Linux Basics for Security Analysts

Name : Syed Azharuddin Ali Khan
Role : Cybersecurity Analyst Intern
Date : 01/16/2026

Objective

To prepare for real-world cybersecurity environments by building **strong operational familiarity with Linux**, focusing on core commands used daily by **security analysts, SOC teams, penetration testers, and system administrators**.

Why Linux Is Widely Used in Cybersecurity

Linux is the **backbone of modern cybersecurity operations** for several critical reasons:

1. Dominance in Servers & Infrastructure

- Most **web servers**, **databases**, **cloud platforms**, **firewalls**, and **SIEM backends** run on Linux.
- Enterprise tools like **Splunk**, **ELK Stack**, **Wazuh**, **Suricata**, and **Snort** are Linux-native.

2. Transparency & Open Source Security

- Linux source code is **openly available**, enabling:
    - Security auditing
    - Faster vulnerability discovery
    - Community-driven patching
- Analysts can **verify behavior**, not blindly trust binaries.

3. Native Security Tooling

Linux is home to:

- Network tools (netstat, ss, tcpdump)
- Process tools (ps, top, htop)
- Log analysis tools (grep, awk, sed)
- Permission & access control systems (file modes, SELinux, AppArmor)

4. Used by Attackers and Defenders

- **Attackers** use Linux-based systems (Kali, Parrot OS)
- **Defenders** protect Linux servers
- A security analyst must **think like both**

5. Automation & Scripting

- Bash scripting enables:
    - Log analysis automation
    - Incident response workflows

- Scheduled security checks

Conclusion: If you understand Linux, you understand **how systems really behave under attack**.

Core Linux Commands for Security Analysts

---

1. ls – List Directory Contents

Purpose

Displays files and directories. Used extensively during:

- File system investigation
- Malware hunting
- Log discovery

Common Usage

ls

ls -l

ls -a

ls -lh

Security-Relevant Flags

- -l → Shows permissions, owner, group
- -a → Shows hidden files (often abused by attackers)
- -h → Human-readable sizes

Example Output Explanation

-rw-r--r-- 1 root root 2048 auth.log

- **Permissions**: rw-r--r--
- **Owner**: root
- **File Size**: 2048 bytes

Security Use Case

- Detect **suspicious hidden files**
- Identify **unauthorized executables**
- Check ownership changes after compromise

```
jamie@debian:~$ ls -l
total 19208
-rwxr-xr-x 1 root  root   4703728 Dec 17 07:01 Battle.net-Setup.exe
drwxr-xr-x 3 jamie jamie     4096 Nov  5 03:30 Desktop
drwxr-xr-x 2 jamie jamie     4096 Jun  5  2018 Documents
drwxr-xr-x 3 jamie jamie     4096 Dec 17 06:49 Downloads
-rw-r--r-- 1 jamie jamie   179765 Dec 17 07:01 Linux_for_beginners.pdf
-rw-r--r-- 1 jamie jamie   458980 Dec 17 06:59 metamorphose2_0.8.2-1_all.deb
drwxr-xr-x 2 jamie jamie     4096 Apr 30  2018 Music
-rw-r--r-- 1 jamie jamie     1520 Dec 17 07:01 Neofetch
-rw-r--r-- 1 root  root  13902480 Dec 17 07:01 pdfsam_3.3.6-1_all.deb
-rw------- 1 root  root    375728 Dec 17 07:01 PDFsam_merge.pdf
drwxr-xr-x 2 jamie jamie     4096 Apr 30  2018 Pictures
drwxr-xr-x 2 jamie jamie     4096 Apr 30  2018 Public
drwxr-xr-x 2 jamie jamie     4096 Apr 30  2018 Templates
drwxr-xr-x 2 jamie jamie     4096 Apr 30  2018 Videos
```

```
ssh webserver
[joshuad@          ~/public_html/jesterblocks/filebase ]$ ls -la
total 24K
drwxrwxrwx 5 joshuad joshuad 4.0K Nov  8 15:55 ./
drwxr-xr-x 8 joshuad joshuad 4.0K Nov  7 16:08 ../
-rw-rw-rw- 1 joshuad joshuad   17 Nov  8 15:14 .htaccess
drwxrwxrwx 2 joshuad joshuad 4.0K Nov  8 15:05 beta/
drwxrwxrwx 2 joshuad joshuad 4.0K Nov  8 15:05 nightly-builds/
drwxr-xr-x 2 joshuad joshuad 4.0K Nov  8 18:13 test/
[joshuad@          ~/public_html/jesterblocks/filebase ]$
```

