

# Task 14: Linux Server Hardening & Secure Configuration

## Tools:

**Primary:** Ubuntu / Kali Linux

**Alternatives:** Lynis, CIS Benchmarks

## 1. Objective

The objective of this task is to harden a Linux server by applying secure configuration practices to reduce attack surfaces and protect against common security threats.

---

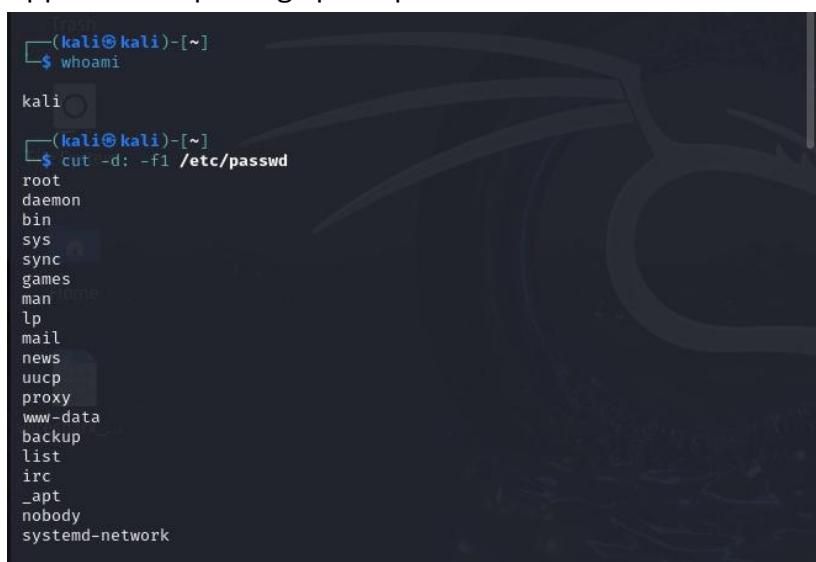
## 2. Tools Used

- Ubuntu / Kali Linux
  - Built-in Linux utilities
  - UFW Firewall
- 

## 3. Hardening Steps Performed

### 3.1 User & Access Control

- Reviewed existing user accounts
- Removed unused users
- Restricted sudo access
- Applied least privilege principle



The screenshot shows a terminal window with a dark background featuring the Kali Linux logo. The terminal prompt is '(kali㉿kali)-[~]'. The user runs the command '\$ whoami' which returns 'kali'. Then, the user runs '\$ cut -d: -f1 /etc/passwd' to list all user accounts. The output shows the following users: root, daemon, bin, sys, sync, games, man, lp, mail, news, uucp, proxy, www-data, backup, list, irc, \_apt, nobody, and systemd-network.

```
(kali㉿kali)-[~]
$ whoami
kali
(kali㉿kali)-[~]
$ cut -d: -f1 /etc/passwd
root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
www-data
backup
list
irc
_apt
nobody
systemd-network
```

```
(kali㉿kali)-[~]
$ systemctl list-units --type=service --state=running
 _UNIT           LOAD   ACTIVE SUB   DESCRIPTION
accounts-daemon.service loaded active running Accounts Service
colord.service     loaded active running Manage, Install and Generate
cron.service      loaded active running Regular background program p
dbus.service       loaded active running D-Bus System Message Bus
getty@tty1.service loaded active running Getty on tty1
haveged.service    loaded active running Entropy Daemon based on the
lightdm.service    loaded active running Light Display Manager
ModemManager.service loaded active running Modem Manager
nessusd.service    loaded active running The Nessus Vulnerability Sca
NetworkManager.service loaded active running Network Manager
open-vm-tools.service loaded active running Service for virtual machines
polkit.service     loaded active running Authorization Manager
rtkit-daemon.service loaded active running RealtimeKit Scheduling Polic
systemd-journald.service loaded active running Journal Service
systemd-logind.service loaded active running User Login Management
systemd-udevd.service loaded active running Rule-based Manager for Devic
udisks2.service    loaded active running Disk Manager
upower.service     loaded active running Daemon for power management
user@1000.service   loaded active running User Manager for UID 1000

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

```
(kali㉿kali)-[~]
$ sudo ss -tuln
[sudo] password for kali:
Netid State      Recv-Q      Send-Q      Local Address:Port      Peer Address:Port
tcp LISTEN      0          1024          0.0.0.0:8834      0.0.0.0:*
tcp LISTEN      0          1024          [::]:8834      [::]:*
```

```
(kali㉿kali)-[~]
$ getent group sudo
sudo:x:27:kali

(kali㉿kali)-[~]
$ sudo deluser username
fatal: The user `username' does not exist.

(kali㉿kali)-[~]
$ sudo deluser username
fatal: The user `username' does not exist.

