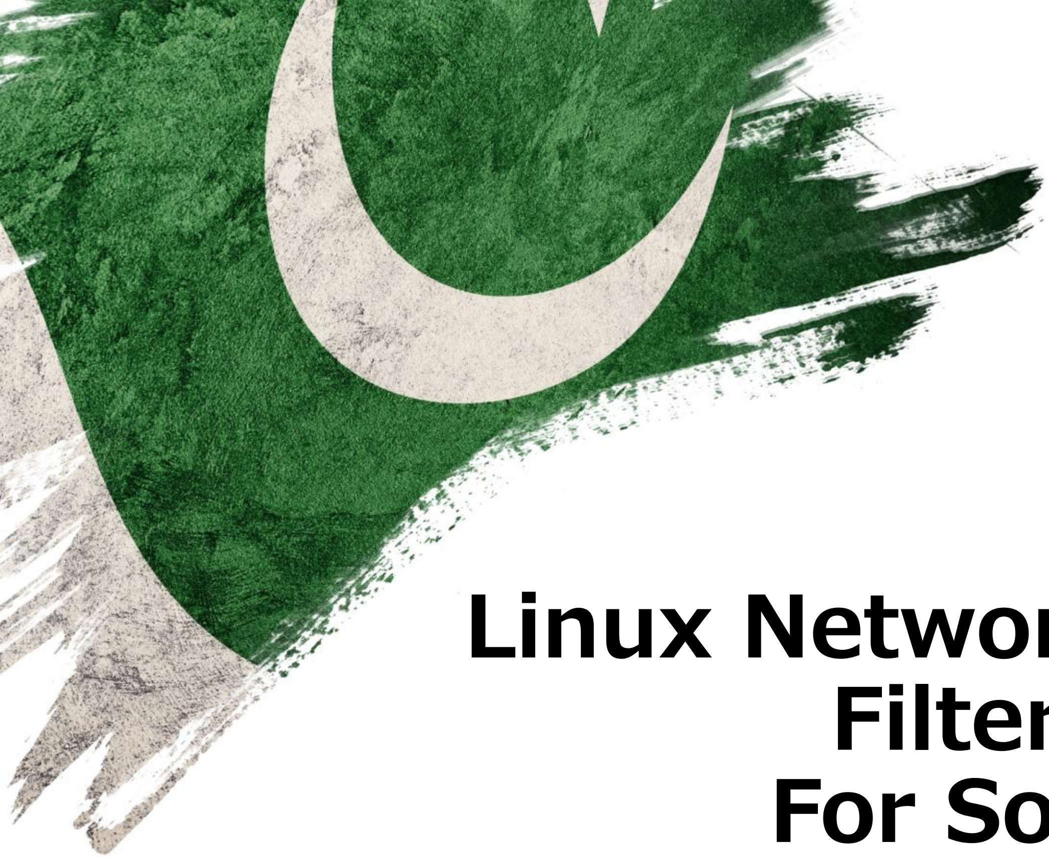

Linux Networking, Compression and Filtering Commands For Software Engineers

Class 15

10/5/2025

Acknowledgement

The series of the IT & Japanese language course is Supported by AOTS and OEC.

Ministry of Economy, Trade and Industry

Overseas Employment Corporation

What you have Learnt Last Week

We were focused on following points.

- Usage of control and loop flow statement
- Performing Linear Algebra in Numpy
- Why Requirement Analysis is so important in the process?
- Machine Learning algorithms
- Software development Life cycle
- Importance of Security compliance
- Basic Linux Commands.

What you will Learn Today

We will focus on following points.

- 1. Understanding the Linux file system hierarchy and user roles
- 2. Step-by-step guide on creating users and groups
- 3. Changing File Permissions and Ownership
- 4. Basic and System level Linux commands
- Quiz
- Q&A Session

Practice of Basic Networking Commands

Essential tools for network diagnostics and monitoring

- Networking is critical for troubleshooting in Linux servers.