---

2. cd – Change Directory

Purpose

Navigates between directories.

cd /var/log

cd ..

cd ~

Security Context

- Analysts move between:
    - /var/log → system & auth logs
    - /etc → configuration files

- /home → user directories (insider threat checks)

Security Use Case

- Jump quickly to log directories during **incident response**
- Traverse compromised user folders

---

3. cat – View File Contents

Purpose

Displays entire file contents.

cat auth.log

Security Relevance

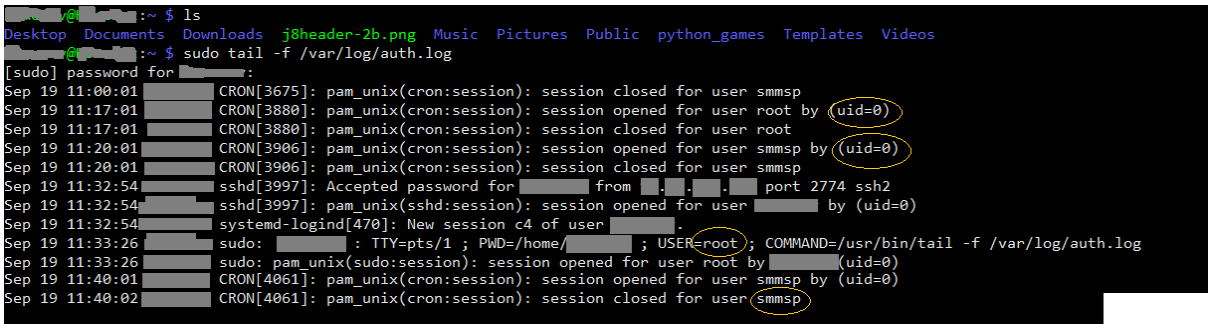
- Inspect:
  - Authentication logs
  - Configuration files
  - Malware scripts

⚠ Risk

- Large files may flood the terminal
- Logs may contain sensitive data

Security Use Case

- Verify **SSH login attempts**
- Inspect **crontab files planted by attackers**



---

4. tail – View Recent File Entries

Purpose

Displays the **last lines of a file**.

tail auth.log

tail -n 50 auth.log

tail -f auth.log

Why Analysts Love tail

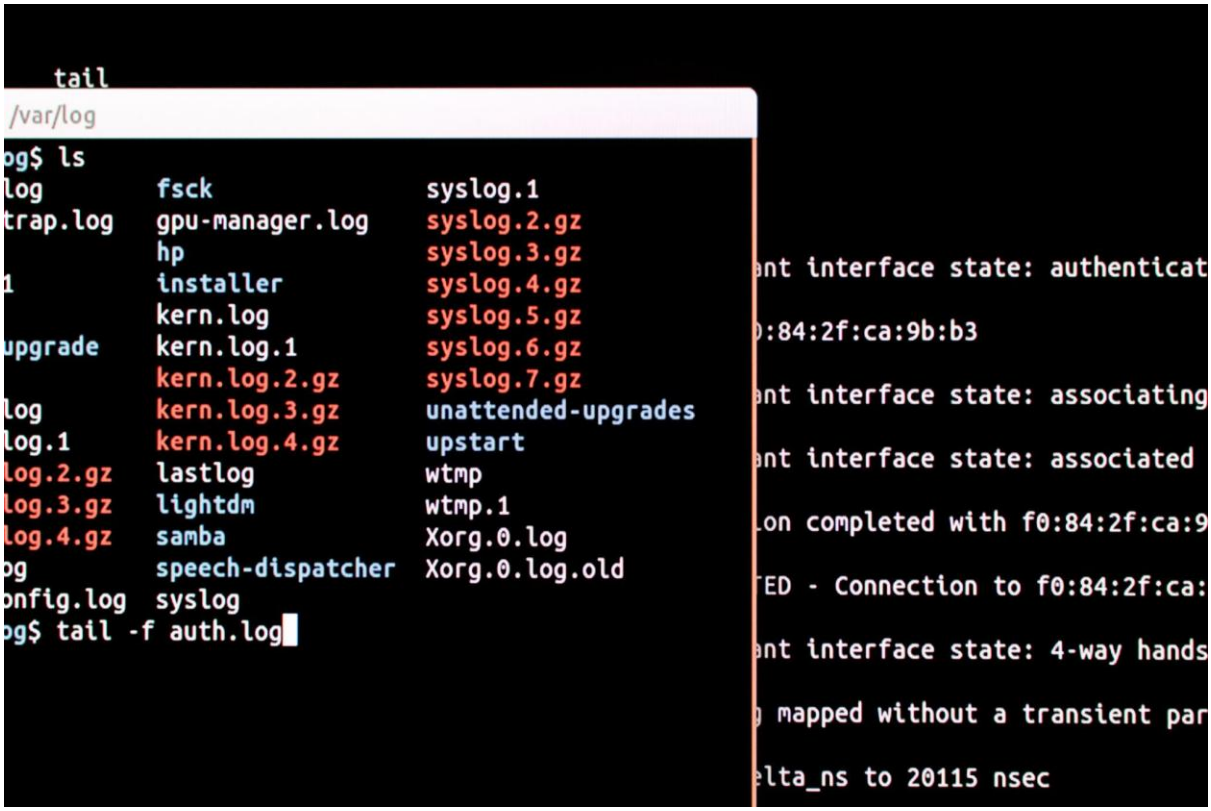
- Logs grow constantly
- You want **latest activity**, not old data

