# ELEVATE LABS
# TASK 16
# Incident Response & Security Breach Simulation

**Student Name:** Chalumuri Sri Venkata Srinivas

**University:** Aditya University

**Domain:** Cybersecurity

**Duration:** 1st January 2026 to 30th April 2026

## 1. Simulate a basic security incident such as repeated failed logins or unauthorized access.

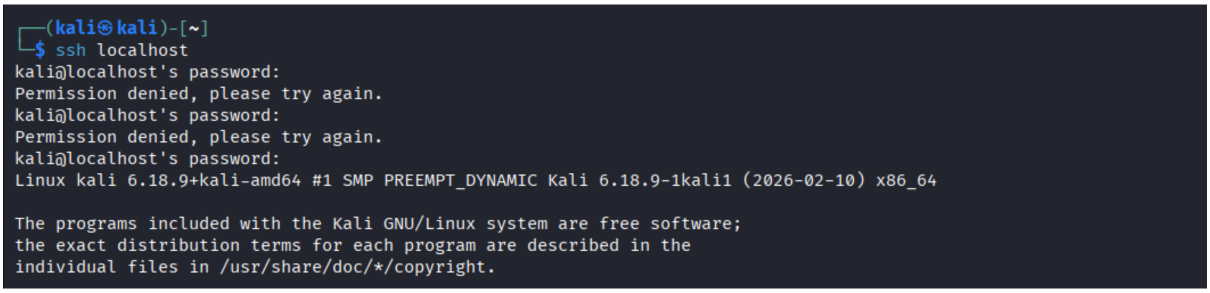
A basic security incident such as repeated failed login attempts usually indicates a brute-force or unauthorized access attack, where an attacker tries multiple passwords to gain entry into a system. In Linux systems, every authentication attempt is recorded in system log files, allowing administrators to detect suspicious behavior. By simulating failed logins in a controlled environment, security analysts can observe how attacks appear in authentication logs, understand attacker patterns, and practice incident detection. This process helps improve monitoring, strengthens access controls, and prepares administrators to respond effectively to real-world threats.

**Step 2: Generate Failed Login Attempts**

Run:

```
ssh localhost
```

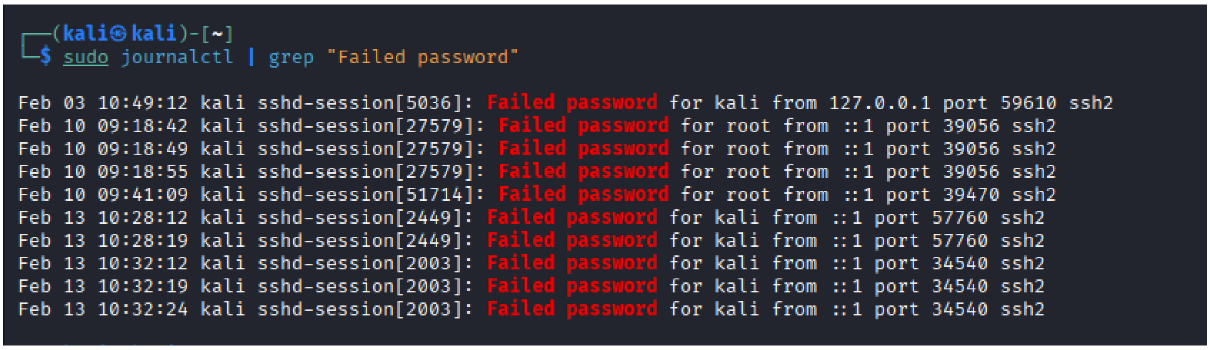
Enter a **wrong password** 3–5 times.



```
┌──(kali㉿kali)-[~]
└─$ ssh localhost
kali@localhost's password:
Permission denied, please try again.
kali@localhost's password:
Permission denied, please try again.
kali@localhost's password:
Linux kali 6.18.9+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.18.9-1kali1 (2026-02-10) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
```

## Step 3: View Authentication Logs

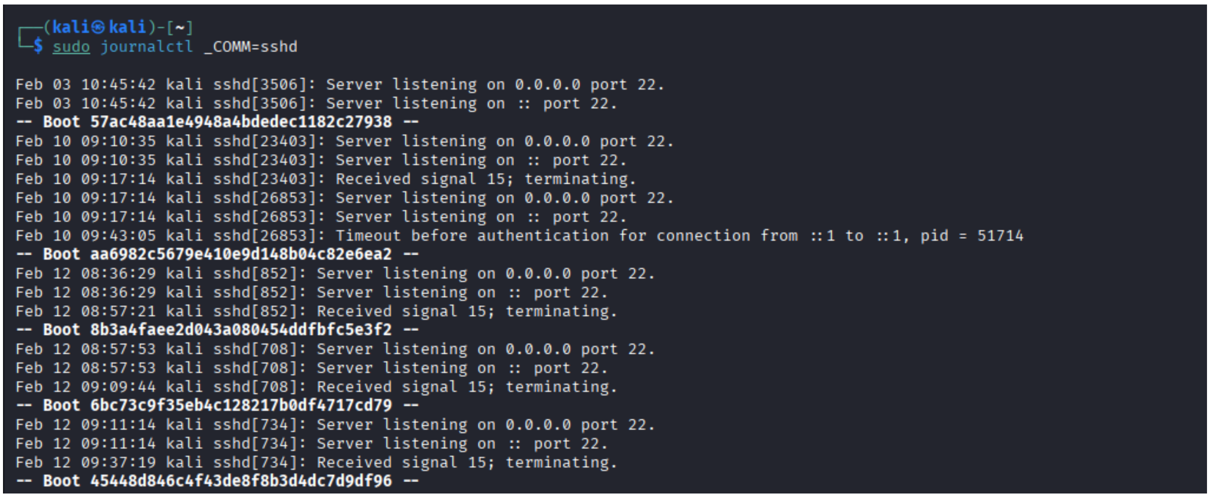
sudo journalctl | grep "Failed password"



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed password"

Feb 03 10:49:12 kali sshd-session[5036]: Failed password for kali from 127.0.0.1 port 59610 ssh2
Feb 10 09:18:42 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:49 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:55 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:41:09 kali sshd-session[51714]: Failed password for root from ::1 port 39470 ssh2
Feb 13 10:28:12 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:28:19 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:32:12 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:19 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:24 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
```

## View SSH Authentication Logs

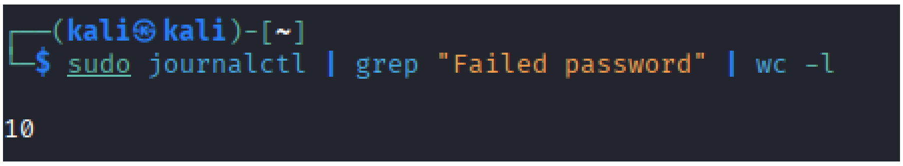
sudo journalctl _COMM=sshd



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl _COMM=sshd

Feb 03 10:45:42 kali sshd[3506]: Server listening on 0.0.0.0 port 22.
Feb 03 10:45:42 kali sshd[3506]: Server listening on :: port 22.
-- Boot 57ac48aa1e4948a4bdedec1182c27938 --
Feb 10 09:10:35 kali sshd[23403]: Server listening on 0.0.0.0 port 22.
Feb 10 09:10:35 kali sshd[23403]: Server listening on :: port 22.
Feb 10 09:17:14 kali sshd[23403]: Received signal 15; terminating.
Feb 10 09:17:14 kali sshd[26853]: Server listening on 0.0.0.0 port 22.
Feb 10 09:17:14 kali sshd[26853]: Server listening on :: port 22.
Feb 10 09:43:05 kali sshd[26853]: Timeout before authentication for connection from ::1 to ::1, pid = 51714
-- Boot aa6982c5679e410e9d148b04c82e6ea2 --
Feb 12 08:36:29 kali sshd[852]: Server listening on 0.0.0.0 port 22.
Feb 12 08:36:29 kali sshd[852]: Server listening on :: port 22.
Feb 12 08:57:21 kali sshd[852]: Received signal 15; terminating.
-- Boot 8b3a4faee2d043a080454ddfbfc5e3f2 --
Feb 12 08:57:53 kali sshd[708]: Server listening on 0.0.0.0 port 22.
Feb 12 08:57:53 kali sshd[708]: Server listening on :: port 22.
Feb 12 09:09:44 kali sshd[708]: Received signal 15; terminating.
-- Boot 6bc73c9f35eb4c128217b0df4717cd79 --
Feb 12 09:11:14 kali sshd[734]: Server listening on 0.0.0.0 port 22.
Feb 12 09:11:14 kali sshd[734]: Server listening on :: port 22.
Feb 12 09:37:19 kali sshd[734]: Received signal 15; terminating.
-- Boot 45448d846c4f43de8f8b3d4dc7d9df96 --
```

## Count Failed Attempts

sudo journalctl | grep "Failed password" | wc -l



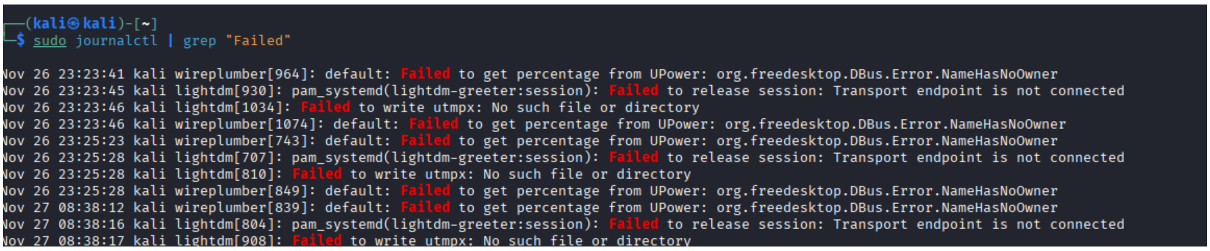
```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed password" | wc -l

10
```

## 2. Identify suspicious activity by analyzing system and authentication logs

Identifying suspicious activity involves analyzing system and authentication logs to detect abnormal behavior such as repeated failed login attempts, unauthorized user access, or logins from unknown sources. Linux systems maintain detailed records of authentication events through system logging services, allowing administrators to trace login failures, successful authentications, and privilege escalation attempts. By examining these logs, security analysts can recognize attack patterns like brute-force attempts or compromised accounts. Log analysis plays a vital role in incident response because it provides forensic evidence, supports early threat detection, and helps determine appropriate containment measures.

