Linux Fundamental

Presentated by RACKSYNC CO., LTD.

Course Objectives

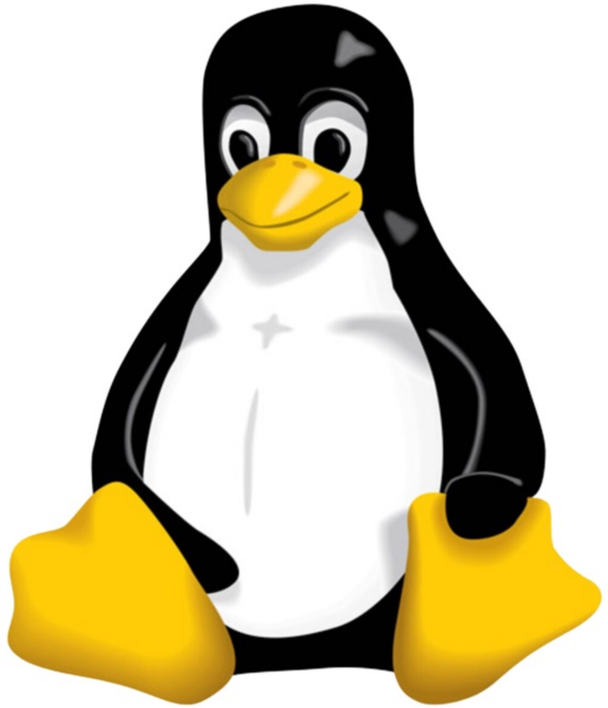
- Understand Linux structure and basic concepts
- Execute fundamental Linux commands
- Manage processes and troubleshoot common issues
- Real-world applications in Server & Cloud environments

Key Takeaways: This course bridges the gap between Windows and Linux.

Introduction to LINUX

What is Linux?

- Free / Non-Subscription
- Open-source operating system
- Based on UNIX principles
- Powers servers
- Cloud Computing
- Embedded systems
- Consumer Technology



Most Cloud Services

CLOUD SERVICES	aws		
Virtual Servers	Instances	VM Instances	VMs
Platform-as-a-Service	Elastic Beanstalk	App Engine	Cloud Services
Serverless Computing	Lambda	Cloud Functions	Azure Functions
Docker Management	ECS	Container Engine	Container Service
Kubernetes Management	EKS	Kubernetes Engine	Kubernetes Service
Object Storage	S3	Cloud Storage	Block Blob
Archive Storage	Glacier	Coldline	Archive Storage
File Storage	EFS	ZFS / Avere	Azure Files
Global Content Delivery	CloudFront	Cloud CDN	Delivery Network
Managed Data Warehouse	Redshift	Big Query	SQL Warehouse

image source: https://medium.com/@redslick84/aws-vs-azure-vs-gcp-4933f48aaae4

Understanding Linux Distributions

- **Popular Distros:** Ubuntu, CentOS, Debian
- **Why so many versions?** (Customization & Use-cases)

Linux Ranking: https://distrowatch.com

Command Line Interface (CLI)

- Offers unparalleled control and flexibility.
- Ideal for automation and advanced system tasks.
- Minimal resource overhead compared to graphical interfaces.
- Empowers developers and system administrators with direct command execution.

Linux Users

Root User:

- The superuser with full system privileges
- Can execute any command and modify all system files
- Use with caution due to high risk of system damage

Normal User:

- Has limited privileges for everyday tasks
- Provides better security by restricting access
- Uses sudo for tasks that require elevated permissions