-f (Follow Mode)

- Real-time monitoring
- Used during **active attacks**

Security Use Case

- Watch live:
    - Brute-force SSH attempts
    - Failed sudo attempts
    - Suspicious service restarts



5. grep – Search Text Patterns

Purpose

Searches for keywords inside files.

grep "Failed password" auth.log

grep -i error syslog

Security Power

This is one of the **most critical tools** in cybersecurity.

Common Security Searches

grep "root" auth.log

grep "sudo" auth.log

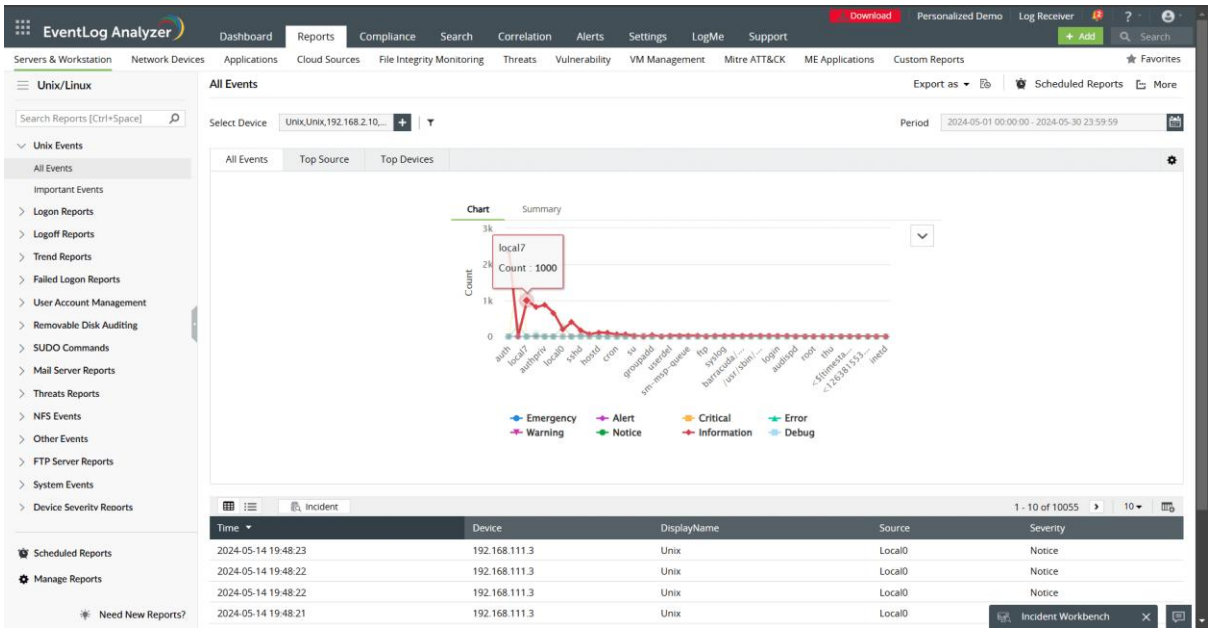
grep "Accepted" auth.log

Flags

- -i → Case insensitive
- -r → Recursive (scan directories)
- -n → Line numbers

Security Use Case

- Detect brute-force attacks
- Find privilege escalation attempts
- Hunt Indicators of Compromise (IOCs)



```
newubuntu@newubuntu-VirtualBox:~$ sudo aureport -au -i --failed

Authentication Report
============================================