Common commands:

- ping, traceroute, mtr
- netstat, ss
- telnet, nc
- ifconfig, ip addr
- nslookup, dig
- wget, curl
- iftop, nload

Using ping, traceroute, and mtr

Diagnosing connectivity and path issues

- **ping** — check reachability

```
ping google.com
```

- **traceroute** — map network path

```
traceroute google.com
```

- **mtr** — dynamic traceroute + ping

```
mtr google.com
```

Checking Active Connections (netstat / ss)

Inspecting listening ports and active sessions

netstat — older tool

- netstat -tulnp

ss — faster and detailed

- ss -tunap

Industry Use: Finding open ports (like 22/SSH) on production servers.

Testing Open Ports (telnet, nc)

Verifying service availability on remote servers

telnet — basic connection test

- telnet example.com 443

nc (netcat) — powerful scanner

- nc -zv example.com 22

Industry Use: Check whether ports like HTTPS or SSH are reachable.

Displaying IP Configurations (ifconfig, ip addr)

Viewing and managing network interfaces

ifconfig — traditional tool

- ifconfig eth0

ip addr — modern replacement

- ip addr show eth0

Industry Use: Checking public or private IP addresses on EC2 instances.

Checking DNS Resolution (nslookup, dig)

Resolving domain names to IP addresses

nslookup — simple query

- nslookup google.com

dig — detailed DNS info

- dig google.com

Industry Use: Troubleshoot DNS issues when websites are inaccessible.

Downloading Files

Retrieving data and interacting with APIs

wget — simple file download

- wget https://example.com/file.zip

curl — interact with APIs and download

- curl https://api.example.com/data

Industry Use: Automated scripts for downloading backups or pulling updates.

Monitoring Real-Time Traffic (iftop, nload)

Observing live bandwidth usage

iftop — connection-wise traffic display

- sudo iftop -i eth0

nload — graphical bandwidth view

- sudo nload

Industry Use: Identify sudden spikes in bandwidth due to DDoS or heavy backups.

Creating Compressed Archives (.tar.gz / .tar.bz2)

Combine and compress multiple files efficiently

Compress with gzip:

```
tar -czvf archive.tar.gz folder/
```

Compress with bzip2:

```
tar -cjvf archive.tar.bz2 folder/
```

- -c: create archive
- -z or -j: gzip or bzip2
- -v: verbose
- -f: file name

Example: Backing up /var/log into logs_backup.tar.gz

Extracting and Viewing Archives

Unpack or inspect contents without full extraction

Extract .tar.gz:

```
tar -xzvf archive.tar.gz
```

Extract .tar.bz2:

```
tar -xjvf archive.tar.bz2
```

List contents without extracting:

```
tar -tf archive.tar.gz
```

Decompression Using gunzip and bunzip2

Unpack individual compressed files

Decompress .gz:

```
gunzip file.gz
```

Decompress .bz2:

```
bunzip2 file.bz2
```

Note: This works only on individual files, not .tar.gz bundles.

Best Practices & Compression Tips

Efficient use of compression tools in real environments

- Use gzip for speed, bzip2 for better compression
- Automate log compression with cron:

```
tar -czf logs_$(date +%F).tar.gz /var/log
```

- Always verify archives before deleting original data
- Compress entire directories recursively:

```
tar -czf backup.tar.gz my_directory/
```

Introduction to Text Filtering with grep and awk

Efficiently search and process text from files and outputs

- Used to **search**, **extract**, and **manipulate** data from files/commands

Tools:

- **grep:** Searches for patterns
- **awk:** Extracts & formats structured text

Use cases:

- Parsing logs
- Extracting IPs
- Analyzing CSV/data files

Searching with grep (Basic & Regex)

Find matching patterns line by line

- **Simple pattern match:**

```
grep "error" syslog.txt
```

- **Regex support:**

```
grep -E "fail|denied" auth.log
```

- **Case-insensitive search:**

```
grep -i "warning" log.txt
```

- **Highlight matches:**

```
grep --color "ssh" secure.log
```

Using grep in Pipelines

Filter command output in real time

- **Filter system users:**

```
cat /etc/passwd | grep "/bin/bash"
```

- **Search listening ports:**

```
ss -tuln | grep "LISTEN"
```

Combine with tail, ps, journalctl, etc.