### View Authentication Failures

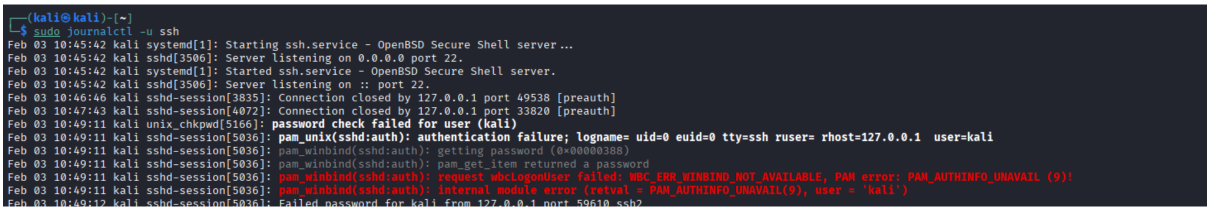
sudo journalctl | grep "Failed"



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed"

Nov 26 23:23:41 kali wireplumber[964]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:23:45 kali lightdm[930]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
Nov 26 23:23:46 kali lightdm[1034]: Failed to write utmpx: No such file or directory
Nov 26 23:23:46 kali wireplumber[1074]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:25:23 kali wireplumber[743]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:25:28 kali lightdm[707]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
Nov 26 23:25:28 kali lightdm[810]: Failed to write utmpx: No such file or directory
Nov 26 23:25:28 kali wireplumber[849]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 27 08:38:12 kali wireplumber[839]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 27 08:38:16 kali lightdm[804]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
Nov 27 08:38:17 kali lightdm[908]: Failed to write utmpx: No such file or directory
```

### View SSH Login Activity

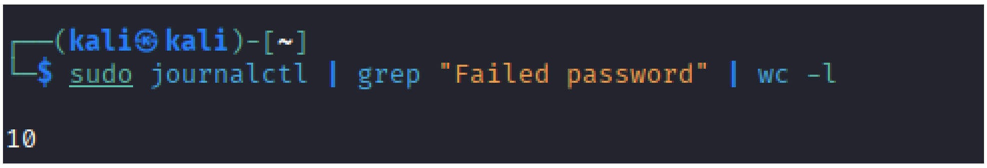
sudo journalctl -u ssh



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl -u ssh
Feb 03 10:45:42 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Feb 03 10:45:42 kali sshd[3506]: Server listening on 0.0.0.0 port 22.
Feb 03 10:45:42 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Feb 03 10:45:42 kali sshd[3506]: Server listening on :: port 22.
Feb 03 10:46:46 kali sshd-session[3835]: Connection closed by 127.0.0.1 port 49538 [preauth]
Feb 03 10:47:43 kali sshd-session[4072]: Connection closed by 127.0.0.1 port 33820 [preauth]
Feb 03 10:49:11 kali unix_chkpwd[5166]: password check failed for user (kali)
Feb 03 10:49:11 kali sshd-session[5036]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=127.0.0.1  user=kali
Feb 03 10:49:11 kali sshd-session[5036]: pam_winbind(sshd:auth): getting password (0x00000388)
Feb 03 10:49:11 kali sshd-session[5036]: pam_winbind(sshd:auth): pam_get_item returned a password
Feb 03 10:49:11 kali sshd-session[5036]: pam_winbind(sshd:auth): request wbcLogonUser failed: WBC_ERR_WINBIND_NOT_AVAILABLE, PAM error: PAM_AUTHINFO_UNAVAIL (9)!
Feb 03 10:49:11 kali sshd-session[5036]: pam_winbind(sshd:auth): internal module error (retval = PAM_AUTHINFO_UNAVAIL(9), user = 'kali')
Feb 03 10:49:12 kali sshd-session[5036]: Failed password for kali from 127.0.0.1 port 59610 ssh2
```

### Count Failed Login Attempts

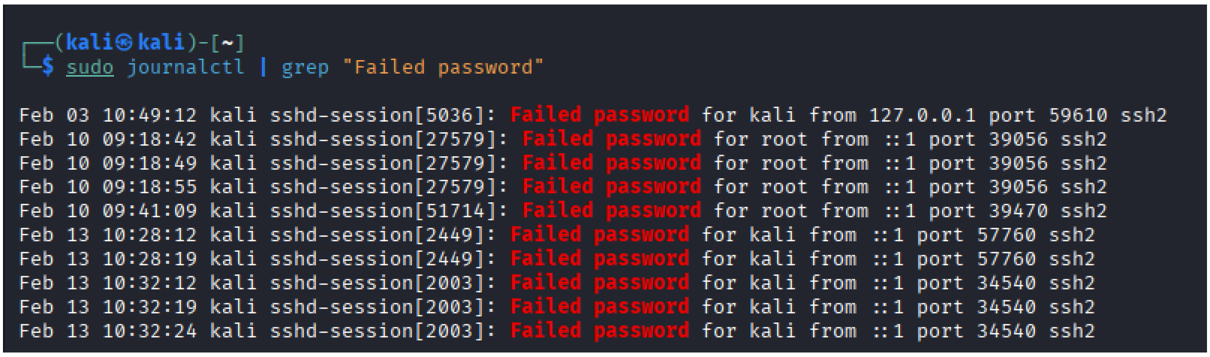
sudo journalctl | grep "Failed password" | wc -l



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed password" | wc -l

10
```

### Step 5: Identify Source IP Addresses

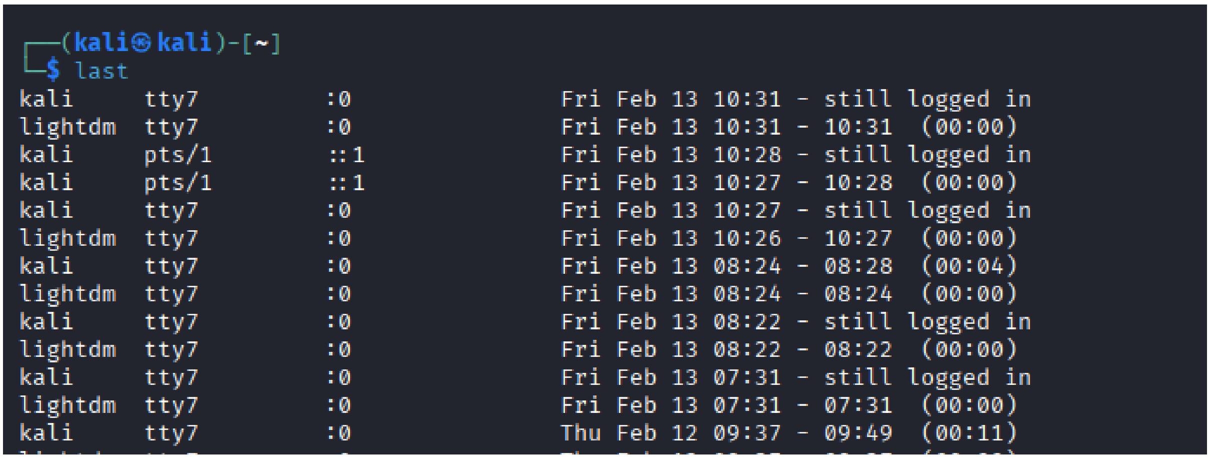
sudo journalctl | grep "Failed password"



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed password"

Feb 03 10:49:12 kali sshd-session[5036]: Failed password for kali from 127.0.0.1 port 59610 ssh2
Feb 10 09:18:42 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:49 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:55 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:41:09 kali sshd-session[51714]: Failed password for root from ::1 port 39470 ssh2
Feb 13 10:28:12 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:28:19 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:32:12 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:19 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:24 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
```

### Check Successful Logins (Compare)

last



```
┌──(kali㉿kali)-[~]
└─$ last
kali     tty7         :0               Fri Feb 13 10:31 - still logged in
lightdm  tty7         :0               Fri Feb 13 10:31 - 10:31  (00:00)
kali     pts/1        ::1              Fri Feb 13 10:28 - still logged in
kali     pts/1        ::1              Fri Feb 13 10:27 - 10:28  (00:00)
kali     tty7         :0               Fri Feb 13 10:27 - still logged in
lightdm  tty7         :0               Fri Feb 13 10:26 - 10:27  (00:00)
kali     tty7         :0               Fri Feb 13 08:24 - 08:28  (00:04)
lightdm  tty7         :0               Fri Feb 13 08:24 - 08:24  (00:00)
kali     tty7         :0               Fri Feb 13 08:22 - still logged in
lightdm  tty7         :0               Fri Feb 13 08:22 - 08:22  (00:00)
kali     tty7         :0               Fri Feb 13 07:31 - still logged in
lightdm  tty7         :0               Fri Feb 13 07:31 - 07:31  (00:00)
kali     tty7         :0               Thu Feb 12 09:37 - 09:49  (00:11)
```

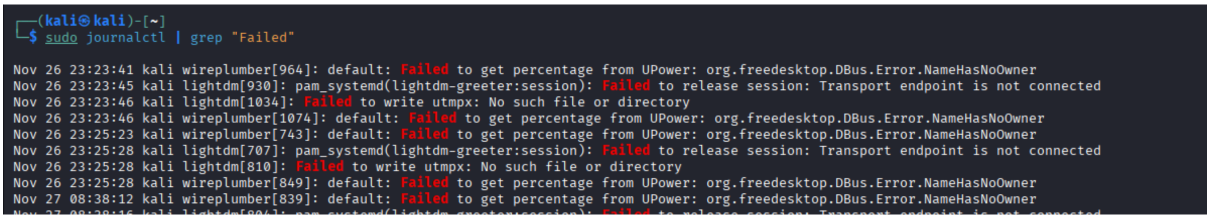
## 3. Classify the incident based on attack type and severity

After identifying suspicious authentication activity, the incident must be classified according to attack type and severity. In Linux environments, repeated failed login attempts usually indicate a brute-force or password-guessing attack, while unexpected successful logins may suggest unauthorized access or account compromise. Severity is determined by evaluating the impact and likelihood of damage. For example, multiple failed logins without success are considered medium severity, whereas a successful login following repeated failures is high severity because it implies system compromise. Proper incident classification helps security teams prioritize response actions, allocate resources effectively, and reduce further risk.

## View Failed Authentication Attempts

On modern Linux (Kali/Ubuntu):

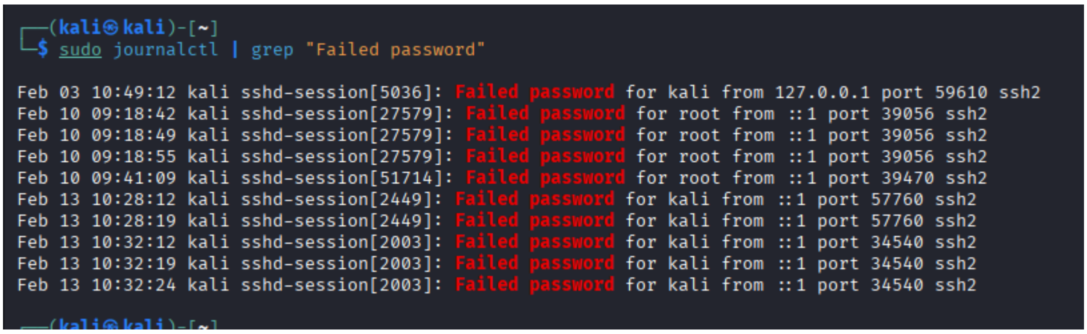
sudo journalctl | grep "Failed"



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed"

Nov 26 23:23:41 kali wireplumber[964]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:23:45 kali lightdm[930]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
Nov 26 23:23:46 kali lightdm[1034]: Failed to write utmpx: No such file or directory
Nov 26 23:23:46 kali wireplumber[1074]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:25:23 kali wireplumber[743]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:25:28 kali lightdm[707]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
Nov 26 23:25:28 kali lightdm[810]: Failed to write utmpx: No such file or directory
Nov 26 23:25:28 kali wireplumber[849]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 27 08:38:12 kali wireplumber[839]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 27 08:38:16 kali lightdm[804]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
```

## Identify Source and Username

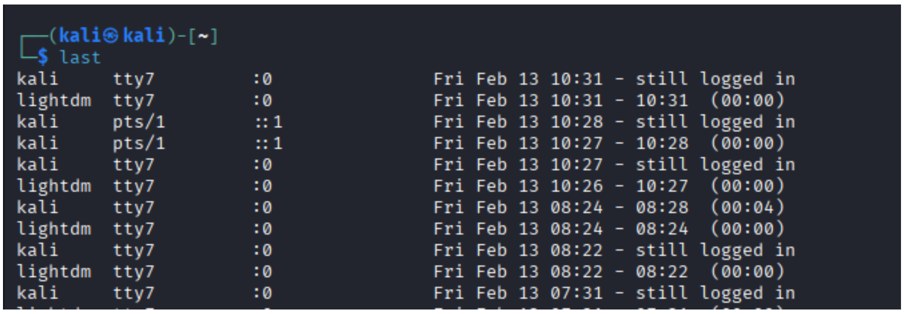
sudo journalctl | grep "Failed password"



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed password"

Feb 03 10:49:12 kali sshd-session[5036]: Failed password for kali from 127.0.0.1 port 59610 ssh2
Feb 10 09:18:42 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:49 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:55 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:41:09 kali sshd-session[51714]: Failed password for root from ::1 port 39470 ssh2
Feb 13 10:28:12 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:28:19 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:32:12 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:19 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:24 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2

┌──(kali㉿kali)-[~]
```