# date time acct host term exe success event
============================================
1. 2017.11.02 18:05:05 newubuntu ? :0 /usr/sbin/lightdm no 115
2. 2017.11.02 18:05:08 newubuntu ? :0 /usr/sbin/lightdm no 116
3. 2017.11.02 18:05:13 ripon ? :0 /usr/sbin/lightdm no 117
4. 2017.11.02 18:05:16 ripon ? :0 /usr/sbin/lightdm no 118
5. 2017.11.02 18:05:25 ripon ? :0 /usr/sbin/lightdm no 128
6. 2017.11.02 18:05:28 ripon ? :0 /usr/sbin/lightdm no 129
7. 2017.11.02 18:05:36 newubuntu ? :0 /usr/sbin/lightdm no 133
newubuntu@newubuntu-VirtualBox:~$
```

---

6. chmod – Change File Permissions

Purpose

Controls **who can read, write, or execute files**.

chmod 755 script.sh

chmod 600 private.key

Permission Model

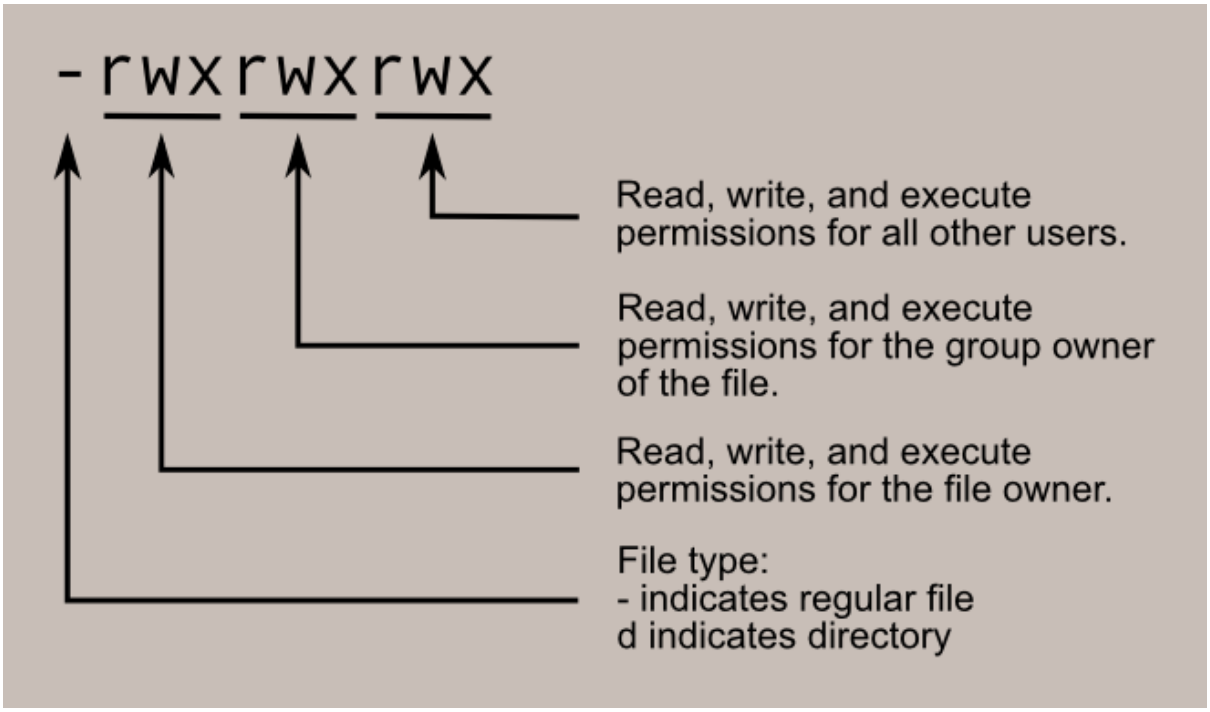
Value	Meaning
r	read
w	write
x	execute

Security Importance

- Prevent unauthorized execution
- Protect sensitive keys & configs
- Attackers often abuse **over-permissive files**

