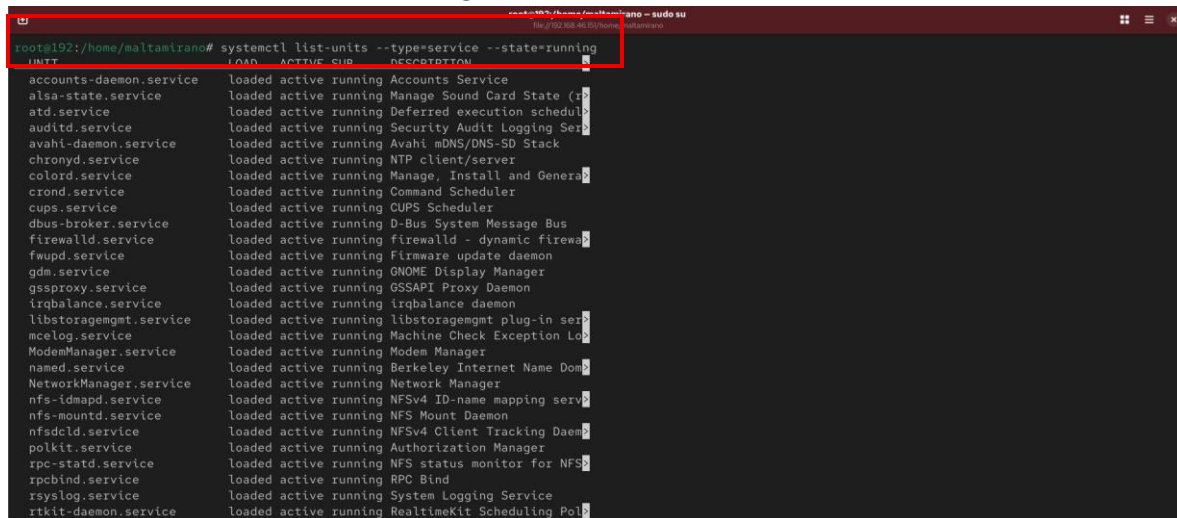


Auditoría y endurecimiento inicial del servidor

1. Desactivación de servicios innecesarios (3 Puntos)

- Analiza el estado actual de los servicios activos y desactiva todos aquellos que no sean requeridos en un entorno de producción básico.
- Deja activo únicamente SSH.
- Entrega: los comandos ejecutados y el resultado de `systemctl list-units --type=service --state=running` antes y después de la desactivación.



```
root@192:/home/maltamirano# systemctl list-units --type=service --state=running
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
accounts-daemon.service            loaded active running Accounts Service
alsa-state.service                 loaded active running Manage Sound Card State (r
atd.service                        loaded active running Deferred execution schedu
auditd.service                    loaded active running Security Audit Logging Ser
avahi-daemon.service               loaded active running Avahi mDNS/DNS-SD Stack
chronyd.service                   loaded active running NTP client/server
colord.service                     loaded active running Manage, Install and Genera
crond.service                     loaded active running Command Scheduler
cups.service                       loaded active running CUPS Scheduler
dbus-broker.service               loaded active running D-Bus System Message Bus
firewalld.service                 loaded active running firewalld - dynamic firewa
fwupd.service                     loaded active running Firmware update daemon
gdm.service                       loaded active running GNOME Display Manager
gssproxy.service                  loaded active running GSSAPI Proxy Daemon
irqbalance.service               loaded active running irqbalance daemon
libstoragemgmt.service            loaded active running libstoragemgmt plug-in ser
mcelog.service                    loaded active running Machine Check Exception Lo
ModemManager.service             loaded active running Modem Manager
named.service                     loaded active running Berkeley Internet Name Dom
NetworkManager.service           loaded active running Network Manager
nfs-idmapd.service                loaded active running NFSv4 ID-name mapping serv
nfs-mountd.service                loaded active running NFS Mount Daemon
nfsdclld.service                 loaded active running NFSv4 Client Tracking Daem
polkit.service                    loaded active running Authorization Manager
rpc-statd.service                 loaded active running NFS status monitor for NFS
rpcbind.service                   loaded active running RPC Bind
rsyslog.service                   loaded active running System Logging Service
rtkit-daemon.service              loaded active running RealtimeKit Scheduling Pol
```