## Check Successful Logins

last

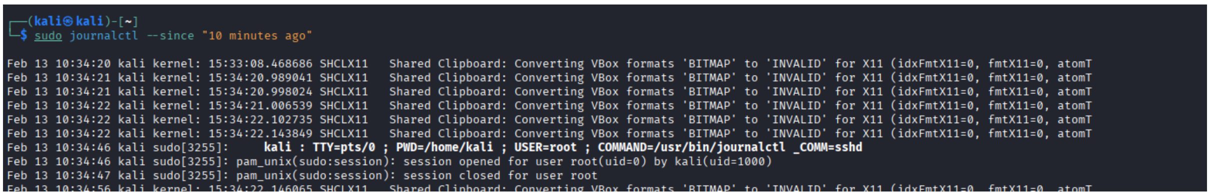


```
┌──(kali㉿kali)-[~]
└─$ last
kali     tty7         :0               Fri Feb 13 10:31 - still logged in
lightdm  tty7         :0               Fri Feb 13 10:31 - 10:31  (00:00)
kali     pts/1        ::1              Fri Feb 13 10:28 - still logged in
kali     pts/1        ::1              Fri Feb 13 10:27 - 10:28  (00:00)
kali     tty7         :0               Fri Feb 13 10:27 - still logged in
lightdm  tty7         :0               Fri Feb 13 10:26 - 10:27  (00:00)
kali     tty7         :0               Fri Feb 13 08:24 - 08:28  (00:04)
lightdm  tty7         :0               Fri Feb 13 08:24 - 08:24  (00:00)
kali     tty7         :0               Fri Feb 13 08:22 - still logged in
lightdm  tty7         :0               Fri Feb 13 08:22 - 08:22  (00:00)
kali     tty7         :0               Fri Feb 13 07:31 - still logged in
```

## Correlate Events (Attack Chain)

Check timeline:

sudo journalctl --since "10 minutes ago"



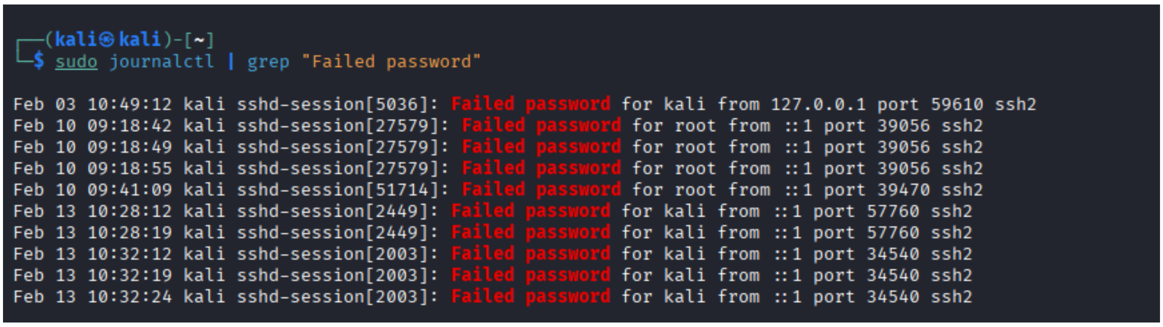
```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl --since "10 minutes ago"

Feb 13 10:34:20 kali kernel: 15:33:08.468686 SHCLX11   Shared Clipboard: Converting VBox formats 'BITMAP' to 'INVALID' for X11 (idxFmtX11=0, fmtX11=0, atomT
Feb 13 10:34:21 kali kernel: 15:34:20.989041 SHCLX11   Shared Clipboard: Converting VBox formats 'BITMAP' to 'INVALID' for X11 (idxFmtX11=0, fmtX11=0, atomT
Feb 13 10:34:21 kali kernel: 15:34:20.998024 SHCLX11   Shared Clipboard: Converting VBox formats 'BITMAP' to 'INVALID' for X11 (idxFmtX11=0, fmtX11=0, atomT
Feb 13 10:34:22 kali kernel: 15:34:21.006539 SHCLX11   Shared Clipboard: Converting VBox formats 'BITMAP' to 'INVALID' for X11 (idxFmtX11=0, fmtX11=0, atomT
Feb 13 10:34:22 kali kernel: 15:34:22.102735 SHCLX11   Shared Clipboard: Converting VBox formats 'BITMAP' to 'INVALID' for X11 (idxFmtX11=0, fmtX11=0, atomT
Feb 13 10:34:22 kali kernel: 15:34:22.143849 SHCLX11   Shared Clipboard: Converting VBox formats 'BITMAP' to 'INVALID' for X11 (idxFmtX11=0, fmtX11=0, atomT
Feb 13 10:34:46 kali sudo[3255]:     kali : TTY=pts/0 ; PWD=/home/kali ; USER=root ; COMMAND=/usr/bin/journalctl _COMM=sshd
Feb 13 10:34:46 kali sudo[3255]: pam_unix(sudo:session): session opened for user root(uid=0) by kali(uid=1000)
Feb 13 10:34:47 kali sudo[3255]: pam_unix(sudo:session): session closed for user root
Feb 13 10:34:56 kali kernel: 15:34:22.146065 SHCLX11   Shared Clipboard: Converting VBox formats 'BITMAP' to 'INVALID' for X11 (idxFmtX11=0, fmtX11=0, atomT
```

## 4. Contain the incident by isolating affected systems or accounts

Containment is a critical phase of incident response that focuses on stopping an active attack and preventing further damage. After identifying suspicious login activity such as brute-force attempts or unauthorized access, administrators must immediately isolate affected accounts or systems. In Linux environments, this typically involves locking compromised user accounts, blocking malicious IP addresses, and disconnecting systems from the network if required. Quick containment limits attacker movement, protects sensitive data, and provides time for investigation and recovery. Effective containment reduces impact and ensures the threat does not spread to other systems.

### Identify the Attacked User or IP (from Logs)

sudo journalctl | grep "Failed password"

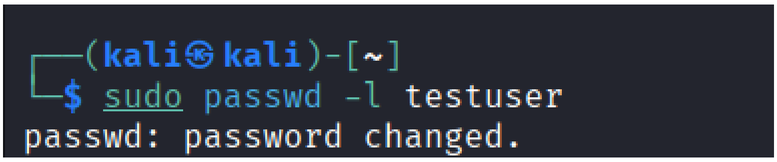


```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed password"

Feb 03 10:49:12 kali sshd-session[5036]: Failed password for kali from 127.0.0.1 port 59610 ssh2
Feb 10 09:18:42 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:49 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:55 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:41:09 kali sshd-session[51714]: Failed password for root from ::1 port 39470 ssh2
Feb 13 10:28:12 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:28:19 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:32:12 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:19 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:24 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
```

### Lock the Affected User Account (Immediate Containment)

Replace username with the real user:

sudo passwd -l testuser



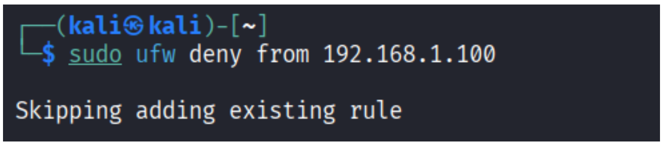
```
┌──(kali㉿kali)-[~]
└─$ sudo passwd -l testuser
passwd: password changed.
```

### Step 3: Block the Attacker IP Address

If you saw an IP (example: 192.168.1.100):

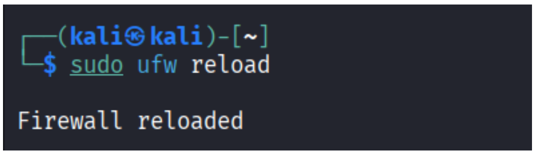
**Using UFW (if enabled):**

sudo ufw deny from 192.168.1.100



```
┌──(kali㉿kali)-[~]
└─$ sudo ufw deny from 192.168.1.100

Skipping adding existing rule
```

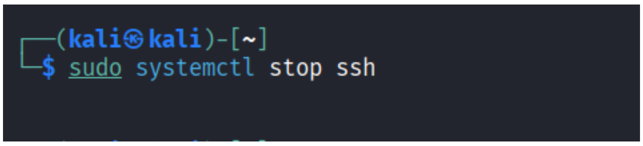
Reload firewall:
sudo ufw reload



```
┌──(kali㉿kali)-[~]
└─$ sudo ufw reload

Firewall reloaded
```

### Step 4 (Optional): Stop SSH Temporarily

If attack is ongoing:
sudo systemctl stop ssh



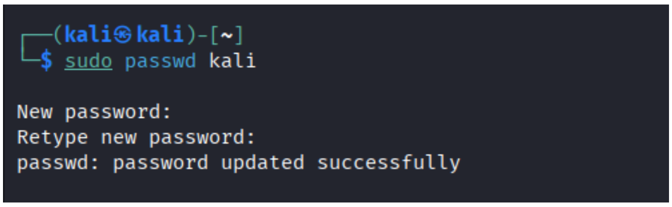
```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl stop ssh
```

## 5. Remove the threat and fix the root cause

After containment, the next phase of incident response is eradication and root-cause remediation. This step focuses on removing the attacker’s access and correcting weaknesses that allowed the incident to occur. In Linux systems, this typically includes resetting compromised passwords, removing unauthorized users, blocking malicious IP addresses, patching the system, and strengthening authentication controls. Addressing the root cause—such as weak passwords, exposed SSH services, or missing security updates—prevents the same attack from recurring. Proper threat removal ensures system integrity is restored and reduces the risk of future compromise.

### Step 1: Reset Password of Affected Account

Replace kali with your real username if different:
sudo passwd kali

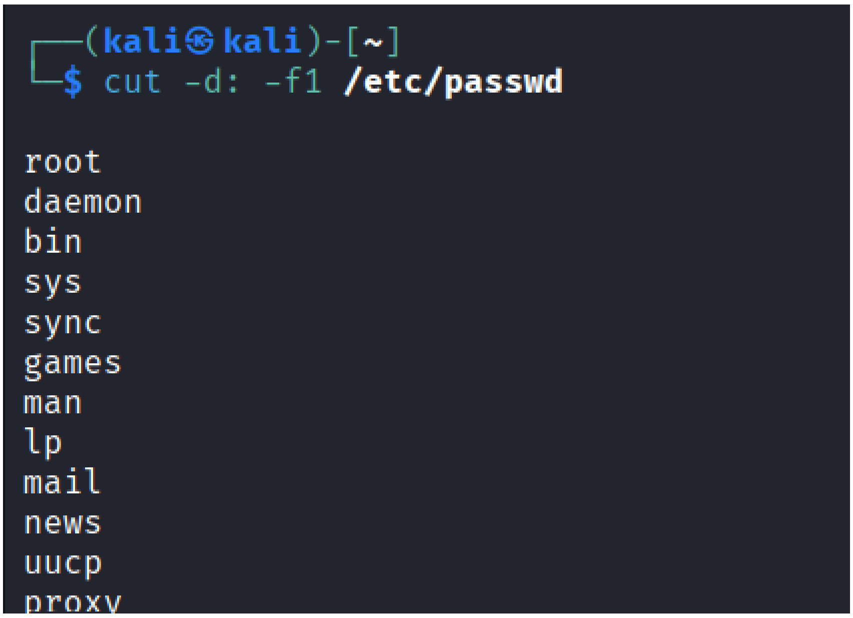


```
┌──(kali㉿kali)-[~]
└─$ sudo passwd kali

New password:
Retype new password:
passwd: password updated successfully
```

### Step 2: Check for Unknown Users (Remove If Found)

List users:

cut -d: -f1 /etc/passwd



```
┌──(kali㉿kali)-[~]
└─$ cut -d: -f1 /etc/passwd

root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
```

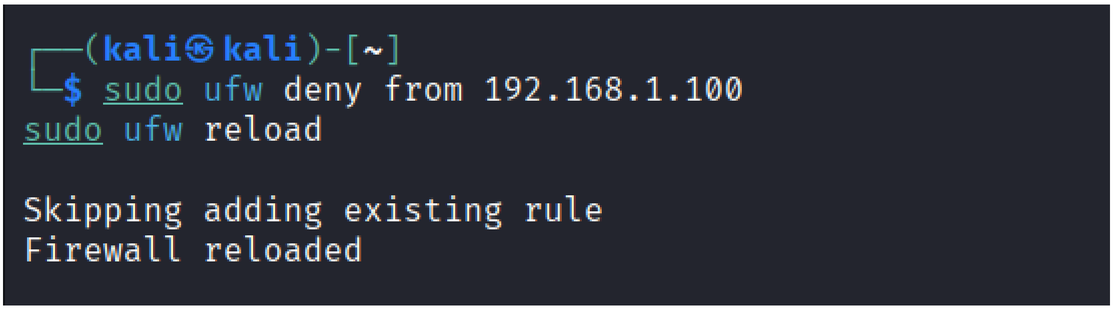
### Step 3: Remove / Block Attacker IP (If Identified)