(kali㉿kali)-[~]
$ sudo deluser username sudo
fatal: The user `username' does not exist.
```

### 3.2 SSH Hardening

- Disabled root login
- Disabled password-based authentication
- Enforced key-based SSH access

```
(kali㉿kali)-[~]
$ sudo nano /etc/ssh/sshd_config
```

```
GNU nano 8.2                                     /etc/ssh/sshd_config
#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes
File System
# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile      .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

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

```
└─(kali㉿kali)-[~]
$ sudo systemctl restart ssh
```

```
└─(kali㉿kali)-[~]  gophish_admin.crt   scan_report.txt
  $ ssh-keygen           gophish_admin.key  static
                        gophish.db    templates
Generating public/private ed25519 key pair. -SSH2-256  Templates
Enter file in which to save the key (/home/kali/.ssh/id_ed25519): Keygen
Enter passphrase for "Keygen" (empty for no passphrase): ERSION
Enter same passphrase again: Videos
Your identification has been saved in Keygen
Your public key has been saved in Keygen.pub
The key fingerprint is:
SHA256:WR9KMM61WyMxYLH/bzmQDuBnMrX9QNNRLArdMr1QCZs kali@kali
The key's randomart image is:
+-- [ED25519 256] --+
|       ....-+o|
|       ... =o* o|
|       =.*oEB + |
|       o.0o*=o+ |
|       S++=+-.. |
|       . =+o.+.. |
|       . . +o|
|       .. |
+--- [SHA256] ---+
```

### 3.3 System Updates

- Updated all packages
- Enabled automatic security updates

```

File Actions Edit View Help
File Actions Edit View Help
(kali㉿kali)-[~]
$ sudo apt update && sudo apt upgrade -y
Get:1 http://kali.download/kali kali-rolling/main amd64 Packages [34.0 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [20.6 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Packages (deb) [52.0 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [118 kB]
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb) [271 kB]
Get:6 http://kali.download/kali kali-rolling/non-free amd64 Packages [188 kB]
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [890 kB]
Fetched 74.2 MB in 23s (3,257 kB/s)
2234 packages can be upgraded. Run 'apt list --upgradable' to see them.
The following packages were automatically installed and are no longer required:
amass-common libgeoip3.13.0 libsqlcipher1
bloodhound.py libgl1-mesa-dev libswscale8
firebird3.0-common libgles-dev libtaglibv5
firebird3.0-common-doc libglvnd-core-dev libtaglibv5-vanilla
firmware-ti-connectivity libglvnd-dev libtagc0
icu-devtools libgtksourceview-3.0-1 libunwind-19
libibus1.20230802 libgtksourceviewmm-3.0-0v5 libutempter0
libaudio2 libgumbo2 libwireshark18
libavfilter10 libicu-dev libwritap15
libavformat61 libinsipatch-1.0-2 libwsutil16
libff10 libmbdecrypto7t64 mesa-vdpau-drivers
libbson-1.0-0t64 libmongoc-1.0-0t64 openjdk-23-jre
libc++11-19 libnet1 openjdk-23-jre-headless
libc++abi11-19 libpaper1 pocketsphinx-en-us
libcapstone4 libplacebo349 python3-appdirs
libconfig++9v5 libpocketsshinx3 python3-bluepy
libconfig-inifiles-perl libpostproc58 python3-click-plugins
libconfig9 python3-pysmi
python3-kismetcapturetladsl
python3-kismetcapturetlamr
python3-nfsclient
python3-ntlm-auth
python3-packaging-whl
python3-poetry-dynamic-versioning
python3-protobuf
python3-requests-ntlm
python3-tomlkit
python3-wheel-whl
python3-yaswfp
python3-zombie-imp
ruby-utf-ext
ruby-zeitwerk
ruby3.1
ruby3.1-dev
ruby3.1-doc

```

```

File Actions Edit View Help
File Actions Edit View Help
(kali㉿kali)-[~]
$ sudo apt install unattended-upgrades
sudo dpkg-reconfigure unattended-upgrades
The following packages were automatically installed and are no longer required:
openjdk-23-jre openjdk-23-jre-headless
Use 'sudo apt autoremove' to remove them.
Installing:
unattended-upgrades python3-unattended-upgrades
Installing dependencies: unattended-upgrades
python3-distro-info
python3-distro-info
Suggested packages:
bsd-mailx default-mta | mail-transport-agent needrestart
Summary:
Upgrading: 0, Installing: 2, Removing: 0, Not Upgrading: 2234
Download size: 74.9 kB
Space needed: 368 kB / 51.1 GB available
Continue? [Y/n] y
Get:1 http://kali.download/kali kali-rolling/main amd64 python3-distro-info all 1.14 [7,848 B]
Get:2 http://http.kali.org/kali kali-rolling/main amd64 unattended-upgrades all 2.12+nmu1 [67.1 kB]
Fetched 74.9 kB in 1s (64.3 kB/s)
Preconfiguring packages ...
Selecting previously unselected package python3-distro-info.
(Reading database ... 401130 files and directories currently installed.)
Preparing to unpack .../python3-distro-info_1.14_all.deb ...
Unpacking python3-distro-info (1.14) ...
Selecting previously unselected package unattended-upgrades.