Security Use Case

- Lock down SSH keys
- Fix misconfigured scripts
- Detect **777 permissions (high risk)**



---

7. ps – View Running Processes

Purpose

Displays active processes.

ps

ps aux

Security-Relevant Fields

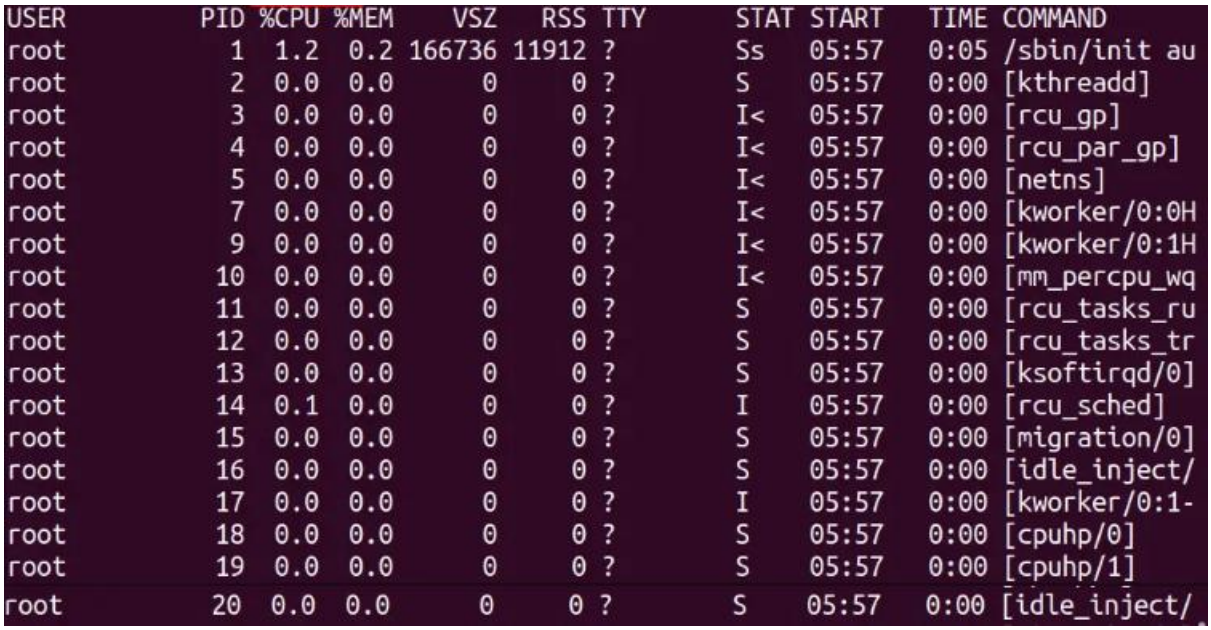
- **PID** → Process ID
- **USER** → Who launched it
- **COMMAND** → What is running

Security Use Case

- Detect:
    - Crypto miners
    - Backdoors

- Unauthorized scripts

- Validate if a suspicious process is running as root



USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	1.2	0.2	166736	11912	?	Ss	05:57	0:05	/sbin/init au
root	2	0.0	0.0	0	0	?	S	05:57	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	I<	05:57	0:00	[rcu_gp]
root	4	0.0	0.0	0	0	?	I<	05:57	0:00	[rcu_par_gp]
root	5	0.0	0.0	0	0	?	I<	05:57	0:00	[netns]
root	7	0.0	0.0	0	0	?	I<	05:57	0:00	[kworker/0:0H
root	9	0.0	0.0	0	0	?	I<	05:57	0:00	[kworker/0:1H
root	10	0.0	0.0	0	0	?	I<	05:57	0:00	[mm_percpu_wq
root	11	0.0	0.0	0	0	?	S	05:57	0:00	[rcu_tasks_ru
root	12	0.0	0.0	0	0	?	S	05:57	0:00	[rcu_tasks_tr
root	13	0.0	0.0	0	0	?	S	05:57	0:00	[ksoftirqd/0]
root	14	0.1	0.0	0	0	?	I	05:57	0:00	[rcu_sched]
root	15	0.0	0.0	0	0	?	S	05:57	0:00	[migration/0]
root	16	0.0	0.0	0	0	?	S	05:57	0:00	[idle_inject/
root	17	0.0	0.0	0	0	?	I	05:57	0:00	[kworker/0:1-
root	18	0.0	0.0	0	0	?	S	05:57	0:00	[cpuhp/0]
root	19	0.0	0.0	0	0	?	S	05:57	0:00	[cpuhp/1]
root	20	0.0	0.0	0	0	?	S	05:57	0:00	[idle_inject/