If logs showed an IP (example 192.168.1.100):

**Using UFW:**

sudo ufw deny from 192.168.1.100

sudo ufw reload

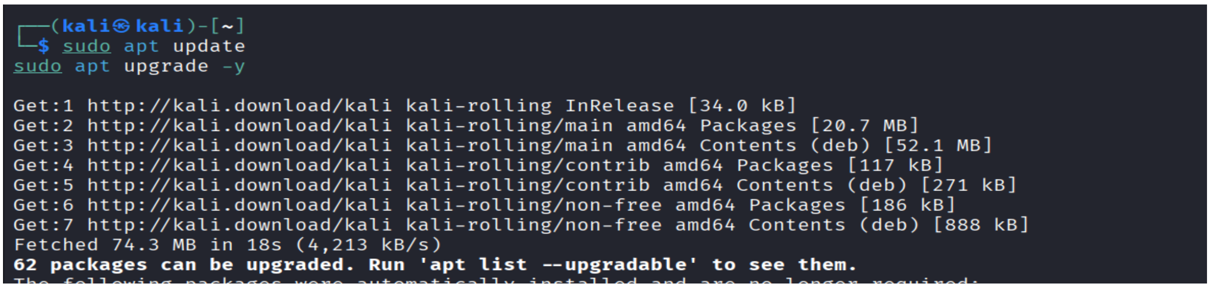


```
┌──(kali㉿kali)-[~]
└─$ sudo ufw deny from 192.168.1.100
sudo ufw reload

Skipping adding existing rule
Firewall reloaded
```

### Step 4: Update System (Fix Vulnerabilities)

sudo apt update

sudo apt upgrade -y

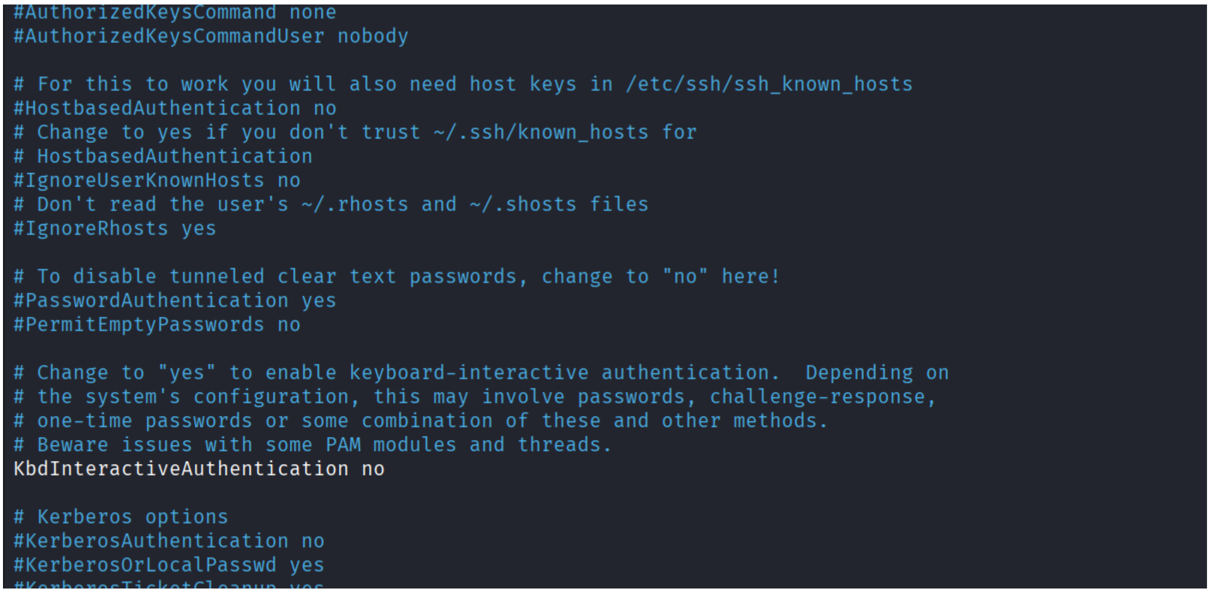


```
┌──(kali㉿kali)-[~]
└─$ sudo apt update
sudo apt upgrade -y

Get:1 http://kali.download/kali kali-rolling InRelease [34.0 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [20.7 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [52.1 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [117 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 [186 kB]
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [888 kB]
Fetched 74.3 MB in 18s (4,213 kB/s)
62 packages can be upgraded. Run 'apt list --upgradable' to see them.
The following packages were automatically installed and are no longer required:
```

## Step 5: Harden SSH (Root Cause Fix)

Edit SSH config:

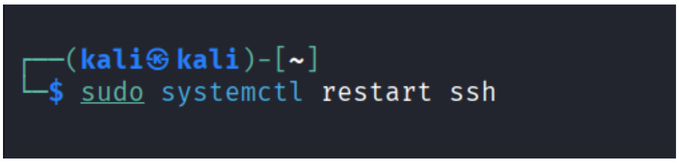
sudo nano /etc/ssh/sshd_config



```
#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 yes
#PermitEmptyPasswords no

# Change to "yes" to enable keyboard-interactive authentication.  Depending on
# the system's configuration, this may involve passwords, challenge-response,
# one-time passwords or some combination of these and other methods.
# Beware issues with some PAM modules and threads.
KbdInteractiveAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
```

## Restart SSH:

sudo systemctl restart ssh



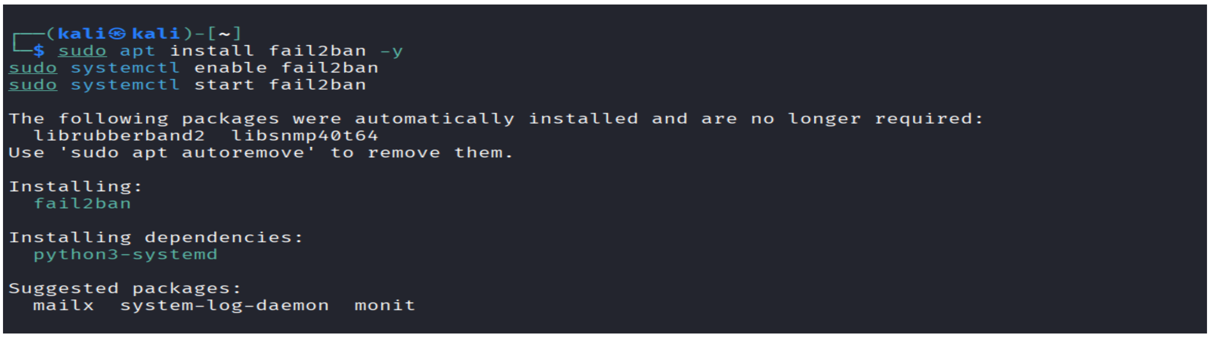
```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl restart ssh
```

## Step 6 (Recommended): Install Fail2Ban (Prevent Future Brute Force)

sudo apt install fail2ban -y

sudo systemctl enable fail2ban

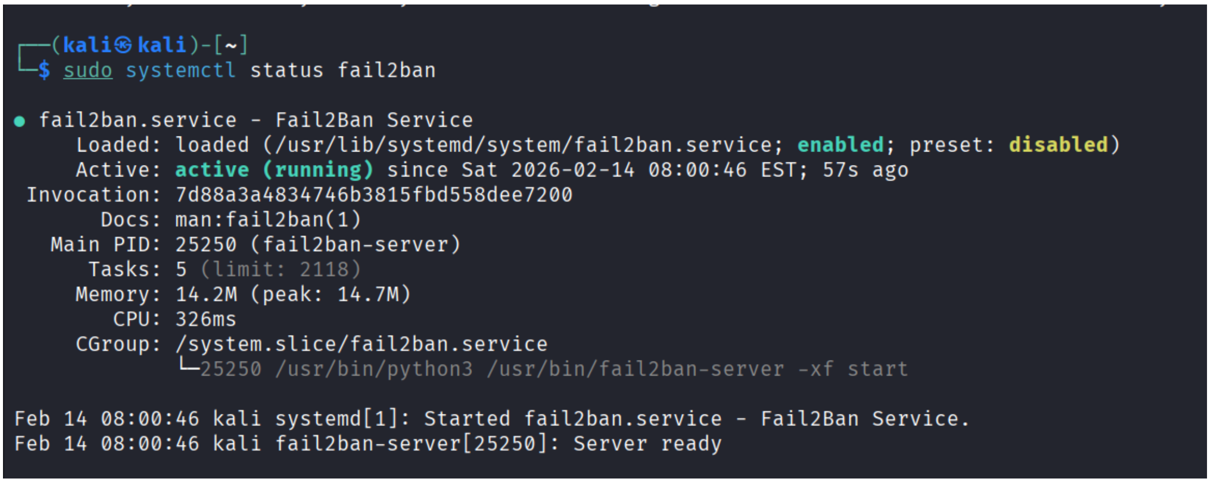
sudo systemctl start fail2ban



```
┌──(kali㉿kali)-[~]
└─$ sudo apt install fail2ban -y
sudo systemctl enable fail2ban
sudo systemctl start fail2ban

The following packages were automatically installed and are no longer required:
  librubberband2  libsnmp40t64
Use 'sudo apt autoremove' to remove them.

Installing:
  fail2ban

Installing dependencies:
  python3-systemd

Suggested packages:
  mailx  system-log-daemon  monit
```

## Check status:

sudo systemctl status fail2ban



```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl status fail2ban

● fail2ban.service - Fail2Ban Service
     Loaded: loaded (/usr/lib/systemd/system/fail2ban.service; enabled; preset: disabled)
     Active: active (running) since Sat 2026-02-14 08:00:46 EST; 57s ago
 Invocation: 7d88a3a4834746b3815fbd558dee7200
       Docs: man:fail2ban(1)
   Main PID: 25250 (fail2ban-server)
      Tasks: 5 (limit: 2118)
     Memory: 14.2M (peak: 14.7M)
        CPU: 326ms
     CGroup: /system.slice/fail2ban.service
             └─25250 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

Feb 14 08:00:46 kali systemd[1]: Started fail2ban.service - Fail2Ban Service.
Feb 14 08:00:46 kali fail2ban-server[25250]: Server ready
```

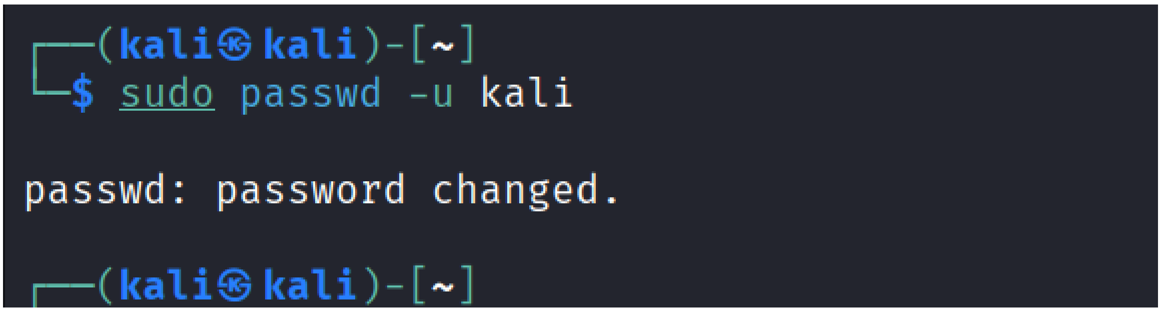
## 6. Restore systems to a secure state

Restoring systems to a secure state is the recovery phase of incident response, performed after the threat has been removed and root causes addressed. This step focuses on bringing the system back to normal operation while ensuring it is hardened against future attacks. In Linux environments, recovery typically involves unlocking legitimate user accounts, restarting essential services, applying updates, verifying file integrity, and monitoring logs for any remaining suspicious activity. The goal is to confirm that the system is clean, stable, and protected. Proper restoration ensures business continuity while maintaining a strong security posture.

