. Para verificar que la configuración se ha realizado correctamente, reiniciamos el servidor. Antes de iniciar, nos pedirá ingresar el usuario root y la contraseña correspondiente.

## Acceso Single User (usuario único) y permisos

El modo de usuario único se utiliza para recuperar el sistema en caso de fallo. Si no establecemos una contraseña para el usuario root, un atacante con acceso al servidor podría obtener privilegios. Para configurar una contraseña en el modo de usuario único, es necesario asignar una contraseña al usuario root mediante el comando ***'passwd'***.

```

root@jose-almiron:~# passwd
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
root@jose-almiron:~#

```

Para gestionar los permisos del archivo de configuración, es crucial limitar el acceso solo al usuario root. Esto se logra ejecutando el siguiente comando: ***"chmod 400 /boot/grub/grub.cfg"***.

```

root@jose-almiron:~# chmod 400 /boot/grub/grub.cfg
root@jose-almiron:~# stat /boot/grub/grub.cfg
  Fichero: /boot/grub/grub.cfg
  Tamaño: 12234      Bloques: 24      Bloque E/S: 4096  fichero regular
Dispositivo: 803h/2051d Nodo-i: 262162  Enlaces: 1
Acceso: (0400/-r-----) Uid: ( 0/   root)  Gid: ( 0/   root)
Acceso: 2023-11-23 11:44:40.325683357 +0100
Modificación: 2023-11-23 11:44:40.453688014 +0100
  Cambio: 2023-11-23 13:03:53.672875497 +0100
  Creación: 2023-10-05 13:37:07.183166664 +0200

```

## Configuración de usuarios y grupos

El siguiente punto aborda la configuración de usuarios y grupos, centrándose especialmente en aspectos relacionados con la configuración de sesiones, autenticación y privilegios de ejecución.

### Configuración de contraseñas

Para configurar contraseñas, es necesario establecer los siguientes parámetros:

- Complejidad
- Reutilización de contraseñas
- Almacenamiento
- Caducidad y cambio de contraseña

Para mejorar la complejidad de las contraseñas, es necesario instalar el paquete **libpam-pwquality**. Una vez instalado, editamos el archivo ***/etc/security/pwquality.conf*** y modificamos los siguientes valores:

- **minlen = 14**: Garantiza que la contraseña tenga al menos 14 caracteres.
- **minclass = 4**: Asegura que las contraseñas incluyan al menos un elemento de cada uno de los siguientes grupos: números, mayúsculas, minúsculas y símbolos.

En caso de estar comentadas estas líneas deben ser documentadas

```
# Minimum acceptable size for the new password (plus one if
# credits are not disabled which is the default). (See pam_cracklib manual.)
# Cannot be set to lower value than 6.
minlen = 14
```

```
#
# The minimum number of required classes of characters for the new
# password (digits, uppercase, lowercase, others).
minclass = 4
```

Para configurar el número de intentos fallidos antes de mostrar el mensaje de error, editamos el archivo ***/etc/pam.d/common-password*** y buscamos la línea que contiene **'retry'**. El valor asignado a la variable debe ser 3 o menor.

```
# here are the per-package modules (the "Primary" block)
password      requisite                                pam_pwquality.so retry=3
password      [success=2 default=ignore]               pam_unix.so obscure use_au
password      sufficient                                pam_sss.so use_authok
```

Para prevenir la reutilización de contraseñas en un período determinado, es esencial configurar el número de contraseñas anteriores que no se pueden utilizar; el valor recomendado es 5. Para realizar esta configuración, editamos el archivo ***"/etc/pam.d/common-password"*** y añadimos la línea ***'password required pam\_pwhistory.so remember=5'***. Además, se sugiere cifrar el almacenamiento de las contraseñas con un algoritmo robusto, y se recomienda utilizar **sha512**.

```
# here are the per-package modules (the "Primary" block)
password      requisite                                pam_pwquality.so retry=3
password      [success=2 default=ignore]               pam_unix.so obscure use_authtok try_first_pass sha512
password      sufficient                               pam_sss.so use_authtok
# here's the fallback if no module succeeds
password      requisite                                pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
password      required                                 pam_permit.so
# and here are more per-package modules (the "Additional" block)
password      optional                                 pam_gnome_keyring.so
# end of pam-auth-update config

password      required pam_pwhistory.so remember=5
```