En la imagen se muestra la ejecución del comando **`systemctl list-units --type=service --state=running`**, que lista todos los servicios activos en el servidor. Este registro inicial sirve como evidencia previa para el Requerimiento 1, permitiendo identificar cuáles procesos se encuentran en ejecución. A partir de esta información se determinarán los servicios innecesarios que serán detenidos y deshabilitados, manteniendo solo los esenciales como `sshd` para el acceso remoto seguro.



```
root@192:/home/maltamirano# sudo systemctl stop cups
root@192:/home/maltamirano# sudo systemctl stop avahi-daemon
Stopping 'avahi-daemon.service', but its triggering units are still active:
avahi-daemon.socket
root@192:/home/maltamirano#
```

En la imagen se muestran los comandos ejecutados para detener los servicios innecesarios **`cups` y `avahi-daemon`** en el servidor. Esta acción forma parte del Requerimiento 1, orientada a reducir la superficie de ataque deshabilitando procesos no esenciales. El mensaje indica que, aunque `avahi-daemon` fue detenido, su socket aún sigue activo, lo que requerirá deshabilitarlo posteriormente para evitar su reinicio automático.

```
root@192:/home/maltamirano# sudo systemctl stop avahi-daemon
root@192:/home/maltamirano# sudo systemctl stop ModemManager
root@192:/home/maltamirano# sudo systemctl stop nfs-idmap
Failed to stop nfs-idmap.service: Unit nfs-idmap.service not loaded.
root@192:/home/maltamirano# sudo systemctl stop nfs-mountd
root@192:/home/maltamirano# sudo systemctl stop nfsdclld
root@192:/home/maltamirano# sudo systemctl stop rpc-statd
root@192:/home/maltamirano# sudo systemctl stop rpcbind
root@192:/home/maltamirano# sudo systemctl stop smb
root@192:/home/maltamirano# sudo systemctl disable avahi-daemon
root@192:/home/maltamirano# sudo systemctl disable ModemManager
Removed '/etc/systemd/system/multi-user.target.wants/modemmanager.service'.
Removed '/etc/systemd/system/dbus-org.freedesktop.ModemManager1.service'.
root@192:/home/maltamirano# sudo systemctl disable nfs-mountd
The unit files have no installation config (WantedBy=, RequiredBy=, UpheldBy=,
Also=, or Alias= settings in the [Install] section, and DefaultInstance= for
template units). This means they are not meant to be enabled or disabled using systemctl.

Possible reasons for having these kinds of units are:
- A unit may be statically enabled by being symlinked from another unit's
  .wants/, .requires/, or .upholds/ directory.
- A unit's purpose may be to act as a helper for some other unit which has
  a requirement dependency on it.
- A unit may be started when needed via activation (socket, path, timer,
  D-Bus, udev, scripted systemctl call, ...).
- In case of template units, the unit is meant to be enabled with some
  instance name specified.
root@192:/home/maltamirano#
```

```
template units). This means they are not meant to be enabled or disabled using systemctl.

Possible reasons for having these kinds of units are:
- A unit may be statically enabled by being symlinked from another unit's
  .wants/, .requires/, or .upholds/ directory.
- A unit's purpose may be to act as a helper for some other unit which has
  a requirement dependency on it.
- A unit may be started when needed via activation (socket, path, timer,
  D-Bus, udev, scripted systemctl call, ...).
- In case of template units, the unit is meant to be enabled with some
  instance name specified.
root@192:/home/maltamirano# sudo systemctl disable rpc-statd
The unit files have no installation config (WantedBy=, RequiredBy=, UpheldBy=,
Also=, or Alias= settings in the [Install] section, and DefaultInstance= for
template units). This means they are not meant to be enabled or disabled using systemctl.

Possible reasons for having these kinds of units are:
- A unit may be statically enabled by being symlinked from another unit's
  .wants/, .requires/, or .upholds/ directory.
- A unit's purpose may be to act as a helper for some other unit which has
  a requirement dependency on it.
- A unit may be started when needed via activation (socket, path, timer,
  D-Bus, udev, scripted systemctl call, ...).
- In case of template units, the unit is meant to be enabled with some
  instance name specified.
root@192:/home/maltamirano# sudo systemctl disable rpcbind
Removed '/etc/systemd/system/multi-user.target.wants/rpcbind.service'.
root@192:/home/maltamirano# sudo systemctl disable smb
Removed '/etc/systemd/system/multi-user.target.wants/smb.service'.
root@192:/home/maltamirano#
```

En esta imagen se procedió a la detención y deshabilitación de servicios no esenciales como **avahi-daemon**, **ModemManager**, **rpcbind**, **smb** y otros relacionados con NFS, con el objetivo de reducir la superficie de ataque del servidor. La ejecución de estos comandos permitió que solo permanezcan activos sshd y los servicios básicos necesarios para el funcionamiento del sistema.