### Step 1: Unlock Legitimate User Account (If Previously Locked)

If you locked your account earlier (example: kali):

sudo passwd -u kali



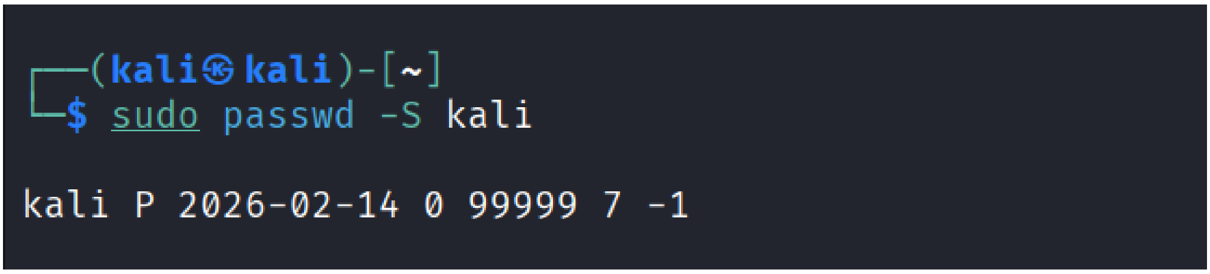
```
┌──(kali㉿kali)-[~]
└─$ sudo passwd -u kali

passwd: password changed.

┌──(kali㉿kali)-[~]
```

**Verify:**

sudo passwd -S kali

You should see:

kali P ...

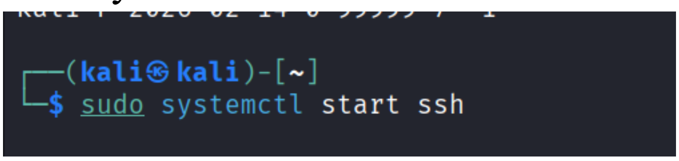


```
┌──(kali㉿kali)-[~]
└─$ sudo passwd -S kali

kali P 2026-02-14 0 99999 7 -1
```

### Step 2: Restart Essential Services (SSH)

If SSH was stopped during containment:

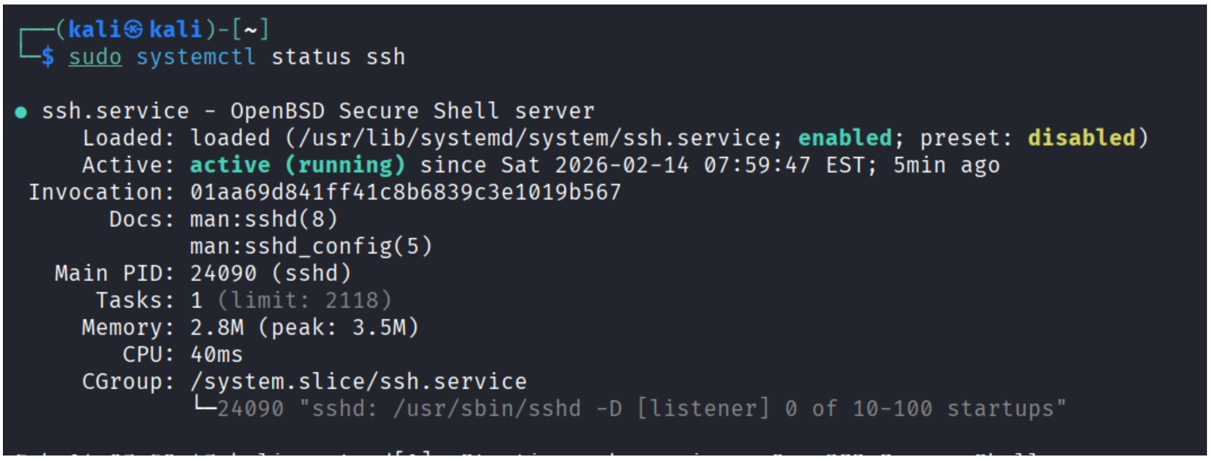
sudo systemctl start ssh



```
kali P 2026-02-14 0 99999 7 -1

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

Check status:

sudo systemctl status ssh

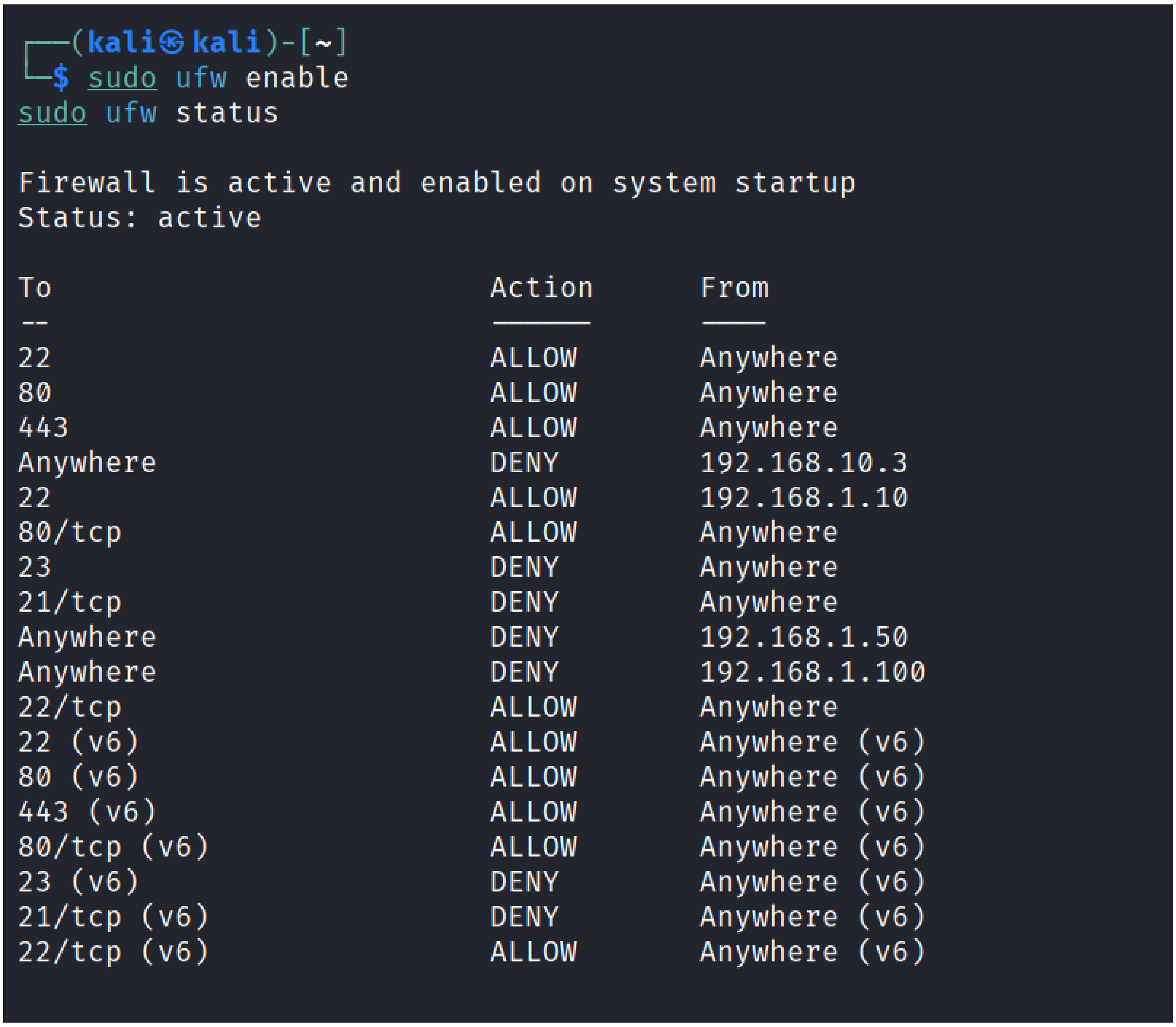


```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl status ssh

● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: disabled)
     Active: active (running) since Sat 2026-02-14 07:59:47 EST; 5min ago
 Invocation: 01aa69d841ff41c8b6839c3e1019b567
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 24090 (sshd)
      Tasks: 1 (limit: 2118)
     Memory: 2.8M (peak: 3.5M)
        CPU: 40ms
     CGroup: /system.slice/ssh.service
             └─24090 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
```

### Step 3: Ensure Firewall Is Enabled

sudo ufw enable

sudo ufw status

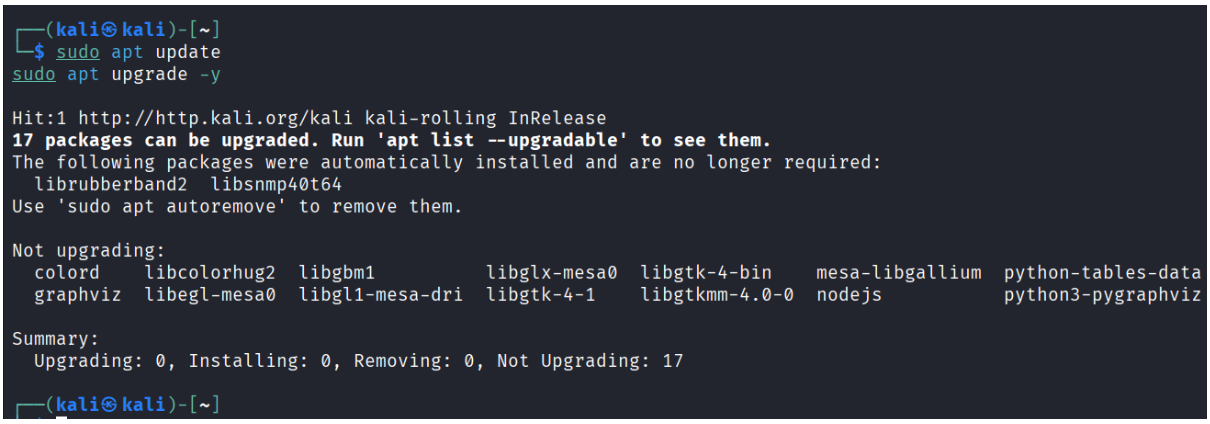


```
┌──(kali㉿kali)-[~]
└─$ sudo ufw enable
sudo ufw status

Firewall is active and enabled on system startup
Status: active

To                         Action      From
--                         ------      ----
22                         ALLOW       Anywhere
80                         ALLOW       Anywhere
443                        ALLOW       Anywhere
Anywhere                   DENY        192.168.10.3
22                         ALLOW       192.168.1.10
80/tcp                     ALLOW       Anywhere
23                         DENY        Anywhere
21/tcp                     DENY        Anywhere
Anywhere                   DENY        192.168.1.50
Anywhere                   DENY        192.168.1.100
22/tcp                     ALLOW       Anywhere
22 (v6)                    ALLOW       Anywhere (v6)
80 (v6)                    ALLOW       Anywhere (v6)
443 (v6)                   ALLOW       Anywhere (v6)
80/tcp (v6)                ALLOW       Anywhere (v6)
23 (v6)                    DENY        Anywhere (v6)
21/tcp (v6)                DENY        Anywhere (v6)
22/tcp (v6)                ALLOW       Anywhere (v6)
```

## Step 4: Verify System Is Fully Updated

sudo apt update

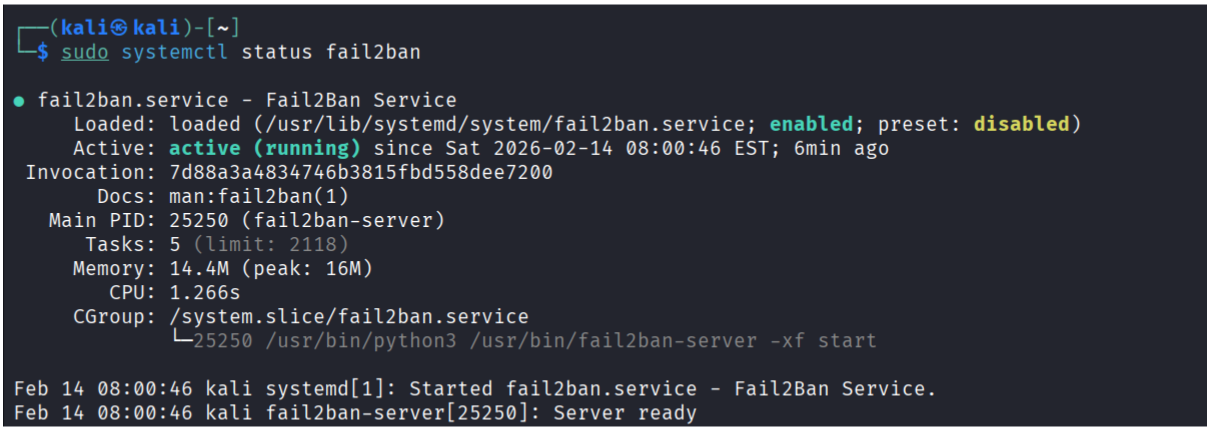
sudo apt upgrade -y



```
┌──(kali㉿kali)-[~]
└─$ sudo apt update
sudo apt upgrade -y

Hit:1 http://http.kali.org/kali kali-rolling InRelease
17 packages can be upgraded. Run 'apt list --upgradable' to see them.
The following packages were automatically installed and are no longer required:
  librubberband2  libsnmp40t64
Use 'sudo apt autoremove' to remove them.