El cambio de contraseña no puede establecerse en períodos de tiempo menores a un día, con el fin de evitar estrategias en las que, al expirar la contraseña actual, se cambia repetidamente hasta poder volver a utilizar la anterior. Además, se recomienda que el cambio de contraseña en un servidor se realice al menos cada 90 días, y en el caso de servidores críticos, cada 45 días. Para llevar a cabo esta configuración, editamos el archivo ***"/etc/login.defs"*** y establecemos los siguientes valores:

```
# Password aging controls:
#
#      PASS_MAX_DAYS   Maximum number of days a password may be used.
#      PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#      PASS_WARN_AGE   Number of days warning given before a password expires.
#
PASS_MAX_DAYS   90
PASS_MIN_DAYS    1
PASS_WARN_AGE    7
```

## Configuración del entorno

El bloqueo de cuentas por inactividad es crucial, ya que las cuentas inactivas pueden convertirse en un punto de entrada para un atacante. Se recomienda establecer este valor en un máximo de 30 días. Para configurar el valor predeterminado al crear cuentas, ejecutamos el comando `'useradd -D -f 30'` y para ajustarlo para usuarios existentes, utilizamos el comando `'chage --inactive 30 nombre_usuario'`

```
root@jose-almiron:~# useradd -D -f 30
root@jose-almiron:~# chage --inactive 30 root
root@jose-almiron:~# chage --inactive 30 jose
root@jose-almiron:~# useradd -D | grep INACTIVE
INACTIVE=30
root@jose-almiron:~# grep -E ^[:]+:[^!*] /etc/shadow | cut -d: -f1,7
root:30
jose:30
root@jose-almiron:~#
```

Un aspecto importante a considerar es el tiempo de espera por inactividad en la consola de comandos. Por defecto, esta característica no tiene ningún valor establecido, por lo que para habilitarla debemos editar el archivo `"/etc/bash.bashrc"` y agregar las siguientes líneas.

```
readonly TMOUT=900
export TMOUT
```

Este comando establece el tiempo de espera en 15 minutos. No establecer este parámetro podría resultar en accesos no autorizados si un usuario deja su puesto desatendido sin bloquear la sesión y con una conexión abierta.

Se debe configurar el sistema para prevenir ataques de fuerza bruta contra contraseñas. Se recomienda establecer un límite de 5 intentos fallidos antes de bloquear temporalmente la cuenta durante 15 minutos. Es crucial evitar bloqueos indefinidos para evitar posibles denegaciones de servicio. Esta configuración se realiza mediante la edición del archivo `"/etc/pam.d/common-auth"` y la adición de la correspondiente línea.

```
# end of pam-auth-update config

auth required pam_tally2.so onerr=tail audit silent deny=5 unlock_time=900
```



A continuación, editamos el archivo `"/etc/pam.d/common-account"` y agregamos las siguientes líneas, en caso de que no estén ya presentes

```
# here's the fallback if no module succeeds
account requisite                                pam_deny.so
account required                                pam_tally.so
# prime the stack with a positive return value if there is
```

Las cuentas de servicio deben configurarse para no permitir consolas interactivas, es decir, no pueden autenticarse en el sistema ni ejecutar comandos. Para lograr esto, es necesario establecer la característica de tipo de consola en **'nologin'** para todas estas cuentas. Modificamos el archivo `"/etc/passwd"` y cambiamos el valor actual por `/usr/sbin/nologin` en las cuentas de servicio correspondientes.

```
GNU nano 6.2 /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
```

El parámetro de configuración **'umask'** determina los permisos predeterminados asignados a un archivo creado por un usuario. El valor recomendado para este parámetro es 027, lo que significa control total para el creador, permisos de lectura y ejecución para los miembros del grupo, y acceso denegado para los demás. Para establecer estos valores, es necesario agregar o modificar el comando **'umask 027'** en los siguientes archivos:

- `/etc/bash.bashrc`
- `/etc/profile`
- `/etc/profile.d/*.sh`

El comando **"su"** facilita la ejecución de comandos en nombre de otro usuario, siendo útil en situaciones que requieren elevación de privilegios. Sin embargo, presenta la limitación de no ofrecer un control preciso sobre los comandos ejecutados; es una situación de todo o nada. Para restringir su uso, crearemos un grupo y añadiremos la siguiente línea al archivo `"/etc/pam.d/su"`.

```
# Uncomment this to force users to be a member of group wheel
# before they can use `su'. You can also add "group=foo"
# to the end of this line if you want to use a group other
# than the default "wheel" (but this may have side effect of
# denying "root" user, unless she's a member of "foo" or explicitly
# permitted earlier by e.g. "sufficient pam_rootok.so").
# (Replaces the `SU_WHEEL_ONLY' option from login.defs)
auth      required    pam_wheel.so use_uid group=sugroup
```



## Configuración de acceso

La mayoría de los servidores Linux son gestionados de forma remota a través de SSH. Para fortalecer el acceso por SSH al servidor, el primer paso consiste en modificar los permisos del archivo de configuración ubicado en `"/etc/ssh/sshd_config"`, otorgándole únicamente permisos de lectura y escritura al usuario root mediante el comando `"chmod 600 /etc/ssh/sshd_config"`.

## Configuración de sesión

Es necesario habilitar el modelo PAM (Pluggable Authentication Modules) para las conexiones SSH, lo cual proporciona un mayor control sobre el acceso de las cuentas de usuario. Para activarlo, editamos el archivo `"/etc/ssh/sshd_config"` y añadimos la línea `"UsePAM yes"`.

```
# PAM authentication, then enable this but set Pa
# and KbdInteractiveAuthentication to 'no'.
UsePAM yes
```

El parámetro `"MaxStartups"` indica el número máximo de conexiones no autenticadas concurrentes. Para prevenir problemas de denegación de servicio, se recomienda configurar este parámetro con los siguientes valores:

```
#UseDNS no
#PidFile /run/sshd.pid
MaxStartups 10:30:60
#PermitTunnel no
#ChrootDirectory none
```

- **10:** Número de conexiones no autenticadas permitidas antes de comenzar a rechazar nuevas conexiones.
- **30:** Porcentaje de conexiones que se empezarán a rechazar una vez superadas las primeras 10.
- **60:** Número máximo de conexiones posibles, al alcanzar este valor, todas las nuevas conexiones serán rechazadas.

Para gestionar el tiempo de expiración o timeout de la sesión SSH,

- **ClientAliveInterval:** Establece el intervalo de tiempo en el cual el sistema verifica la existencia de actividad.
- **ClientAliveCountMax:** Define el número de veces que el sistema debe preguntar y recibir respuesta de inactividad antes de cerrar la sesión.

```
#Compression delayed
ClientAliveInterval 300
ClientAliveCountMax 0
#UseDNS no
```

El parámetro "**MaxAuthTries**" establece el número máximo de intentos de autenticación permitidos por conexión. Se recomienda un valor de 4 para prevenir ataques de fuerza bruta.

```
#PermitRootLogin p
#StrictModes yes
MaxAuthTries 4
#MaxSessions 10
```

Para el parámetro "**IgnoreRhosts**", se establece que se ignorarán los archivos .rhosts y .shosts para la autenticación basada en RSA y host. Estos archivos permiten la autenticación a través del nombre, lo cual es potencialmente inseguro. Al configurar este parámetro, se refuerza la restricción de acceso a través de la autenticación mediante contraseña.

```
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
IgnoreRhosts yes
```

El parámetro "**HostBasedAuthentication**" permite la conexión basada en host de confianza. Establecer este parámetro en 'no', junto con IgnoreRhosts, requerirá que los usuarios se autenticuen siempre con contraseña.

```
# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
HostbasedAuthentication no
```