Durante el proceso de endurecimiento se deshabilitaron servicios no necesarios para el entorno de producción, como **cups**, **avahi-daemon**, **ModemManager**, **rpcbind**, **smb** y procesos relacionados con **NFS**, que corresponden a funciones de **impresión**, **descubrimiento de red** y **compartición de recursos no utilizadas en este servidor**. Se mantuvieron activos, además de **SSHD**, solo los servicios esenciales para el correcto funcionamiento y administración del sistema, como **systemd**, **dbus**, **chronyd**, **firewalld** y **NetworkManager**, ya que su desactivación afectaría la conectividad, la seguridad y la estabilidad del servidor.

```
root@192:/home/maltamirano# systemctl list-units --type=service --state=running
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
accounts-daemon.service	loaded	active	running	Accounts Service
alsa-state.service	loaded	active	running	Manage Sound Card State (restore and store)
atd.service	loaded	active	running	Deferred execution scheduler
auditd.service	loaded	active	running	Security Audit Logging Service
chronyd.service	loaded	active	running	NTP client/server
colord.service	loaded	active	running	Manage, Install and Generate Color Profiles
crond.service	loaded	active	running	Command Scheduler
dbus-broker.service	loaded	active	running	D-Bus System Message Bus
firewalld.service	loaded	active	running	firewalld - dynamic firewall daemon
fprintd.service	loaded	active	running	Fingerprint Authentication Daemon
fwupd.service	loaded	active	running	Firmware update daemon
gdm.service	loaded	active	running	GNOME Display Manager
getty@tty3.service	loaded	active	running	Getty on tty3
gssproxy.service	loaded	active	running	GSSAPI Proxy Daemon
irqbalance.service	loaded	active	running	irqbalance daemon
libstoragemgmt.service	loaded	active	running	libstoragemgmt plug-in server daemon
mcelog.service	loaded	active	running	Machine Check Exception Logging Daemon
named.service	loaded	active	running	Berkeley Internet Name Domain (DNS)
NetworkManager.service	loaded	active	running	Network Manager
polkit.service	loaded	active	running	Authorization Manager
rsyslog.service	loaded	active	running	System Logging Service
rtkit-daemon.service	loaded	active	running	RealtimeKit Scheduling Policy Service
sshd.service	loaded	active	running	OpenSSH server daemon
sssd-kcm.service	loaded	active	running	SSSD Kerberos Cache Manager
switcheroo-control.service	loaded	active	running	Switcheroo Control Proxy service
systemd-journald.service	loaded	active	running	Journal Service
systemd-logind.service	loaded	active	running	User Login Management
systemd-udev.service	loaded	active	running	Rule-based Manager for Device Events and Files

En esta imagen se muestra nuevamente la ejecución del comando **systemctl list-units --type=service --state=running** tras realizar la detención y deshabilitación de algunos servicios. Esta salida sirve como evidencia posterior en el Requerimiento 1, permitiendo comparar el estado actual con el listado inicial. Con esta verificación se confirma qué servicios siguen activos y si efectivamente se han reducido los procesos innecesarios, manteniendo operativos únicamente los esenciales para el servidor.