---

8. netstat – Network Connections

Purpose

Displays open ports and active connections.

netstat -tulnp

Critical Flags

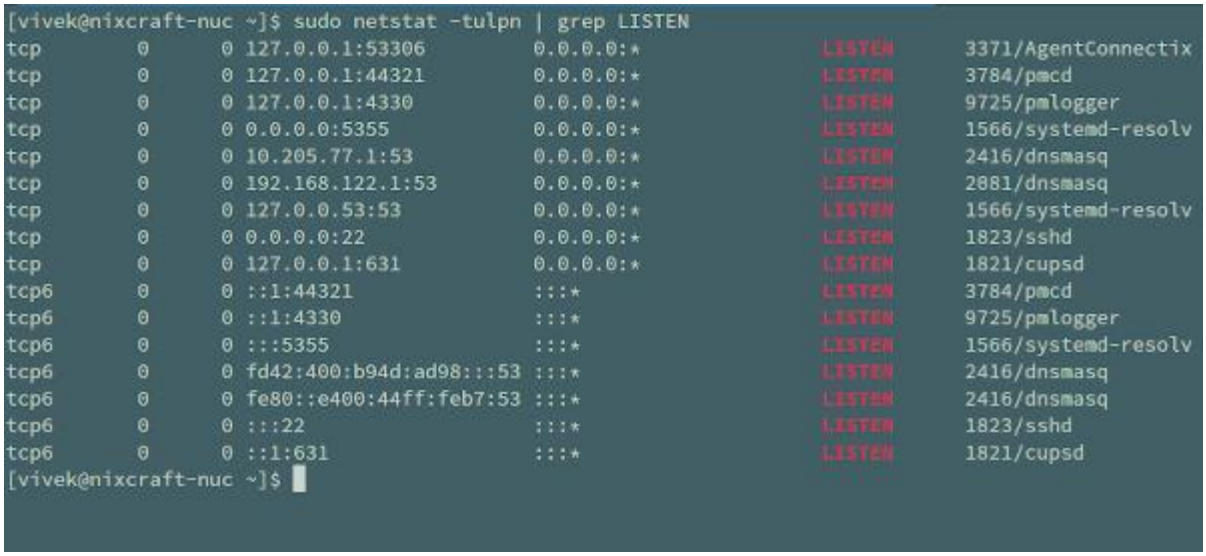
- -t → TCP
- -u → UDP
- -l → Listening ports
- -n → Numerical output
- -p → Process using the port

Security Use Case

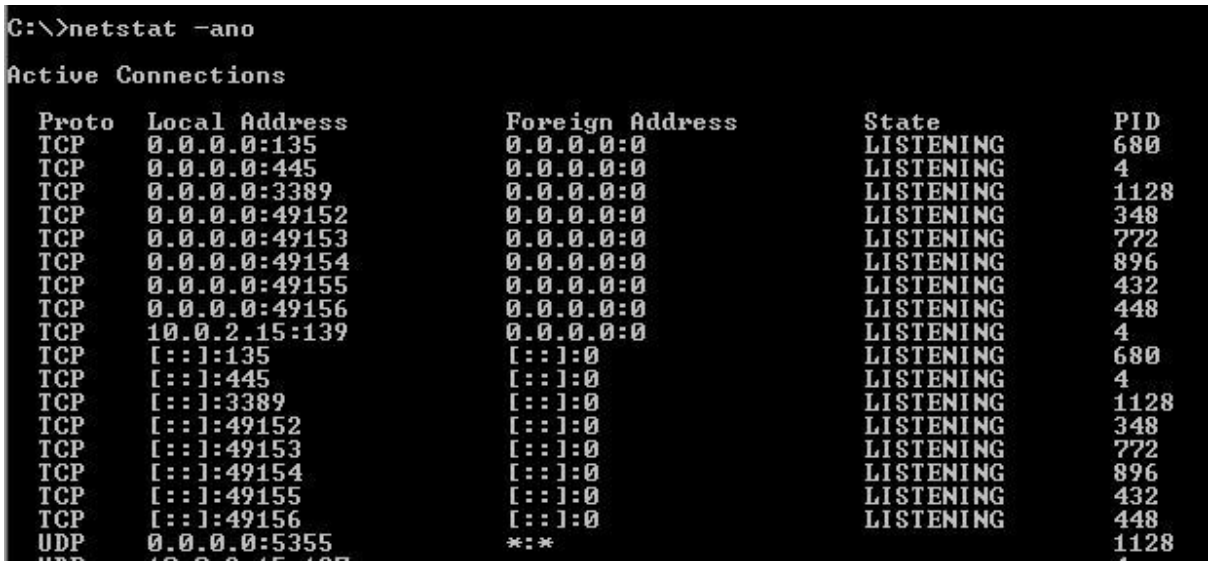
- Detect unauthorized services
- Identify malware communicating externally
- Validate firewall behavior