El parámetro "**PermitEmptyPasswords**" permite la autenticación sin contraseña. Para fortalecer la seguridad, estableceremos su valor en 'no'

```
# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
PermitEmptyPasswords no
```

Al establecer “**PermitRootLogin**” en 'no', se evita la autenticación directa como root, lo que obliga a obtener privilegios una vez autenticado como usuario. Esto proporciona una mayor trazabilidad y refuerza la seguridad del sistema.

```
#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
```

## Configuración del cifrado

SSH permite el uso de diversos tipos de cifrado, así como diferentes algoritmos MAC y algoritmos para el intercambio de claves. Es crucial seleccionar algoritmos sin vulnerabilidades conocidas y con claves lo suficientemente robustas para resistir compromisos. Para configurar estas opciones, agregaremos las líneas correspondientes a los parámetros Ciphers, MACS y KexAlgorithms.

### *Ciphers*

*chacha20-poly1305@openssh.com,aes256-gcm@openssh.com,aes128-gcm@openssh.com,aes256-ctr,aes192-ctr,aes128-ctr*

### *MACs*

*hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512,hmac-sha2-256*

### *KexAlgorithms*

*curve25519-sha256,[curve25519-sha256@libssh.org](https://libssh.org),diffie-hellman-group14-sha256,diffie-hellman-group16-sha512,diffie-hellman-group18-sha512,ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-nistp256,diffie-hellman-group-exchange-sha256*

```
# Ciphers and keying
Ciphers chacha20-poly1305@openssh.com,aes256-gcm@openssh.com,aes128-gcm@openssh.com,aes256-ctr,aes192-ctr,aes128-ctr

MACs hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512,hmac-sha2-256

KexAlgorithms curve25519-sha256,curve25519-sha256@libssh.org,diffie-hellman-group14-sha256,diffie-hellman-group16-sha512,diffie-hellman-group18-sha512,ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-nistp256,diffie-hellman-group-exchange-sha256

# Logging
```

## Configuración de sudo

Sudo puede ser configurado para permitir su uso sólo desde un pseudoterminal (**pty**). Esta medida evita que los atacantes ejecuten comandos con sudo y los dejen en segundo plano, ya que al forzar la existencia de un pseudoterminal **pty**, al cerrar el proceso principal también se cerraría el que se ejecuta en segundo plano. Para verificar o modificar este parámetro, editamos el archivo “*/etc/sudoers*” y añadimos la siguiente línea

```
# See the man page for details on how to write a s
#
Defaults      env_reset
Defaults      mail_badpass
Defaults      secure_path="/usr/local/sbin:/usr/
Defaults      use_pty
Defaults      logfile="/var/log/sudo.log"
```

La ejecución de sudo puede configurarse a nivel individual por usuario o por grupo. La diferencia radica en que, si es a nivel de usuario individual, la línea comienza con el nombre del usuario, mientras que si es por grupo, la línea inicia con '%' seguido del nombre del grupo.

```
# Host alias specification
Host_Alias BBDD = 192.168.0.23
# User alias specification
User_Alias BDOP = usuario,mysqlUser
# Cmnd alias specification
Cmnd_Alias BDBKP = /bin/mysqldump, /usr/bin/backup.sh
# User privilege specification
root    ALL=(ALL:ALL) ALL

BDOP    BBDD=(root:root) NOPASSWD:BDBKP

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
```

## Configuración de servicios

Se describe de manera genérica cómo deben configurarse los servicios en un servidor de propósito general. Dada la amplia gama de servicios que se pueden configurar en un servidor, nos centraremos en aquellos que deben configurarse en todos los servidores, independientemente de su propósito específico.