```
root@192:/home/maltamirano# systemctl list-units --type=socket --state=active
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
cockpit.socket	loaded	active	listening	Cockpit Web Service Socket
dbus.socket	loaded	active	running	D-Bus System Message Bus Socket
dm-event.socket	loaded	active	listening	Device-mapper event daemon FIFOs
iscsid.socket	loaded	active	listening	Open-iscsi iscsid Socket
iscsiuto.socket	loaded	active	listening	Open-iscsi iscsiuto Socket
lvm2-lvmpolld.socket	loaded	active	listening	LVM2 poll daemon socket
pcscd.socket	loaded	active	listening	PC/SC Smart Card Daemon Activation Socket
sshd-unix-local.socket	loaded	active	listening	OpenSSH Server Socket (systemd-ssh-generator, AF_UNIX Local)
sshd-vsock.socket	loaded	active	listening	OpenSSH Server Socket (systemd-ssh-generator, AF_VSOCK)
sssd-kcm.socket	loaded	active	running	SSSD Kerberos Cache Manager responder socket
systemd-bootctl.socket	loaded	active	listening	Boot Entries Service Socket
systemd-coredump.socket	loaded	active	listening	Process Core Dump Socket
systemd-creds.socket	loaded	active	listening	Credential Encryption/Decryption
systemd-hostnamed.socket	loaded	active	listening	Hostname Service Socket
systemd-initctl.socket	loaded	active	listening	initctl Compatibility Named Pipe
systemd-journald-dev-log.socket	loaded	active	running	Journal Socket (/dev/log)
systemd-journald.socket	loaded	active	running	Journal Sockets
systemd-rfkill.socket	loaded	active	listening	Load/Save RF Kill Switch Status /dev/rfkill Watch
systemd-sysext.socket	loaded	active	listening	System Extension Image Management
systemd-udev-control.socket	loaded	active	running	udev Control Socket
systemd-udev-kernel.socket	loaded	active	running	udev Kernel Socket
systemd-userdbd.socket	loaded	active	running	User Database Manager Socket

Legend: LOAD → Reflects whether the unit definition was properly loaded.
ACTIVE → The high-level unit activation state, i.e. generalization of SUB.
SUB → The low-level unit activation state, values depend on unit type.

22 loaded units listed.

En la imagen se visualiza la salida del comando **systemctl list-units --type=socket --state=active**, que muestra los sockets actualmente activos en el sistema. Esta revisión es parte de la verificación final del Requerimiento 1, ya que permite identificar sockets que podrían reactivar servicios previamente detenidos.

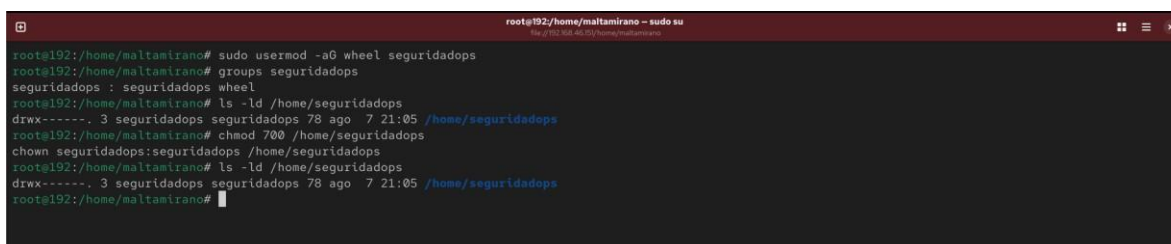
2. Configuración de usuarios y permisos (3 Puntos)

- Crea un nuevo usuario llamado **seguridadops**, asigna una contraseña segura, agrégalo al grupo **sudo**, y asegúrate de que su carpeta personal tenga los permisos adecuados.

A terminal window titled 'root@192:/home/maltamirano - sudo su' showing the process of creating a new user. The user runs 'sudo useradd -m -s /bin/bash seguridadops' and then 'sudo passwd seguridadops'. The password command prompts for a new password, which is rejected as being in the dictionary. After a second attempt, the password is accepted, and the user is ready for use.

```
root@192:/home/maltamirano# sudo useradd -m -s /bin/bash seguridadops
root@192:/home/maltamirano# sudo passwd seguridadops
Nueva contraseña:
CONTRASEÑA INCORRECTA: La contraseña no supera la verificación de diccionario - está basada en una palabra del diccionario
Vuelva a escribir la nueva contraseña:
Las contraseñas no coinciden.
passwd: Error de manipulación del testigo de autenticación
passwd: no se ha cambiado la contraseña
root@192:/home/maltamirano# sudo passwd seguridadops
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
root@192:/home/maltamirano#
```

En esta imagen se evidencia la creación del usuario **seguridadops** mediante el comando **useradd -m -s /bin/bash**, especificando la creación de su carpeta personal y la asignación de la **shell bash**. Después, se configuró correctamente una contraseña que cumplía con los requisitos, quedando el usuario listo para su uso con credenciales seguras, tal como exige el Requerimiento 2.