Not upgrading:
  colord    libcolorhug2  libgbm1         libglx-mesa0  libgtk-4-bin    mesa-libgallium  python-tables-data
  graphviz  libegl-mesa0  libgl1-mesa-dri  libgtk-4-1    libgtkmm-4.0-0  nodejs           python3-pygraphviz

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 17

┌──(kali㉿kali)-[~]
```

## Step 5: Confirm Fail2Ban Is Running (If Installed)

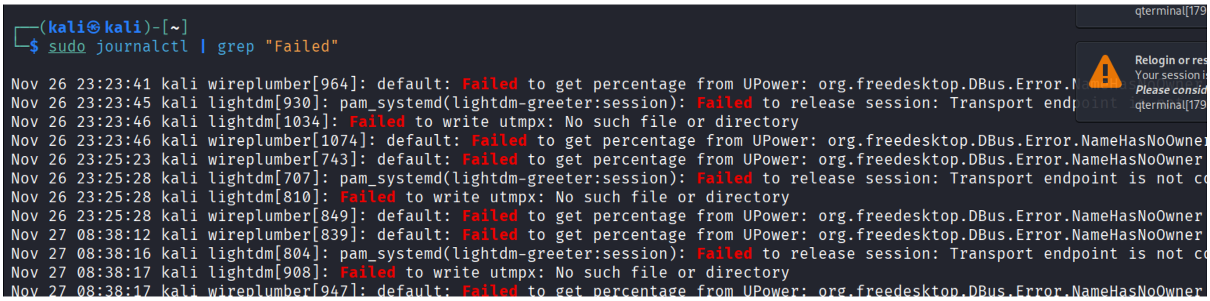
sudo systemctl status fail2ban



```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl status fail2ban

● fail2ban.service - Fail2Ban Service
     Loaded: loaded (/usr/lib/systemd/system/fail2ban.service; enabled; preset: disabled)
     Active: active (running) since Sat 2026-02-14 08:00:46 EST; 6min ago
 Invocation: 7d88a3a4834746b3815fbd558dee7200
       Docs: man:fail2ban(1)
   Main PID: 25250 (fail2ban-server)
      Tasks: 5 (limit: 2118)
     Memory: 14.4M (peak: 16M)
        CPU: 1.266s
     CGroup: /system.slice/fail2ban.service
             └─25250 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

Feb 14 08:00:46 kali systemd[1]: Started fail2ban.service - Fail2Ban Service.
Feb 14 08:00:46 kali fail2ban-server[25250]: Server ready
```

## Step 6: Monitor Logs for Any New Suspicious Activity

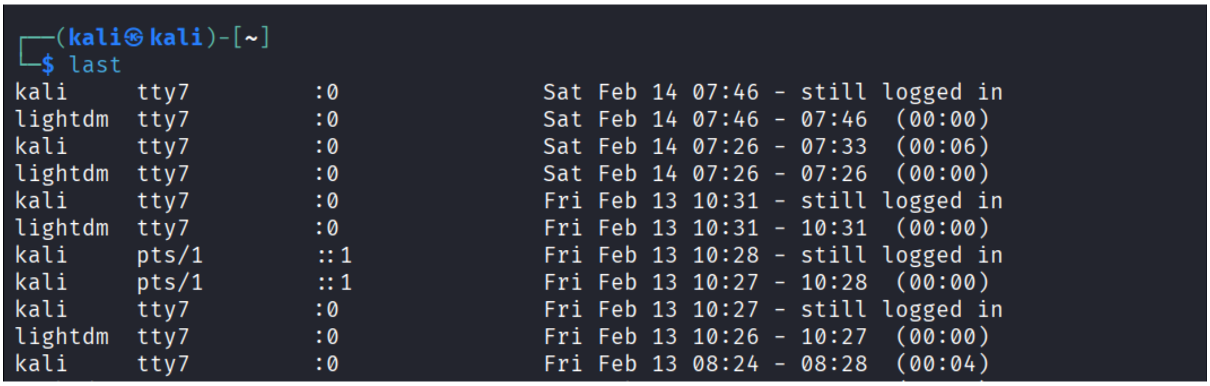
sudo journalctl | grep "Failed"



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed"

Nov 26 23:23:41 kali wireplumber[964]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.N
Nov 26 23:23:45 kali lightdm[930]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint
Nov 26 23:23:46 kali lightdm[1034]: Failed to write utmpx: No such file or directory
Nov 26 23:23:46 kali wireplumber[1074]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:25:23 kali wireplumber[743]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:25:28 kali lightdm[707]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not c
Nov 26 23:25:28 kali lightdm[810]: Failed to write utmpx: No such file or directory
Nov 26 23:25:28 kali wireplumber[849]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 27 08:38:12 kali wireplumber[839]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 27 08:38:16 kali lightdm[804]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not c
Nov 27 08:38:17 kali lightdm[908]: Failed to write utmpx: No such file or directory
Nov 27 08:38:17 kali wireplumber[947]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
```

and:

last



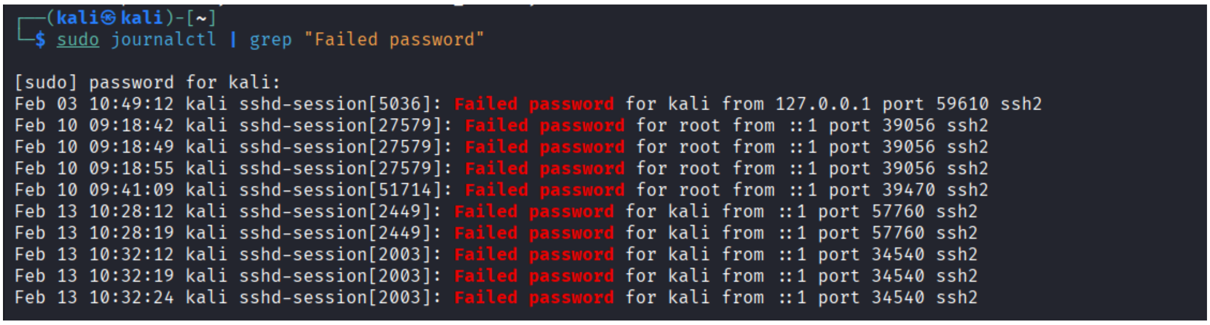
```
┌──(kali㉿kali)-[~]
└─$ last
kali     tty7         :0               Sat Feb 14 07:46 - still logged in
lightdm  tty7         :0               Sat Feb 14 07:46 - 07:46  (00:00)
kali     tty7         :0               Sat Feb 14 07:26 - 07:33  (00:06)
lightdm  tty7         :0               Sat Feb 14 07:26 - 07:26  (00:00)
kali     tty7         :0               Fri Feb 13 10:31 - still logged in
lightdm  tty7         :0               Fri Feb 13 10:31 - 10:31  (00:00)
kali     pts/1        ::1              Fri Feb 13 10:28 - still logged in
kali     pts/1        ::1              Fri Feb 13 10:27 - 10:28  (00:00)
kali     tty7         :0               Fri Feb 13 10:27 - still logged in
lightdm  tty7         :0               Fri Feb 13 10:26 - 10:27  (00:00)
kali     tty7         :0               Fri Feb 13 08:24 - 08:28  (00:04)
```

## 7. Document the incident timeline and actions taken.

Documenting the incident timeline is an essential part of incident response because it provides a clear record of what happened, when it happened, and how it was handled. A timeline typically includes detection of suspicious activity, identification of affected accounts or systems, containment actions, threat removal, and system recovery. In Linux environments, authentication and system logs supply timestamps and event details that help reconstruct the sequence of events. Proper documentation supports forensic analysis, improves future response procedures, and serves as evidence for audits or reports. A well-maintained timeline ensures accountability and helps prevent similar incidents in the future.

### Step 1: Extract Failed Login Events with Timestamps

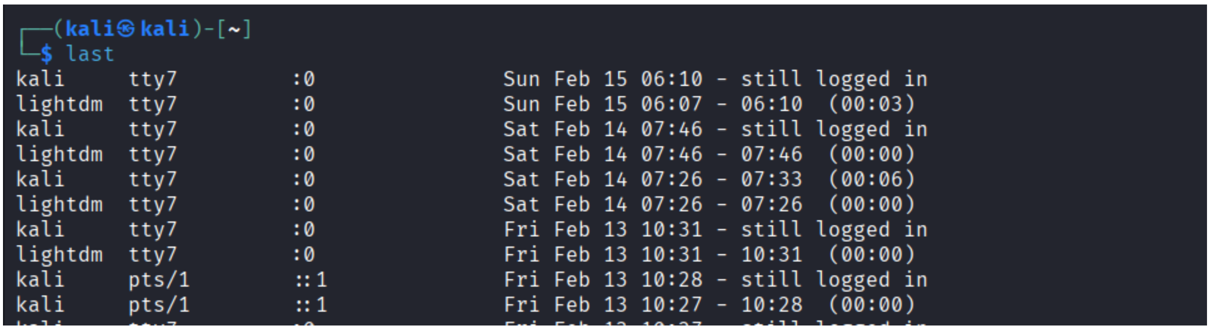
sudo journalctl | grep "Failed password"



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed password"

[sudo] password for kali:
Feb 03 10:49:12 kali sshd-session[5036]: Failed password for kali from 127.0.0.1 port 59610 ssh2
Feb 10 09:18:42 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:49 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:18:55 kali sshd-session[27579]: Failed password for root from ::1 port 39056 ssh2
Feb 10 09:41:09 kali sshd-session[51714]: Failed password for root from ::1 port 39470 ssh2
Feb 13 10:28:12 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:28:19 kali sshd-session[2449]: Failed password for kali from ::1 port 57760 ssh2
Feb 13 10:32:12 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:19 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
Feb 13 10:32:24 kali sshd-session[2003]: Failed password for kali from ::1 port 34540 ssh2
```

### Step 2: Check Successful Logins (If Any)

last

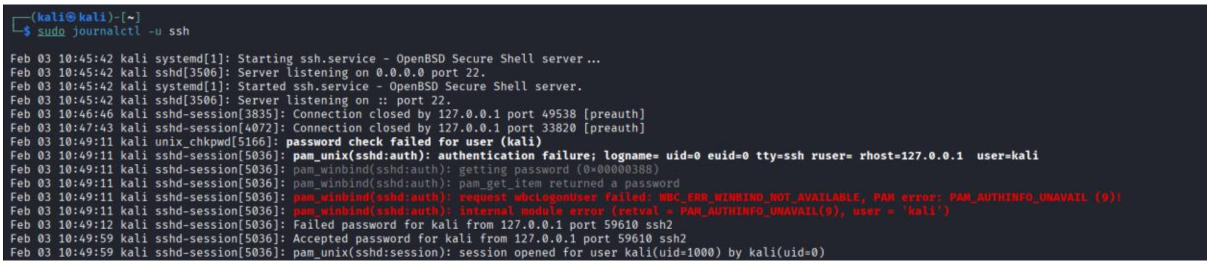