### Configuración de sincronización de tiempo

Es crucial que todos los sistemas en nuestra red mantengan una sincronización de reloj precisa. Podemos habilitar este servicio mediante el siguiente comando

```
[Time]
NTP=0.ubuntu.pool.ntp.org 1.ubuntu.pool.ntp.org 2.ubuntu.pool.ntp.org
FallbackNTP=ntp.ubuntu.com 3.ubuntu.pool.ntp.org
RootDistanceMaxSec=1
#PollIntervalMinSec=32
```

```
root@jose-almiron:~# systemctl enable systemd-timesyncd.service
root@jose-almiron:~# nano /etc/systemd/timesyncd.conf
root@jose-almiron:~# systemctl start systemd-timesyncd.service
root@jose-almiron:~# timedatectl set-ntp true
root@jose-almiron:~#
```

### Servicios a desinstalar

Existen una serie de servicios que, por norma general, no deberían estar instalados en un servidor. Estos incluyen:

- X Windows System
- Avahi
- Todo servicio no relacionado con el objetivo

Para obtener un listado de los servicios instalados, ejecutamos el comando **"service --status-all"**. Esto mostrará una lista de los servicios instalados e indicará cuáles están actualmente en ejecución.

```
root@server:~# service --status-all
[ + ] apparmor
[ + ] apport
[ - ] console-setup.sh
[ + ] cron
[ - ] cryptdisks
[ - ] cryptdisks-early
[ + ] dbus
[ - ] grub-common
[ - ] hwclock.sh
[ - ] irqbalance
[ - ] iscsid
[ - ] keyboard-setup.sh
[ + ] kmod
[ - ] lvm2
[ - ] lvm2-lvmpolld
[ + ] multipath-tools
[ - ] open-iscsi
[ - ] open-vm-tools
[ + ] plymouth
[ + ] plymouth-log
[ + ] procps
```

## Configuración del sistema de ficheros

La configuración del sistema de archivos abarca particiones, formatos y puntos de montaje, y puede llevarse a cabo durante la instalación del sistema o posteriormente, mediante la modificación de particiones existentes. Se recomienda realizar esta configuración durante la instalación para evitar la necesidad de mover datos críticos y sectores especiales del disco, minimizando así el riesgo de corrupción del sistema de archivos.

### Definición de las particiones

Se sugiere utilizar particiones separadas para los siguientes puntos de montaje:

- **/tmp**: un directorio accesible y modificable por todos los usuarios del sistema, utilizado para almacenamiento temporal.
- **/var**: empleado por demonios y otros servicios del sistema para almacenar temporalmente datos dinámicos.
- **/var/tmp**: accesible en modo lectura, escritura y ejecución por todos los usuarios; se recomienda tenerlo en una partición separada.

- **/var/log y /var/log/audit:** particulares, ya que se busca evitar el agotamiento de recursos del sistema por el aumento no previsto de datos almacenados, además de proteger de manera específica los archivos de registro y auditoría del sistema.
- **/home:** es necesario crear una partición aparte para evitar el consumo de recursos por parte de usuarios locales y restringir las acciones que pueden realizar los usuarios en sus directorios personales.

El formato recomendado para todas las particiones en **ext4**, la última versión del sistema de archivos nativo de Linux, que garantiza características de seguridad avanzadas.

| Filesystem setup    |         |          |                  |                |
|---------------------|---------|----------|------------------|----------------|
| FILE SYSTEM SUMMARY |         |          |                  |                |
| MOUNT POINT         | SIZE    | TYPE     | DEVICE           | TYPE           |
| [ /                 | 13.998G | new ext4 | new partition of | local disk ▶ ] |
| [ /home             | 10.000G | new ext4 | new partition of | local disk ▶ ] |
| [ /tmp              | 1.000G  | new ext4 | new partition of | local disk ▶ ] |
| [ /var              | 10.000G | new ext4 | new partition of | local disk ▶ ] |
| [ /var/log          | 1.000G  | new ext4 | new partition of | local disk ▶ ] |
| [ /var/log/audit    | 1.000G  | new ext4 | new partition of | local disk ▶ ] |
| [ /var/tmp          | 1.000G  | new ext4 | new partition of | local disk ▶ ] |
| [ SWAP              | 2.000G  | new swap | new partition of | local disk ▶ ] |