Basic awk for Column Extraction

Process structured files and outputs

- **Print 1st column (e.g., usernames):**

```
awk '{print $1}' users.txt
```

- **Display specific fields in logs:**

```
cat auth.log | awk '{print $1, $3}'
```

- **By default, awk separates by spaces; -F can set custom delimiters:**

```
awk -F ':' '{print $1, $3}' /etc/passwd
```

Conditional awk & Combining with grep

Apply conditions to filter data precisely

- **Only lines with UID > 1000:**

```
awk -F ':' '$3 > 1000 {print $1, $3}' /etc/passwd
```

- **Combine grep and awk:**

```
grep "sshd" auth.log | awk '{print $1, $5}'
```

Example: Show SSH login attempts with IPs

Practical Examples

Real-world filtering tasks

- **Parse syslog for reboots:**

```
grep "reboot" /var/log/syslog
```

- **Extract IP addresses:**

```
grep -oE "¥b([0-9]{1,3}¥.){3}[0-9]{1,3}¥b" access.log
```

- **Summarize memory usage:**

```
free -m | awk '/Mem/ {print "Used: " $3 "MB of " $2 "MB"}'
```

Task 1

Search for Keywords

Search for the word "error" in a log file and highlight it.

Step 1: Create a sample log file(syslog.txt)

```
cat <<EOF > syslog.txt
Jan 30 10:23:01 server systemd: Started Session 1 of user root.
Jan 30 10:23:15 server sshd[1354]: error: PAM: Authentication
failure for root from 192.168.1.5
Jan 30 10:23:20 server sshd[1354]: Accepted password for root
from 192.168.1.5 port 22 ssh2
EOF
```

Step 2: Search and highlight the keyword "error"

Task 2

Use Regex with grep

Use regex to find either the word "fail" or "denied" in a log file.

Step 1: Create a sample auth.log file

```
cat <<EOF > auth.log
Jan 30 11:12:01 server sshd[2001]: Failed password for invalid user test from 10.0.0.1 port 54720 ssh2
Jan 30 11:12:03 server sshd[2001]: Connection closed by 10.0.0.1 port 54720 [preauth]
Jan 30 11:12:05 server sshd[2002]: Access denied for user admin from 10.0.0.2
EOF
```

Step 2: Use regex to match both "fail" and "denied"

Quiz Section

Quiz

Everyone student should click on submit button before time ends otherwise MCQs will not be submitted

[Guidelines of MCQs]

1. There are 20 MCQs
2. Time duration will be 10 minutes
3. This link will be share on 12:25pm (Pakistan time)
4. MCQs will start from 12:30pm (Pakistan time)
5. This is exact time and this will not change
6. Everyone student should click on submit button otherwise MCQs will not be submitted after time will finish
7. Every student should submit Github profile and LinkedIn post link for every class. It include in your performance

Assignment

Assignment should be submit before the next class

[Assignments Requirements]

1. Create a post of today's lecture and post on LinkedIn.

2. Make sure to tag @Plus W @Pak-Japan Centre and instructors LinkedIn profile

3. Upload your code of assignment and lecture on GitHub and share your GitHub profile in respective

   your region group WhatsApp group

4. If you have any query regarding assignment, please share on your region WhatsApp group.

5. Students who already done assignment, please support other students

Q&A Session

ありがとうございます。

Thank you.

شکریا

+W

For the World with Diverse Individualities