```
┌──(kali㉿kali)-[~]
└─$ last
kali     tty7         :0               Sun Feb 15 06:10 - still logged in
lightdm  tty7         :0               Sun Feb 15 06:07 - 06:10  (00:03)
kali     tty7         :0               Sat Feb 14 07:46 - still logged in
lightdm  tty7         :0               Sat Feb 14 07:46 - 07:46  (00:00)
kali     tty7         :0               Sat Feb 14 07:26 - 07:33  (00:06)
lightdm  tty7         :0               Sat Feb 14 07:26 - 07:26  (00:00)
kali     tty7         :0               Fri Feb 13 10:31 - still logged in
lightdm  tty7         :0               Fri Feb 13 10:31 - 10:31  (00:00)
kali     pts/1        ::1              Fri Feb 13 10:28 - still logged in
kali     pts/1        ::1              Fri Feb 13 10:27 - 10:28  (00:00)
```

### Step 3: Review Containment Actions (Service Stops / Account Locks)

Check SSH service history:

sudo journalctl -u ssh

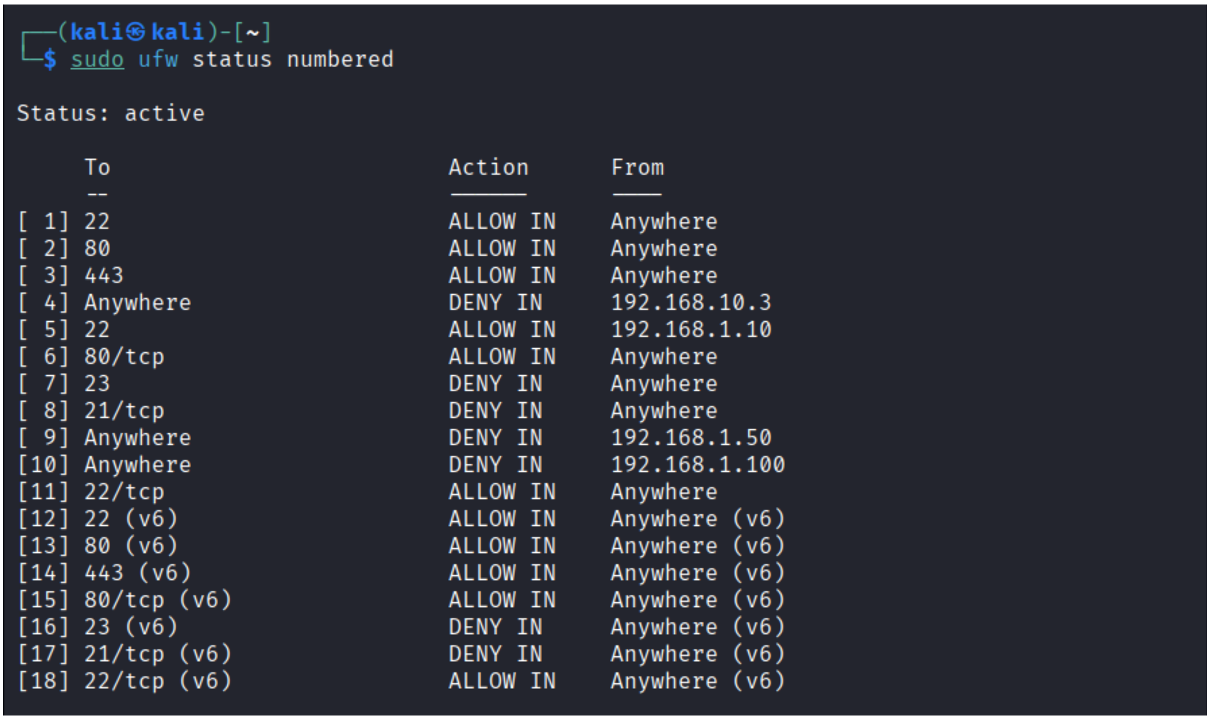


```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl -u ssh

Feb 03 10:45:42 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server ...
Feb 03 10:45:42 kali sshd[3506]: Server listening on 0.0.0.0 port 22.
Feb 03 10:45:42 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Feb 03 10:45:42 kali sshd[3506]: Server listening on :: port 22.
Feb 03 10:46:46 kali sshd-session[3835]: Connection closed by 127.0.0.1 port 49538 [preauth]
Feb 03 10:47:43 kali sshd-session[4072]: Connection closed by 127.0.0.1 port 33820 [preauth]
Feb 03 10:49:11 kali unix_chkpwd[5166]: password check failed for user (kali)
Feb 03 10:49:11 kali sshd-session[5036]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=127.0.0.1  user=kali
Feb 03 10:49:11 kali sshd-session[5036]: pam_winbind(sshd:auth): getting password (0×00000388)
Feb 03 10:49:11 kali sshd-session[5036]: pam_winbind(sshd:auth): pam_get_item returned a password
Feb 03 10:49:11 kali sshd-session[5036]: pam_winbind(sshd:auth): request wbcLogonUser failed: WBC_ERR_WINBIND_NOT_AVAILABLE, PAM error: PAM_AUTHINFO_UNAVAIL (9)!
Feb 03 10:49:11 kali sshd-session[5036]: pam_winbind(sshd:auth): internal module error (retval = PAM_AUTHINFO_UNAVAIL(9), user = 'kali')
Feb 03 10:49:12 kali sshd-session[5036]: Failed password for kali from 127.0.0.1 port 59610 ssh2
Feb 03 10:49:59 kali sshd-session[5036]: Accepted password for kali from 127.0.0.1 port 59610 ssh2
Feb 03 10:49:59 kali sshd-session[5036]: pam_unix(sshd:session): session opened for user kali(uid=1000) by kali(uid=0)
```

## Step 4: Verify Firewall or Fail2Ban Actions (If Used)

For UFW:

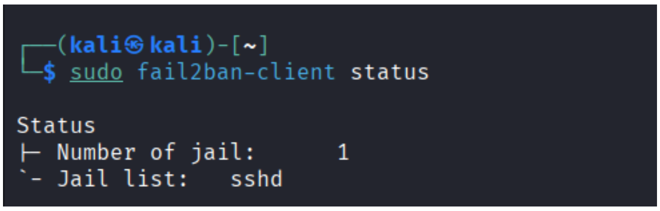
sudo ufw status numbered



```
┌──(kali㉿kali)-[~]
└─$ sudo ufw status numbered

Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 22                         ALLOW IN    Anywhere
[ 2] 80                         ALLOW IN    Anywhere
[ 3] 443                        ALLOW IN    Anywhere
[ 4] Anywhere                   DENY IN     192.168.10.3
[ 5] 22                         ALLOW IN    192.168.1.10
[ 6] 80/tcp                     ALLOW IN    Anywhere
[ 7] 23                         DENY IN     Anywhere
[ 8] 21/tcp                     DENY IN     Anywhere
[ 9] Anywhere                   DENY IN     192.168.1.50
[10] Anywhere                   DENY IN     192.168.1.100
[11] 22/tcp                     ALLOW IN    Anywhere
[12] 22 (v6)                    ALLOW IN    Anywhere (v6)
[13] 80 (v6)                    ALLOW IN    Anywhere (v6)
[14] 443 (v6)                   ALLOW IN    Anywhere (v6)
[15] 80/tcp (v6)                ALLOW IN    Anywhere (v6)
[16] 23 (v6)                    DENY IN     Anywhere (v6)
[17] 21/tcp (v6)                DENY IN     Anywhere (v6)
[18] 22/tcp (v6)                ALLOW IN    Anywhere (v6)
```

For Fail2Ban:

sudo fail2ban-client status

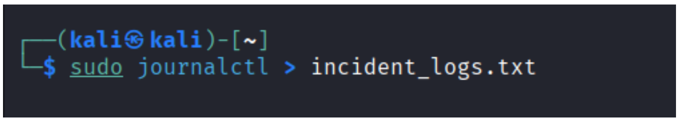


```
┌──(kali㉿kali)-[~]
└─$ sudo fail2ban-client status

Status
|- Number of jail:      1
`- Jail list:   sshd
```

## Step 5: Save Evidence (Optional but Recommended)

Export logs:

sudo journalctl > incident_logs.txt



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl > incident_logs.txt
```

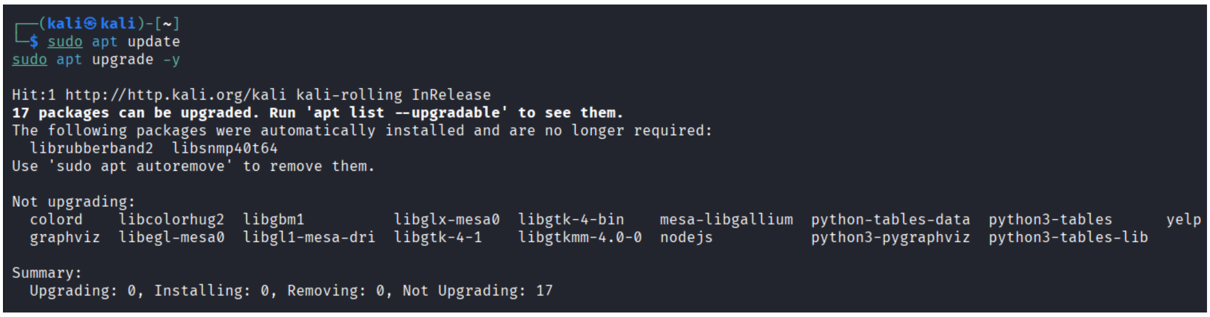
## 8. Recommend preventive security improvements

Preventive security improvements focus on reducing the likelihood of future incidents by strengthening system configuration, access controls, and monitoring mechanisms. After handling a brute-force or unauthorized access incident, Linux systems should be hardened by enforcing strong passwords, limiting remote access, enabling firewalls, keeping systems updated, and deploying intrusion-prevention tools. Continuous log monitoring and automated blocking mechanisms help detect attacks early, while proper user management minimizes insider threats. Implementing these preventive controls improves overall system resilience and helps maintain a secure operational environment.

### 1. Keep System Updated (Patch Vulnerabilities)

sudo apt update

sudo apt upgrade -y



```
┌──(kali㉿kali)-[~]
└─$ sudo apt update
sudo apt upgrade -y

Hit:1 http://http.kali.org/kali kali-rolling InRelease
17 packages can be upgraded. Run 'apt list --upgradable' to see them.
The following packages were automatically installed and are no longer required:
  librubberband2  libsnmp40t64
Use 'sudo apt autoremove' to remove them.

Not upgrading:
  colord    libcolorhug2  libgbm1          libglx-mesa0  libgtk-4-bin    mesa-libgallium  python-tables-data  python3-tables      yelp
  graphviz  libegl-mesa0  libgl1-mesa-dri  libgtk-4-1    libgtkmm-4.0-0  nodejs           python3-pygraphviz  python3-tables-lib

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 17
```

### 2. Enforce Strong Passwords

Edit PAM password policy:

sudo nano /etc/security/pwquality.conf

Set (or confirm):

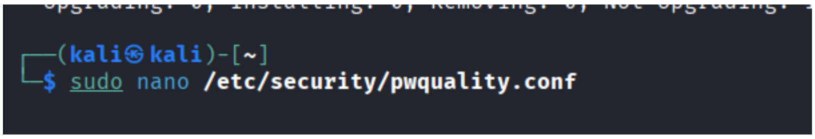
minlen = 10

ucredit = -1

lcredit = -1

dcredit = -1

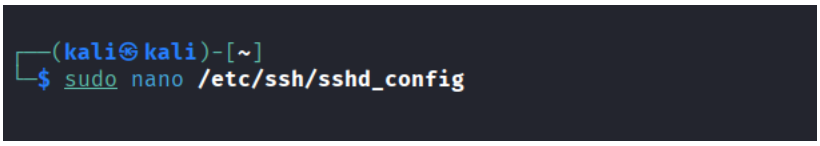
ocredit = -1



```
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1

┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/security/pwquality.conf
```