## Configuración de FSTAB

En el archivo **"/etc/fstab"** se encuentra la configuración inicial de carga de todas las particiones al inicio del sistema, junto con las opciones definidas. Desde el punto de vista de la seguridad, es esencial establecer ciertas opciones en las particiones para prevenir el uso indebido por parte de usuarios malintencionados. En particular, para las particiones destinadas a almacenar archivos temporalmente, se recomienda configurarlas con las opciones **'nodev'**, **'nosuid'** y **'noexec'**. Esto evita que se monten sistemas de archivos en estas particiones y que se ejecuten archivos desde ellas

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
# defaults = rw, suid, dev, exec, auto, nouser, and async
#
# <file system> <mount point> <type> <options> <dump> <pass>
/dev/disk/by-uuid/7f9fb162-0b94-4903-9db5-5ac9772a9fd4 none swap sw 0 0
# / was on /dev/sda9 during curtin installation
/dev/disk/by-uuid/d2c32d66-7bb0-4540-9a42-5eafe25b0f9f / ext4 defaults 0 0
# /var was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/b26c3642-58e2-4304-a0dd-e01b5639fc35 /var ext4 defaults 0 0
# /tmp was on /dev/sda3 during curtin installation
/dev/disk/by-uuid/b9034008-a70f-4e64-a90b-9c74a8694b6d /tmp ext4 rw,auto,nouser,async,nodev,nosuid,noexec 0 0
# /var/log was on /dev/sda4 during curtin installation
/dev/disk/by-uuid/01a42114-bd5e-40de-b623-181ec3324563 /var/log ext4 defaults 0 0
# /var/log/audit was on /dev/sda5 during curtin installation
/dev/disk/by-uuid/e3933579-6b3d-46bc-8a30-c5b38f6399c4 /var/log/audit ext4 defaults 0 0
# /var/tmp was on /dev/sda6 during curtin installation
/dev/disk/by-uuid/6e90052d-86ce-4663-982c-67d8d62b3328 /var/tmp ext4 rw,auto,nouser,async,nodev,nosuid,noexec 0 0
# /home was on /dev/sda7 during curtin installation
/dev/disk/by-uuid/6c39a950-32e1-4f95-aeb8-073819c5eeac /home ext4 rw,auto,nouser,async,suid,exec,nodev 0 0
# /dev/shm
tmpfs /dev/shm tmpfs rw,auto,nouser,async,nodev,nosuid,noexec 0 0
```



## Configuración de red

Vamos a configurar los parámetros de red del servidor, centrándonos principalmente en evitar que el servidor actúe como un enrutador de tráfico. Además, nos aseguraremos de verificar el origen y destino de la comunicación, y registramos acciones que puedan ser sospechosas de algún tipo de ataque. Estas configuraciones se encuentran definidas en el archivo *"/etc/sysctl.conf"*

### Redirecciones y enrutado

Dado que la función principal del servidor no será enrutar tráfico, es esencial deshabilitar todas las opciones de redirección de paquetes.

- Deshabilitar redirección ICMP

```
sysctl -w net.ipv4.conf.all.send_redirects=0
sysctl -w net.ipv4.conf.default.send_redirects=0
sysctl -w net.ipv4.route.flush=1
```
- Deshabilitar IP Forwarding

```
sysctl -u net.ipv4.ip_forward=0
```
- Deshabilitar respuesta ICMP Broadcast

```
sysctl -w net.ipv4.icmp_echo_ignore_broadcasts=1
sysctl -w net.ipv4.route.flush=1
```
- Registro solo de paquetes que cumplan estándares

```
sysctl -w net.ipv4.icmp_ignore_bogus_error_responses=1
sysctl -w net.ipv4.route.flush=1
```
- Aseguramiento del origen

```
sysctl -w net.ipv4.conf.all.rp_filter=1
sysctl -w net.ipv4.conf.default.rp_filter=1
sysctl -w net.ipv4.route.flush=1
```
- TCP SYN Cookies

```
sysctl -w net.ipv4.tcp_syncookies=1
sysctl -w net.ipv4.route.flush=1
```

## Protocolos no habituales

Existen protocolos de red específicos para ciertos servicios que, si no se van a utilizar, deben deshabilitarse, ya que podrían representar un riesgo. Estos incluyen:

- **Datagram Congestion Control Protocol:** utilizado para streaming.
- **Stream Control Transmission Protocol:** empleado para la comunicación orientada a mensajes.
- **Reliable Datagram Sockets:** para comunicaciones con baja latencia y gran ancho de banda.
- **Transparent Inter-Process Communication:** diseñado para proporcionar comunicación entre los nodos de un clúster.

Para configurar estos protocolos, es necesario crear un archivo con su acrónimo en el directorio `"/etc/modprobe.d/"`. Dentro de cada archivo, debe incluirse la línea **"install acrónimo /bin/true"**.

