Log Analysis & Intrusion Detection

1. Setup

**Enabling System Logging **

Kali Linux uses systemd for logging instead of /var/log/auth.log by default.
Ensure logging is enabled:

sudo systemctl status systemd-journald
If it’s not running, enable and start it:

sudo systemctl enable --now systemd-journald

To persist logs across reboots, modify /etc/systemd/journald.conf:

sudo nano /etc/systemd/journald.conf

Ensure the following lines are set:

Storage=persistent

Then restart the service:

sudo systemctl restart systemd-journald

```
└─$ sudo systemctl enable --now systemd-journald
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.
```

Checking SSH Logs on Kali Linux

To monitor SSH authentication logs:

sudo journalctl -u ssh --no-pager
For real-time monitoring:

sudo journalctl -u ssh -f
If /var/log/auth.log exists, verify SSH logs using:

sudo tail -f /var/log/auth.log
Image

2. Exploit - Simulating Multiple Failed SSH Login Attempts

To simulate a brute-force attack:

for i in {1..10}; do ssh invaliduser@localhost; done
This will generate failed SSH attempts recorded in the system logs.

Analyzing Logs for Failed SSH Attempts

If logs are stored in journalctl, analyze failures using:

sudo journalctl -u ssh | grep "Failed password"

Or if /var/log/auth.log exists:

grep "Failed password" /var/log/auth.log

Example output:

Mar 25 12:34:56 kali sshd[12345]: Failed password for invaliduser from 192.168.1.100 port 54721 ssh2
Mar 25 12:34:57 kali sshd[12346]: Failed password for invaliduser from 192.168.1.100 port 54722 ssh2

```
└─$ sudo journalctl -u ssh | grep "Failed password"
Mar 11 13:34:36 nexulean sshd-session[57198]: Failed password for nexulean from 192.168.0.118 port 52072 ssh2
Mar 11 13:34:36 nexulean sshd-session[57200]: Failed password for nexulean from 192.168.0.118 port 52090 ssh2
Mar 11 13:34:36 nexulean sshd-session[57199]: Failed password for nexulean from 192.168.0.118 port 52080 ssh2
Mar 11 13:34:36 nexulean sshd-session[57202]: Failed password for nexulean from 192.168.0.118 port 52092 ssh2
Mar 11 13:34:38 nexulean sshd-session[57199]: Failed password for nexulean from 192.168.0.118 port 52080 ssh2
Mar 11 13:34:38 nexulean sshd-session[57202]: Failed password for nexulean from 192.168.0.118 port 52092 ssh2
Mar 11 13:34:54 nexulean sshd-session[57357]: Failed password for nexulean from 192.168.0.118 port 51810 ssh2
Mar 11 13:34:54 nexulean sshd-session[57359]: Failed password for nexulean from 192.168.0.118 port 51814 ssh2
Mar 11 13:34:54 nexulean sshd-session[57361]: Failed password for nexulean from 192.168.0.118 port 51808 ssh2
Mar 11 13:34:54 nexulean sshd-session[57358]: Failed password for nexulean from 192.168.0.118 port 51812 ssh2
Mar 11 13:34:58 nexulean sshd-session[57357]: Failed password for nexulean from 192.168.0.118 port 51810 ssh2
Mar 11 13:34:59 nexulean sshd-session[57361]: Failed password for nexulean from 192.168.0.118 port 51808 ssh2
Mar 11 13:34:59 nexulean sshd-session[57359]: Failed password for nexulean from 192.168.0.118 port 51814 ssh2
Mar 11 13:36:12 nexulean sshd-session[58027]: Failed password for root from 192.168.0.118 port 55808 ssh2
Mar 11 13:36:12 nexulean sshd-session[58028]: Failed password for root from 192.168.0.118 port 55806 ssh2
Mar 11 13:36:12 nexulean sshd-session[58026]: Failed password for root from 192.168.0.118 port 55798 ssh2
Mar 11 13:36:12 nexulean sshd-session[58029]: Failed password for root from 192.168.0.118 port 55826 ssh2
Mar 11 13:36:15 nexulean sshd-session[58027]: Failed password for root from 192.168.0.118 port 55808 ssh2
Mar 11 13:36:15 nexulean sshd-session[58028]: Failed password for root from 192.168.0.118 port 55806 ssh2
Mar 11 13:36:15 nexulean sshd-session[58029]: Failed password for root from 192.168.0.118 port 55826 ssh2
Mar 11 13:36:15 nexulean sshd-session[58026]: Failed password for root from 192.168.0.118 port 55798 ssh2
Mar 11 13:36:18 nexulean sshd-session[58027]: Failed password for root from 192.168.0.118 port 55808 ssh2
Mar 11 13:36:18 nexulean sshd-session[58028]: Failed password for root from 192.168.0.118 port 55806 ssh2
Mar 11 13:36:18 nexulean sshd-session[58029]: Failed password for root from 192.168.0.118 port 55826 ssh2
Mar 11 13:36:18 nexulean sshd-session[58026]: Failed password for root from 192.168.0.118 port 55798 ssh2
```