A terminal window titled 'root@192:/home/maltamirano - sudo su' showing the configuration of the 'seguridadops' user. The user runs 'sudo usermod -aG wheel seguridadops' to add the user to the wheel group. Then, 'groups seguridadops' is run to verify the group membership. Finally, 'ls -ld /home/seguridadops' and 'chown seguridadops:seguridadops /home/seguridadops' are used to set permissions and ownership on the user's home directory.

```
root@192:/home/maltamirano# sudo usermod -aG wheel seguridadops
root@192:/home/maltamirano# groups seguridadops
seguridadops : seguridadops wheel
root@192:/home/maltamirano# ls -ld /home/seguridadops
drwx----- 3 seguridadops seguridadops 78 ago  7 21:05 /home/seguridadops
root@192:/home/maltamirano# chmod 700 /home/seguridadops
chown seguridadops:seguridadops /home/seguridadops
root@192:/home/maltamirano# ls -ld /home/seguridadops
drwx----- 3 seguridadops seguridadops 78 ago  7 21:05 /home/seguridadops
root@192:/home/maltamirano#
```

En esta imagen se muestra cómo el usuario **seguridadops** fue agregado al grupo **wheel**, otorgándole privilegios de administración. Se verificó su pertenencia al grupo mediante el comando **groups**. Además, se revisaron los permisos de su carpeta personal, ajustándolos a **700** y asegurando que el propietario y grupo sean **seguridadops**. Con estos pasos, el usuario queda correctamente configurado con privilegios de sudo y con una carpeta personal protegida, cumpliendo las exigencias del Requerimiento 2 en cuanto a seguridad y control de acceso.

- Luego, elimina cualquier usuario innecesario creado por defecto.

```
root@192:/home/maltamirano - sudo su
root@192:/home/maltamirano# cut -d: -f1 /etc/passwd
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
operator
gnome-initial-setup
sshd
chrony
dnsmasq
tcpdump
maltamirano
smtpd
smtpp
named
ipc
ipccuser
seguridadops
usuario1
usuario2
root@192:/home/maltamirano#
```

Al ejecutar el comando **cut -d: -f1 /etc/passwd** nos hemos percatado que hay dos usuarios que no deben estar en estos privilegios (usuario1 y 2) así procederemos a eliminarlos.

```
root@192:/home/maltamirano - sudo su
root@192:/home/maltamirano# sudo userdel -r usuario1
root@192:/home/maltamirano# sudo userdel -r usuario2
root@192:/home/maltamirano# cut -d: -f1 /etc/passwd
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
operator
games
ftp
nobody
dbus
tss
avahi
systemd-oom
polkitd
rtkit
pipewire
geoclue
clevis
sssd
gnome-remote-desktop
libstoragemgmt
systemd-coredump
wsdd
colord
setroubleshoot
flatpak
gdm
gnome-initial-setup
sshd
chrony
dnsmasq
tcpdump
maltamirano
smtpd
smtpq
named
rpc
rpcuser
seguridadops
root@192:/home/maltamirano#
```

En esta imagen se evidencia la eliminación de los usuarios innecesarios **usuario1** y **usuario2** mediante el comando **userdel -r**, que además borra sus carpetas personales. Posteriormente, se utilizó **cut -d: -f1 /etc/passwd** para listar todos los usuarios del sistema, confirmando que solo permanece el usuario creado (seguridadops) y las cuentas esenciales y las creadas para la administración. Con esto se completa el Requerimiento 2, asegurando que no existan cuentas redundantes que puedan representar un riesgo de seguridad.

3. Políticas básicas de seguridad y gestión de actualizaciones (4 Puntos)

- Aplica una política de caducidad de contraseña de 90 días para todos los usuarios.



```
root@192:/home/maltamirano - sudo su
root@192:/home/maltamirano# sudo chage -M 90 seguridadops
root@192:/home/maltamirano# sudo chage -l seguridadops
Último cambio de contraseña          : ago 08, 2025
La contraseña caduca                  : nov 06, 2025
Contraseña inactiva                   : nunca
La cuenta caduca                      : nunca
Número de días mínimo entre cambio de contraseña : 0
Número de días máximo entre cambio de contraseña : 90
Número de días de aviso antes de que caduque la contraseña : 7
root@192:/home/maltamirano#
```

En esta imagen se muestra la configuración de caducidad de contraseña para el usuario **seguridadops** establecida en 90 días mediante el comando **chage -M 90**. Posteriormente, con **chage -l seguridadops** se verificaron los parámetros, confirmando que el máximo de días entre cambios de contraseña es de 90 y que existe un aviso de 7 días antes de la expiración. Este ajuste cumple con la política de seguridad exigida en el Requerimiento 3, fortaleciendo el control de credenciales en el servidor.