### 3. Disable Root Login via SSH

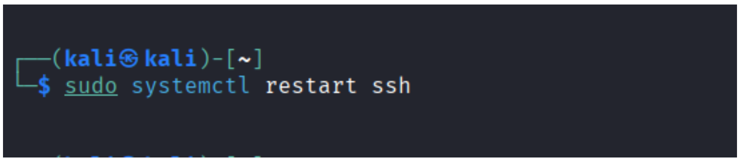
sudo nano /etc/ssh/sshd_config



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

Restart SSH:

sudo systemctl restart ssh



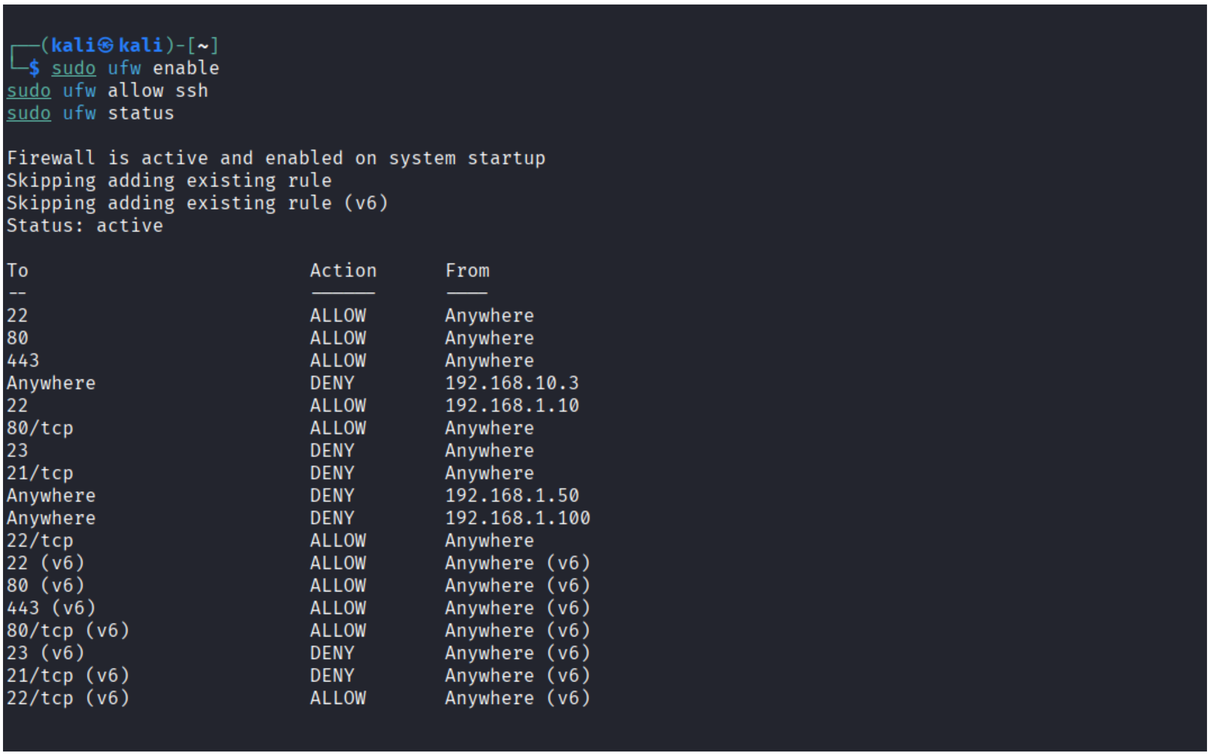
```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl restart ssh
```

### 4. Enable Firewall (UFW)

sudo ufw enable

sudo ufw allow ssh

sudo ufw status



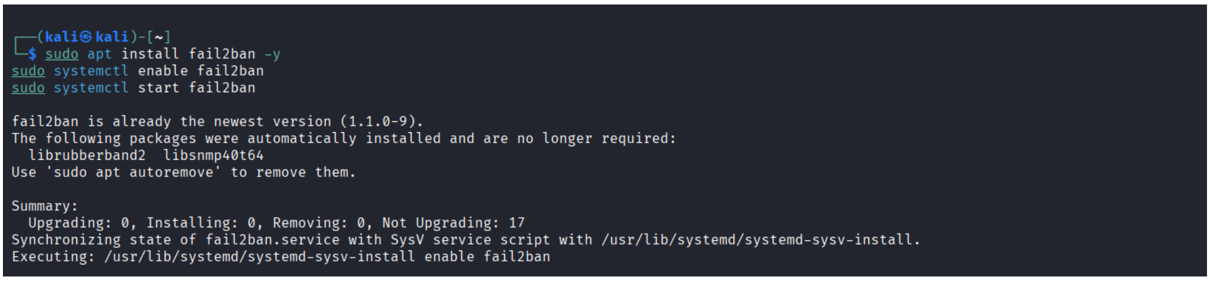
```
┌──(kali㉿kali)-[~]
└─$ sudo ufw enable
sudo ufw allow ssh
sudo ufw status

Firewall is active and enabled on system startup
Skipping adding existing rule
Skipping adding existing rule (v6)
Status: active

To                         Action      From
--                         ------      ----
22                         ALLOW       Anywhere
80                         ALLOW       Anywhere
443                        ALLOW       Anywhere
Anywhere                   DENY        192.168.10.3
22                         ALLOW       192.168.1.10
80/tcp                     ALLOW       Anywhere
23                         DENY        Anywhere
21/tcp                     DENY        Anywhere
Anywhere                   DENY        192.168.1.50
Anywhere                   DENY        192.168.1.100
22/tcp                     ALLOW       Anywhere
22 (v6)                    ALLOW       Anywhere (v6)
80 (v6)                    ALLOW       Anywhere (v6)
443 (v6)                   ALLOW       Anywhere (v6)
80/tcp (v6)                ALLOW       Anywhere (v6)
23 (v6)                    DENY        Anywhere (v6)
21/tcp (v6)                DENY        Anywhere (v6)
22/tcp (v6)                ALLOW       Anywhere (v6)
```

### 5. Install Fail2Ban (Auto-block Brute Force)

sudo apt install fail2ban -y

sudo systemctl enable fail2ban

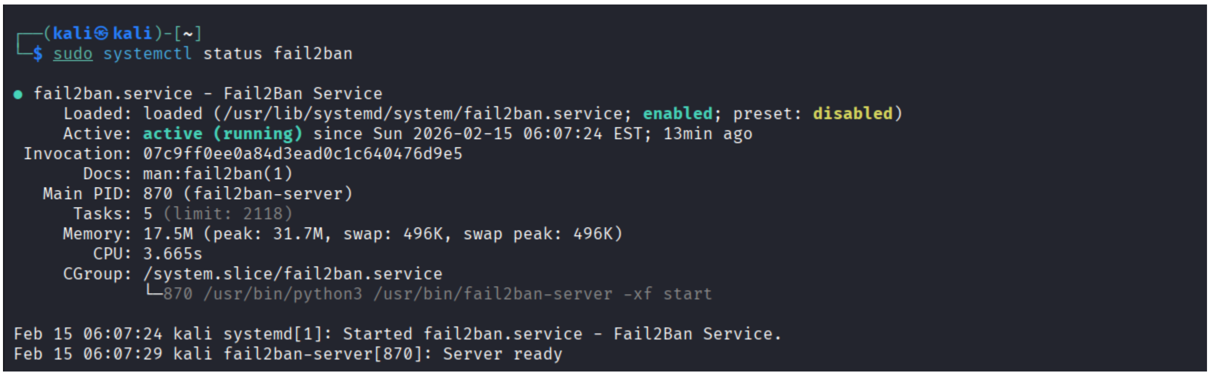
sudo systemctl start fail2ban



```
┌──(kali㉿kali)-[~]
└─$ sudo apt install fail2ban -y
sudo systemctl enable fail2ban
sudo systemctl start fail2ban

fail2ban is already the newest version (1.1.0-9).
The following packages were automatically installed and are no longer required:
  librubberband2  libsnmp40t64
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 17
Synchronizing state of fail2ban.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable fail2ban
```

Check:

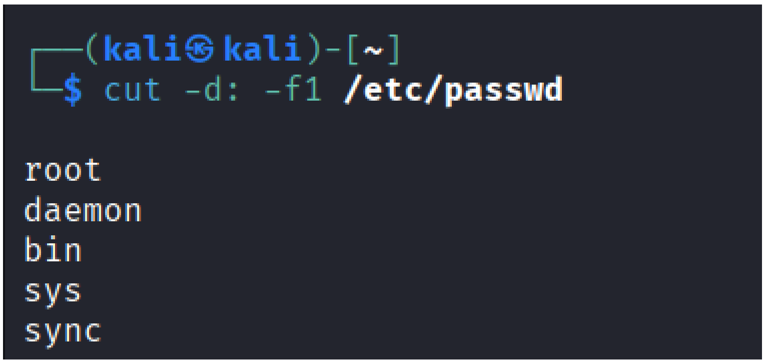
sudo systemctl status fail2ban



```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl status fail2ban

● fail2ban.service - Fail2Ban Service
     Loaded: loaded (/usr/lib/systemd/system/fail2ban.service; enabled; preset: disabled)
     Active: active (running) since Sun 2026-02-15 06:07:24 EST; 13min ago
 Invocation: 07c9ff0ee0a84d3ead0c1c640476d9e5
       Docs: man:fail2ban(1)
   Main PID: 870 (fail2ban-server)
      Tasks: 5 (limit: 2118)
     Memory: 17.5M (peak: 31.7M, swap: 496K, swap peak: 496K)
        CPU: 3.665s
     CGroup: /system.slice/fail2ban.service
             └─870 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

Feb 15 06:07:24 kali systemd[1]: Started fail2ban.service - Fail2Ban Service.
Feb 15 06:07:29 kali fail2ban-server[870]: Server ready
```

## 6. Remove Unnecessary Users

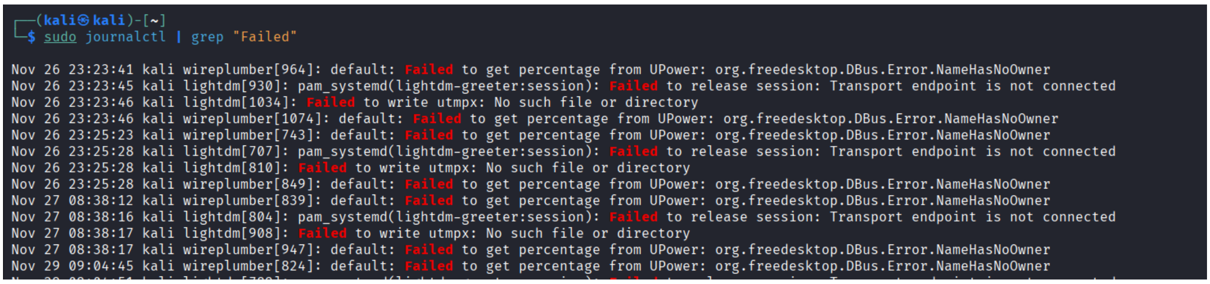
cut -d: -f1 /etc/passwd



```
┌──(kali㉿kali)-[~]
└─$ cut -d: -f1 /etc/passwd

root
daemon
bin
sys
sync
```

## 7. Regular Log Monitoring

sudo journalctl | grep "Failed"



```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl | grep "Failed"

Nov 26 23:23:41 kali wireplumber[964]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:23:45 kali lightdm[930]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
Nov 26 23:23:46 kali lightdm[1034]: Failed to write utmpx: No such file or directory
Nov 26 23:23:46 kali wireplumber[1074]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:25:23 kali wireplumber[743]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 26 23:25:28 kali lightdm[707]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
Nov 26 23:25:28 kali lightdm[810]: Failed to write utmpx: No such file or directory
Nov 26 23:25:28 kali wireplumber[849]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 27 08:38:12 kali wireplumber[839]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 27 08:38:16 kali lightdm[804]: pam_systemd(lightdm-greeter:session): Failed to release session: Transport endpoint is not connected
Nov 27 08:38:17 kali lightdm[908]: Failed to write utmpx: No such file or directory
Nov 27 08:38:17 kali wireplumber[947]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
Nov 29 09:04:45 kali wireplumber[824]: default: Failed to get percentage from UPower: org.freedesktop.DBus.Error.NameHasNoOwner
```

## Conclusion :-

In this lab, a basic security incident was simulated by generating repeated failed login attempts on a Linux system (such as Kali Linux). System and authentication logs were analyzed to identify suspicious activity, and the incident was classified based on attack type and severity. Appropriate containment actions were taken by isolating affected accounts and services. The threat was then removed by resetting passwords, blocking malicious access, and fixing configuration weaknesses. Afterward, the system was restored to a secure state by restarting essential services and verifying security controls. An incident timeline was documented using log data, and preventive security measures such as system updates, strong passwords, firewall configuration, and log monitoring were recommended. Overall, this exercise demonstrated the complete incident response process and highlighted the importance of proactive monitoring and security hardening to protect Linux systems from future attacks.