Detecting Brute-force Attacks
To count failed attempts per IP:

sudo journalctl -u ssh | grep "Failed password" | awk '{print $(NF-3)}' | sort | uniq -c | sort -nr

Example output:

10 192.168.1.100
5 192.168.1.101

```
└─$ grep "Failed password" /var/log/auth.log
grep: /var/log/auth.log: No such file or directory
└─$ sudo journalctl -u ssh | grep "Failed password" | awk '{print $(NF-3)}' | sort | uniq -c | sort -nr
    96 192.168.0.118
```

This helps identify brute-force attempts.

3. Mitigation Measures

Installing and Configuring Fail2Ban on Kali Linux
Fail2Ban prevents brute-force attacks by banning IPs after repeated failures.

Install Fail2Ban
sudo apt update && sudo apt install fail2ban -y
Configure SSH Protection
Edit the jail configuration file:

sudo nano /etc/fail2ban/jail.local
Add the following:

[sshd]
enabled = true
port = ssh
maxretry = 3
findtime = 600
bantime = 3600
logpath = %(syslog_auth)s
Restart fail2ban:

sudo systemctl restart fail2ban

```
└─$ sudo apt update && sudo apt install fail2ban -y
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 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) [49.4 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [115 kB]
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb) [268 kB]
Get:6 http://kali.download/kali kali-rolling/non-free amd64 Packages [195 kB]
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [880 kB]
Get:8 http://kali.download/kali kali-rolling/non-free-firmware amd64 Packages [10.6 kB]
Get:9 http://kali.download/kali kali-rolling/non-free-firmware amd64 Contents (deb) [24.3 kB]
Fetched 71.7 MB in 47s (1,538 kB/s)
1823 packages can be upgraded. Run 'apt list --upgradable' to see them.
fail2ban is already the newest version (1.1.0-7).
The following packages were automatically installed and are no longer required:
  libpython3.12-dev  python3.12-dev
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1823
```

Check banned IPs:

sudo fail2ban-client status sshd
Setting up Log Monitoring with Logwatch
Logwatch provides automated log analysis.

sudo apt install logwatch -y
Generate a report:

sudo logwatch --detail high --mailto admin@example.com
Setting up Rsyslog for Remote Logging
Enable Rsyslog:

sudo systemctl enable --now rsyslog
Edit /etc/rsyslog.conf to enable remote logging:

. @192.168.1.200:514
Restart Rsyslog:

sudo systemctl restart rsyslog

4. Conclusion

Logs were successfully analyzed for failed SSH attempts.
Brute-force attacks were detected and mitigated.
Fail2Ban was deployed to block repeated login failures.
Logwatch and Rsyslog were set up for automated log monitoring.
journalctl was used in place of missing traditional log files.