- Deshabilita el login de root por SSH.


```
GNU nano 8.1 /etc/ssh/sshd_config
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:
#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes
#AllowUsers maltamirano

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
```

```
GNU nano 8.1 /etc/ssh/sshd_config
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:
#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes
#AllowUsers maltamirano

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
131 líneas escritas
```

Ayuda Guardar Buscar Cortar Ejecutar Ubicación M-U Deshacer M-A Poner marca M-] A llave
Salir Leer fich. Reemplazar Pegar Justificar Ir a línea M-E Rehacer M-G Copiar M-B Buscar atrás

En esta evidencia se observa la edición del archivo de configuración **/etc/ssh/sshd_config** para deshabilitar el inicio de sesión del usuario root mediante SSH. Se modificó la directiva **PermitRootLogin** de su valor por defecto a no, impidiendo que el superusuario pueda autenticarse remotamente. Este cambio fortalece la seguridad del servidor, ya que evita intentos de acceso directo a la cuenta más privilegiada, cumpliendo así uno de los puntos clave del Requerimiento 3.


```
root@192:/home/maltamirano - sudo su
root@192:/home/maltamirano# sudo nano /etc/ssh/sshd_config
root@192:/home/maltamirano# sudo systemctl restart sshd
root@192:/home/maltamirano#
```

En esta imagen se muestra la finalización de la configuración para deshabilitar el inicio de sesión remoto de root por SSH. Tras editar el archivo **/etc/ssh/sshd_config** para establecer **PermitRootLogin no**, se ejecutó el comando **systemctl restart sshd** para aplicar los cambios. Con esta acción, el servidor refuerza su seguridad al impedir conexiones directas del superusuario, cumpliendo así con uno de los apartados críticos del Requerimiento 3.

- Ejecuta una actualización completa del sistema y deja registro de ello.

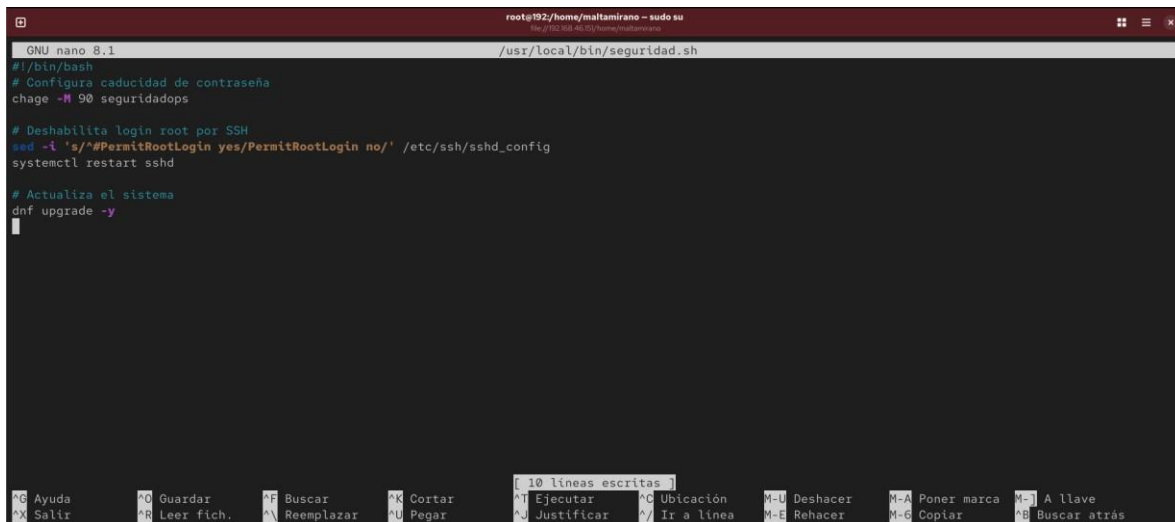
```
root@192:/home/maltamirano - sudo su
root@192:/home/maltamirano# sudo dnf check-update
Última comprobación de caducidad de metadatos hecha hace 1:10:29, el jue 07 ago 2025 20:29:18.

libxml2.x86_64                2.12.5-8.el10_0            baseos
libxslt.x86_64                1.1.39-8.el10_0            appstream
python3-libxml2.x86_64        2.12.5-8.el10_0            baseos
root@192:/home/maltamirano# sudo dnf upgrade -y
Última comprobación de caducidad de metadatos hecha hace 1:10:49, el jue 07 ago 2025 20:29:18.
Dependencias resueltas.
=====
Paquete                        Arquitectura              Versión                   Repositorio              Tam.
=====
Actualizando:
libxml2                        x86_64                    2.12.5-8.el10_0          baseos                    691 k
libxslt                        x86_64                    1.1.39-8.el10_0          appstream                 189 k
python3-libxml2                x86_64                    2.12.5-8.el10_0          baseos                    223 k
=====
Resumen de la transacción
=====
Actualizar 3 Paquetes

Tamaño total de la descarga: 1.1 M
Descargando paquetes:
(1/3): python3-libxml2-2.12.5-8.el10_0.x86_64.rpm                2.3 MB/s | 223 kB  00:00
(2/3): libxslt-1.1.39-8.el10_0.x86_64.rpm                        1.8 MB/s | 189 kB  00:00
(3/3): libxml2-2.12.5-8.el10_0.x86_64.rpm                        5.3 MB/s | 691 kB  00:00
-----
Total                                                                791 kB/s | 1.1 MB  00:01
Ejecutando verificación de operación
Verificación de operación exitosa.
```