```
root@server:/etc/modprobe.d# echo "install tipc /bin/true" > tipc.conf
root@server:/etc/modprobe.d# echo "install rds /bin/true" > rds.conf
root@server:/etc/modprobe.d# echo "install sctp /bin/true" > sctp.conf
root@server:/etc/modprobe.d# echo "install dccp /bin/true" > dccp.conf
root@server:/etc/modprobe.d# ls -l
total 52
-rw-r--r-- 1 root root 154 ago 21 00:55 amd64-microcode-blacklist.conf
-rw-r--r-- 1 root root 325 ago 17 2021 blacklist-ath_pci.conf
-rw-r--r-- 1 root root 1518 ago 17 2021 blacklist.conf
-rw-r--r-- 1 root root 210 ago 17 2021 blacklist-firewire.conf
-rw-r--r-- 1 root root 677 ago 17 2021 blacklist-framebuffer.conf
-rw-r--r-- 1 root root 583 ago 17 2021 blacklist-rare-network.conf
-rw-r--r-- 1 root root 23 nov 23 19:52 dccp.conf
-rw-r--r-- 1 root root 154 nov 15 01:03 intel-microcode-blacklist.conf
-rw-r--r-- 1 root root 347 ago 17 2021 iwlwifi.conf
-rw-r--r-- 1 root root 379 abr 11 2023 mdadm.conf
-rw-r--r-- 1 root root 22 nov 23 19:52 rds.conf
-rw-r--r-- 1 root root 23 nov 23 19:52 sctp.conf
-rw-r--r-- 1 root root 23 nov 23 19:51 tipc.conf
root@server:/etc/modprobe.d#
```

## Configuración del FireWall

Los firewalls basados en host proporcionan servicios de control de tráfico tanto interno como externo, deteniendo intrusiones y ofreciendo un sólido método de control de acceso. Para simplificar esta configuración, se recomienda utilizar **uFW** (uncomplicated Firewall).

## Configuración de UFW (Uncomplicated FireWall)

Como norma general, se denegará todas las conexiones entrantes y salientes, y se habilitarán únicamente aquellas conexiones que sean estrictamente necesarias. Para instalarlo, utilizamos el comando **"apt install ufw"**, y comenzamos aplicando la política de denegar cualquier conexión con el comando **"ufw default deny incoming"**.

```
root@server:/etc/modprobe.d# ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
root@server:/etc/modprobe.d# ufw default deny outgoing
Default outgoing policy changed to 'deny'
(be sure to update your rules accordingly)
root@server:/etc/modprobe.d# ufw default deny routed
Default routed policy changed to 'deny'
(be sure to update your rules accordingly)
root@server:/etc/modprobe.d# _
```

A continuación, es necesario habilitar el tráfico loopback, pero aislarlo del resto de interfaces. Para lograrlo, ejecutamos

```
root@server:~# ufw allow in on lo
Rules updated
Rules updated (v6)
root@server:~# ufw deny in from 127.0.0.0/8
Rules updated
root@server:~# _
```

A continuación, configuraremos las conexiones entrantes y salientes permitidas. Las salientes coinciden con los servicios a los que queremos acceder desde el servidor, mientras que las entrantes se corresponden con los servicios que ofrece el servidor.

- Para las conexiones salientes se ejecuta el comando

*ufw allow out to <IP o any> port <puerto>*

- Para las conexiones entrantes, es necesario crear una regla por cada puerto que ofrezca un servicio, de manera que no se rechacen las conexiones. El comando es el siguiente:

*ufw allow in <Puerto>/(tcp o udp)*

Para verificar el estado, ejecutamos “**ufw status**”.