Example

If you see a process listening on **unexpected ports**, investigate immediately.



```
[vivek@nixcraft-nuc ~]$ sudo netstat -tulpn | grep LISTEN
tcp        0      0 127.0.0.1:53306         0.0.0.0:*               LISTEN      3371/AgentConnectix
tcp        0      0 127.0.0.1:44321         0.0.0.0:*               LISTEN      3784/pmcd
tcp        0      0 127.0.0.1:4330          0.0.0.0:*               LISTEN      9725/pmlogger
tcp        0      0 0.0.0.0:5355            0.0.0.0:*               LISTEN      1566/systemd-resolv
tcp        0      0 10.205.77.1:53          0.0.0.0:*               LISTEN      2416/dnsmasq
tcp        0      0 192.168.122.1:53        0.0.0.0:*               LISTEN      2081/dnsmasq
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      1566/systemd-resolv
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      1823/sshd
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      1821/cupsd
tcp6       0      0 ::1:44321               :::*                    LISTEN      3784/pmcd
tcp6       0      0 ::1:4330                :::*                    LISTEN      9725/pmlogger
tcp6       0      0 :::5355                 :::*                    LISTEN      1566/systemd-resolv
tcp6       0      0 fd42:400:b94d:ad98:::53 :::*                    LISTEN      2416/dnsmasq
tcp6       0      0 fe80::e400:44ff:feb7:53 :::*                    LISTEN      2416/dnsmasq
tcp6       0      0 :::22                   :::*                    LISTEN      1823/sshd
tcp6       0      0 ::1:631                 :::*                    LISTEN      1821/cupsd
[vivek@nixcraft-nuc ~]$
```



```
C:\>netstat -ano

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       680
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:3389           0.0.0.0:0              LISTENING       1128
  TCP    0.0.0.0:49152          0.0.0.0:0              LISTENING       348
  TCP    0.0.0.0:49153          0.0.0.0:0              LISTENING       772
  TCP    0.0.0.0:49154          0.0.0.0:0              LISTENING       896
  TCP    0.0.0.0:49155          0.0.0.0:0              LISTENING       432
  TCP    0.0.0.0:49156          0.0.0.0:0              LISTENING       448
  TCP    10.0.2.15:139          0.0.0.0:0              LISTENING       4
  TCP    [::]:135               [::]:0                 LISTENING       680
  TCP    [::]:445               [::]:0                 LISTENING       4
  TCP    [::]:3389              [::]:0                 LISTENING       1128
  TCP    [::]:49152             [::]:0                 LISTENING       348
  TCP    [::]:49153             [::]:0                 LISTENING       772
  TCP    [::]:49154             [::]:0                 LISTENING       896
  TCP    [::]:49155             [::]:0                 LISTENING       432
  TCP    [::]:49156             [::]:0                 LISTENING       448
  UDP    0.0.0.0:5355           *:*                                    1128
```

Outcome: Operational Readiness

After completing this task, a cybersecurity analyst should be able to:

- ✅ Navigate Linux confidently
- ✅ Investigate logs efficiently
- ✅ Detect suspicious files & processes
- ✅ Monitor real-time attacks
- ✅ Understand permission-based security
- ✅ Identify network anomalies
