Advanced Programming

WorkShop: Linux Fundamentals

Instructor: Ali Najimi

Lecturer: Hossein Masihi

Department of Computer Engineering

Sharif University of Technology

Fall 2025

Table of Contents

What is Linux?

- Open-source operating system kernel
- Core of distributions like Ubuntu, Debian, Fedora
- Used in servers, DevOps, IoT, and embedded systems
- Key features: **stability**, **security**, **flexibility**

File System Structure

Directory	Description
`/`	Root directory
`/home`	User home folders
`/etc`	System configurations
`/var/log`	Log files
`/usr/bin`	Installed apps and binaries

- **Absolute paths** start with `/`
- **Relative paths** depend on current directory

Terminal Basics

Command	Description
`pwd`	Print current directory
`ls -l`	List files in detail
`cd`	Change directory
`mkdir` / `rmdir`	Create or remove directory
`cp`, `mv`, `rm`	Copy, move, remove files
`cat`, `nano`, `less`	View or edit files

Example Session

```
pwd                         # Show current directory path
ls -lah                     # List all files with size & details
cd /etc                     # Change directory to /etc
mkdir test && cd test       # Create folder 'test' and move into it
cp /var/log/syslog .        # Copy syslog file to current directory
mv syslog logs.txt          # Rename syslog to logs.txt
rm logs.txt                 # Remove the file
```

Permissions

- Access types: `r` (read), `w` (write), `x` (execute)
- Levels: user, group, others

```
ls -l                          # View permissions and owners
chmod +x script.sh              # Add execute permission
chmod 640 config.env            # Owner read/write, group read only
chown root:root /etc/app.conf   # Change owner and group to root
```

Users & Processes

```
whoami          # Show current user
adduser test    # Add a new user named 'test'
ps aux          # Show all running processes
kill -9 1234    # Force kill process with PID 1234
```

View running tasks interactively:

```
top             # Display real-time process list
htop            # Better visual process monitor (install separately)
```

Package Management

Distribution	Tool	Example
Ubuntu/Debian	`apt`	`sudo apt install curl`
Fedora	`dnf`	`sudo dnf update`
Arch	`pacman`	`sudo pacman -S git`

```
sudo apt update && sudo apt upgrade -y   # Update and upgrade all packages
sudo apt install build-essential git     # Install packages
sudo apt remove nano                     # Remove a package
```

Bash & Shell Scripting

Automate tasks using Bash scripts

File extension: `.sh`

```
#!/bin/bash                    # Shebang line to use Bash
echo "Welcome to Bash!"        # Print a message to terminal
```

Save as `myscript.sh`, then:

```
chmod +x myscript.sh           # Make file executable
./myscript.sh                  # Run the script
```

Variables

```
#!/bin/bash
name="Hossein"                  # Define a variable
echo "Hello $name"              # Access variable value
```

Output:

```
Hello Hossein
```

Conditionals

```
#!/bin/bash
read -p "Enter a number: " num    # Ask for user input
if [ $num -gt 10 ]; then          # If number greater than 10
  echo "Greater than 10"
else                              # Otherwise
  echo "10 or less"
fi
```

Loops

```
#!/bin/bash
for i in 1 2 3 4 5                # Loop from 1 to 5
do
  echo "Count: $i"                # Print each iteration
done
```

or:

```
count=1                           # Initialize variable
while [ $count -le 3 ]            # While count ≤ 3
do
  echo "Loop $count"              # Print current count
  ((count++))                     # Increment count
done
```

Small Project: “Backup Logs”

```
#!/bin/bash
# Simple backup script
mkdir -p ~/backup                 # Create backup directory
cp /var/log/syslog ~/backup/      # Copy system log file
echo "Backup completed!"          # Print confirmation
```

Hands-on Practice

1. Create a file named `hello.sh`
2. Add a shebang line (`#!/bin/bash`)
3. Print your name and current date (`date`)
4. Make it executable and run it
5. Add a loop or if-condition

Summary & References

- Linux = stable, secure, flexible
- Know file system + terminal basics
- Manage users, permissions, packages
- Start writing small Bash scripts

Resources:

- linuxjourney.com
- linuxcommand.org

End of Presentation

Sharif University Workshop Series — Linux Fundamentals