```

### 3.4 Firewall Configuration

- Enabled UFW firewall
- Allowed only essential ports (SSH, HTTP, HTTPS)
- Blocked all other traffic by default

```

(kali㉿kali)-[~]
$ sudo ufw enable
Firewall is active and enabled on system startup
correct.log gophish_admin.crt
(kali㉿kali)-[~] gophish_admin.key
$ sudo ufw allow ssh gophish_db
sudo ufw allow 80 gophish-v0.12.3-Linux-64bit.zip
sudo ufw allow 443 jaunu.log
jaunu.log LICENSE
Rule added
Rule added (v6) Music
Skipping adding existing rule
Skipping adding existing rule (v6)
Skipping adding existing rule
Skipping adding existing rule (v6)

```

```
(kali㉿kali)-[~] $ sudo ufw status
Status: active
To                         Action      From
--                         Proto     --
23                         DENY       Anywhere
22                         ALLOW      Anywhere
80                         ALLOW      Anywhere
443                        ALLOW      Anywhere
21                         DENY       Anywhere
Anywhere                   DENY       192.168.1.100
22/tcp                     ALLOW      Anywhere
23 (v6)                   DENY       Anywhere (v6)
22 (v6)                   ALLOW      Anywhere (v6)
80 (v6)                   ALLOW      Anywhere (v6)
21 (v6)                   DENY       Anywhere (v6)
443 (v6)                  ALLOW      Anywhere (v6)
22/tcp (v6)                ALLOW      Anywhere (v6)
```

### 3.5 Service Management

- Identified unnecessary running services
- Stopped and disabled unused services

```
(kali㉿kali)-[~] $ systemctl list-unit-files --type=service
UNIT FILE                                     STATE   PRESET
accounts-daemon.service                      enabled  enabled
apache-htcacheclean.service                 disabled  disabled
apache-htcacheclean@.service                disabled  static
apache2.service                            disabled  temporary
apache2@.service                          disabled  temporary
apparmor.service                           disabled  update-disabled
apt-daily-upgrade.service                 static   version-
apt-daily.service                           static   video-
atftpd.service                           indirect  disabled
auth-rpcgss-module.service                static   -
autovt@.service                           alias    -
avahi-daemon.service                      disabled  disabled
blueman-mechanism.service                disabled  disabled
bluetooth.service                         disabled  disabled
capsule@.service                           static   -
colord.service                           static   -
configure-printer@.service               static   -
console-getty.service                    disabled  disabled
console-setup.service                    enabled   enabled
container-getty@.service                 static   -
cron.service                             enabled   enabled
cryptdisks-early.service                 masked   disabled
cryptdisks.service                       masked   disabled
dbus-org.freedesktop.hostname1.service  alias    -
dbus-org.freedesktop.locale1.service    alias    -
dbus-org.freedesktop.login1.service     alias    -
dbus-org.freedesktop.ModemManager1.service alias    -
```

```
(kali㉿kali)-[~] $ sudo systemctl stop apache2
(kali㉿kali)-[~] $ sudo systemctl disable apache2
Synchronizing state of apache2.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install disable apache2
```

### 3.6 File Permission Hardening

- Secured sensitive system files
- Restricted SSH directory permissions

```
[firewall.rules, EXC] - [public]
└──(kali㉿kali)-[~]
    $ sudo chmod 644 /etc/passwd
    $ ls

└──(kali㉿kali)-[~]
    $ sudo chmod 600 /etc/shadow

└──(kali㉿kali)-[~]
    $ chmod 700 ~/.ssh
    chmod 600 ~/.ssh/authorized_keys

chmod: cannot access '/home/kali/.ssh': No such file or directory
chmod: cannot access '/home/kali/.ssh/authorized_keys': No such file or directory
```

### 3.7 Log Monitoring

- Reviewed authentication and system logs
- Checked failed login attempts

```
└──(kali㉿kali)-[~]  [root@kali ~] - [Time: 00:00:00]
    $ sudo journalctl -u sshd.log
    update_burp.sh
    LICENSE
    VERSION
Feb 10 13:06:17 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server ...
Feb 10 13:06:17 kali sshd[7993]: Server listening on 0.0.0.0 port 22.
Feb 10 13:06:17 kali sshd[7993]: Server listening on :: port 22.
Feb 10 13:06:17 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Feb 10 13:08:21 kali sshd[7993]: Received signal 15; terminating.
Feb 10 13:08:21 kali systemd[1]: Stopping ssh.service - OpenBSD Secure Shell server ...
Feb 10 13:08:21 kali systemd[1]: ssh.service: Deactivated successfully.
Feb 10 13:08:21 kali systemd[1]: Stopped ssh.service - OpenBSD Secure Shell server.
Feb 10 13:08:21 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server ...
Feb 10 13:08:21 kali sshd[9010]: Server listening on 0.0.0.0 port 22.
Feb 10 13:08:21 kali sshd[9010]: Server listening on :: port 22.
Feb 10 13:08:21 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.

└──(kali㉿kali)-[~]
    $ sudo lastb
    sudo: lastb: command not found
```

## 4. Security Benefits

- Reduced risk of brute-force attacks
- Limited unauthorized access
- Improved system integrity
- Lowered attack surface

## 5. Conclusion

Linux server hardening is essential to protect systems from real-world attacks. This task provided hands-on experience in securing Linux servers using industry best practices.