En esta captura se evidencia la ejecución de la actualización completa del sistema en Rocky Linux. Primero se utilizó **dnf check-update** para verificar la disponibilidad de paquetes más recientes, identificando tres actualizaciones pendientes. Luego, con **dnf upgrade -y**, se descargaron e instalaron dichos paquetes, finalizando con la verificación de operación exitosa. Este procedimiento asegura que el sistema esté protegido frente a vulnerabilidades conocidas, cumpliendo con el apartado de gestión de actualizaciones del Requerimiento 3.

- El script que automatiza estas acciones y un log de la ejecución.



```
GNU nano 8.1 /usr/local/bin/seguridad.sh
#!/bin/bash
# Configura caducidad de contraseña
chage -M 90 seguridadops

# Deshabilita login root por SSH
sed -i 's/^PermitRootLogin yes/PermitRootLogin no/' /etc/ssh/sshd_config
systemctl restart sshd

# Actualiza el sistema
dnf upgrade -y
```

En esta captura se presenta el script **seguridad.sh** creado en **/usr/local/bin** para automatizar las tareas del Requerimiento 3. El script incluye tres funciones clave: establecer la caducidad de la contraseña del usuario **seguridadops** a **90 días**, deshabilitar el inicio de sesión remoto de **root por SSH** editando el archivo de configuración y reiniciando el servicio, y finalmente ejecutar la actualización completa del sistema con **dnf upgrade -y**. Este enfoque garantiza que las medidas de seguridad puedan aplicarse de forma rápida y repetible, cumpliendo con las exigencias de automatización y trazabilidad solicitadas.



```
root@192:/home/maltamirano# sudo nano /usr/local/bin/seguridad.sh
root@192:/home/maltamirano# sudo chmod +x /usr/local/bin/seguridad.sh
root@192:/home/maltamirano# sudo /usr/local/bin/seguridad.sh | tee /var/log/seguridad.log
Última comprobación de caducidad de metadatos hecha hace 1:19:05, el jue 07 ago 2025 20:29:18.
Dependencias resueltas.
Nada por hacer.
¡Listo!
root@192:/home/maltamirano#
```

En esta imagen se muestra la ejecución final del script **seguridad.sh** ubicado en **/usr/local/bin**, el cual fue previamente editado y configurado con permisos de ejecución (**chmod +x**). El script se ejecutó utilizando **tee** para registrar toda la salida en el archivo **/var/log/seguridad.log**, permitiendo mantener evidencia de las acciones realizadas. El resultado indica que no había actualizaciones pendientes y que el proceso se completó con éxito, mostrando el mensaje de confirmación “¡Listo!”. Esto valida que la automatización de las tareas de seguridad fue aplicada correctamente y quedó documentada.