```
root@server:~# ufw enable
Firewall is active and enabled on system startup
root@server:~# ufw status
Status: active

To Action From
--
Anywhere on 10 ALLOW Anywhere
Anywhere DENY 127.0.0.0/8
Anywhere (v6) on 10 ALLOW Anywhere (v6)
root@server:~#
```

## Actualizaciones de software

En general, las distribuciones basadas en Debian, y en particular Ubuntu, utilizan el sistema Advanced Package Tool (**APT**) para la actualización de paquetes de software. Estas actualizaciones deben estar definidas con las políticas establecidas, y se recomienda llevar a cabo actualizaciones de forma periódica.

### Configuración de Advanced package tool

En este aspecto, es crucial configurar los repositorios con fuentes confiables y asegurarse de que estén configuradas las claves GPG para verificar la integridad de los paquetes durante la instalación. Los repositorios se configuran en el archivo “**/etc/apt/sources.list**”, y por defecto, incluye los repositorios oficiales. Es posible añadir repositorios adicionales. De los repositorios oficiales, existen varios tipos: “**Main Restricted**”, “**Universe**” y “**Multiverse**”. Se recomienda habilitar únicamente los repositorios “Main Restricted”, ya que son los únicos cien por cien probados y soportados

## Registros

Los registros de actividad son una parte crucial para depurar configuraciones y errores en el sistema, así como para investigar posibles incidentes de seguridad. Es necesario registrar todo tipo de acciones que puedan poner en peligro el sistema, como el acceso a archivos de configuración, modificaciones en ficheros y configuraciones, conexiones de red, entre otros.

## Configurar “System Auditing”

Por defecto, el sistema de auditoría (System Auditing) auditará los accesos denegados por SELinux, los AVC, los inicios de sesión del sistema, las modificaciones de cuentas y eventos de autenticación. Los eventos se registran en `“/var/log/audit/audit.log”`. Para instalar el servicio de auditoría, es necesario ejecutar `“apt install audit”`, y para iniciar el servicio, `“systemctl enable --now audit”`.

## Eventos a auditar

En líneas generales, los eventos a auditar dependen de la criticidad del servicio y de los datos que maneje. A continuación, presentamos una lista de eventos que se consideran importantes para registrar:

- Modificaciones de usuarios y grupos.
- Cambios en la configuración de red.
- Intentos de inicio de sesión (tanto válidos como fallidos) y cierres de sesión.
- Modificaciones de permisos en archivos de configuración del sistema.
- Intentos fallidos de acceso.
- Comandos ejecutados con sudo y del usuario root.
- Eliminación de archivos.
- Habilitación y deshabilitación de módulos del kernel.

## Configuración de Rsyslog

Es un software que facilita el envío de registros a un servidor centralizado a través de una conexión TCP. Admite varios formatos para almacenar los registros en bases de datos y ofrece soporte para cifrado. Si se planea centralizar los registros de eventos, es necesario instalarlo con el comando `“apt install rsyslog”` y activarlo con `“systemctl enable --now rsyslog”`. La configuración se encuentra en los archivos `“/etc/rsyslog.conf”` y `“/etc/rsyslog.d/.conf”`. Se recomienda restringir el acceso de los usuarios a estos archivos, ya que pueden contener información confidencial. Para lograrlo, se debe agregar `'$FileCreateMode 0640'` a la configuración. Para especificar los servidores a los que se deben enviar los registros, se agrega la línea `'*. * @@<IP o nombre>'`. El archivo de configuración `/etc/rsyslog.conf` se vería así:

```
#  
# Set the default permissions for all log files.  
#  
$FileOwner syslog  
$FileGroup adm  
FileCreateMode 0640  
$DirCreateMode 0755  
$Umask 0022  
$PrivDropToUser syslog  
$PrivDropToGroup syslog  
  
#  
# Servers  
#  
  
*,* @@192.168.0.55
```