Linux Kernel

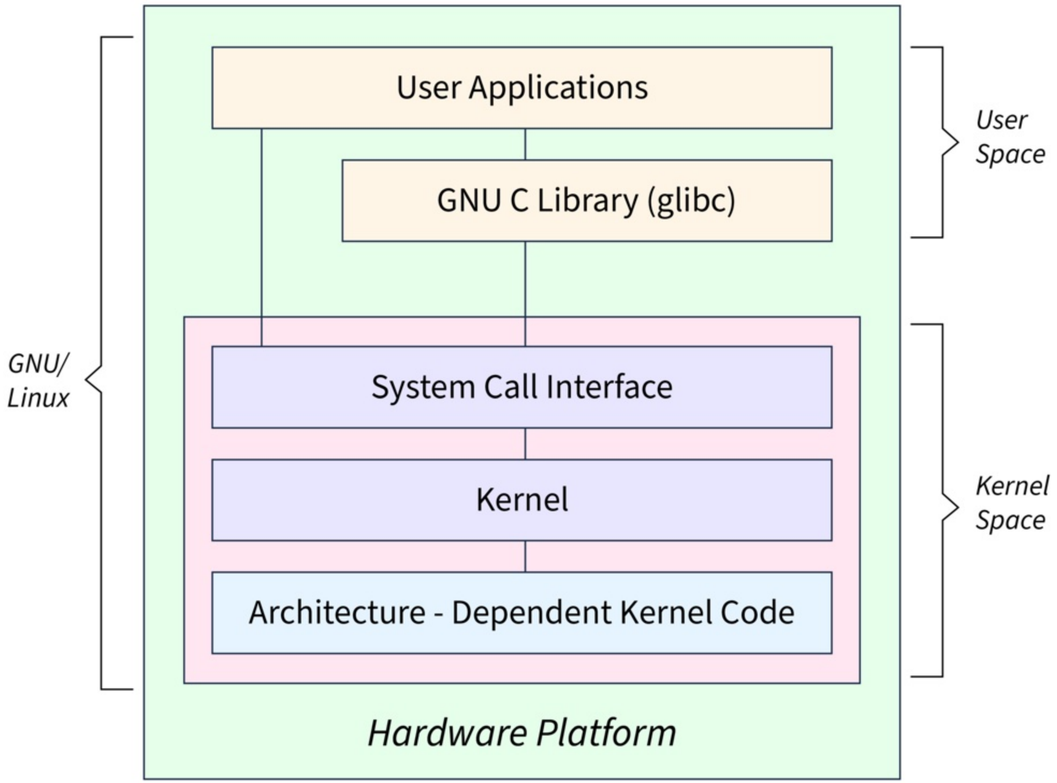


image source: https://www.scaler.com/topics/linux-kernel-architecture/

Setting Up Linux Environment

- **Virtual Machine** (hands-on labs)
- **Cloud Provider** (Production Grade Server)
- **WSL or Windows Subsystem for Linux** (Dev Environment)

Key Takeaways: Practice is key—experiment in a virtual environment.

Differences from Windows

- File System: C:\, D:\ vs /, /home, /etc
- Permissions: GUI-based vs rwx permissions
- System Management: Task Manager vs top/htop

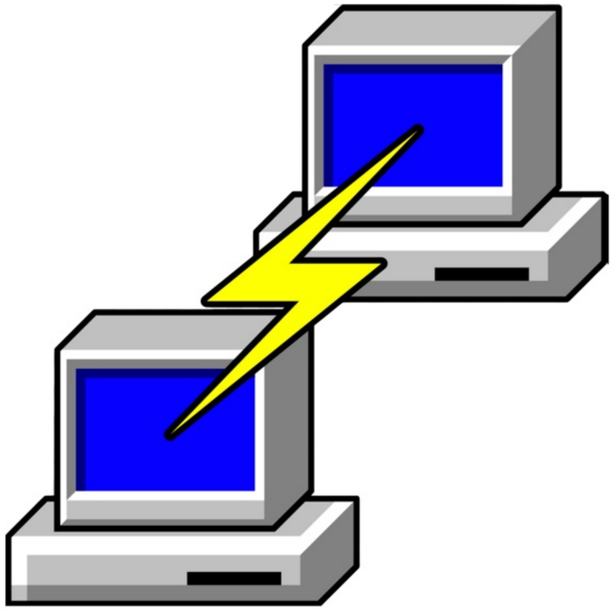
Key Takeaways: Linux is command-line-centric but powerful and flexible.

Tools for Windows Users

Windows Tools

- Windows Subsystem for Linux (WSL/WSL2)
- PuTTY (Remote Access)
- PuTTY Key Generator (puttygen)
- MobaXterm (Terminal emulator)

puTTY



https://www.putty.org

Dive into the LINUX system

Linux File System Hierarchy

- - /: Root directory
- - /home: User files
- - /etc: Configuration files
- - /var/log: System logs

Command Example: tree / (Visualize directory structure)

Linux File System Hierarchy

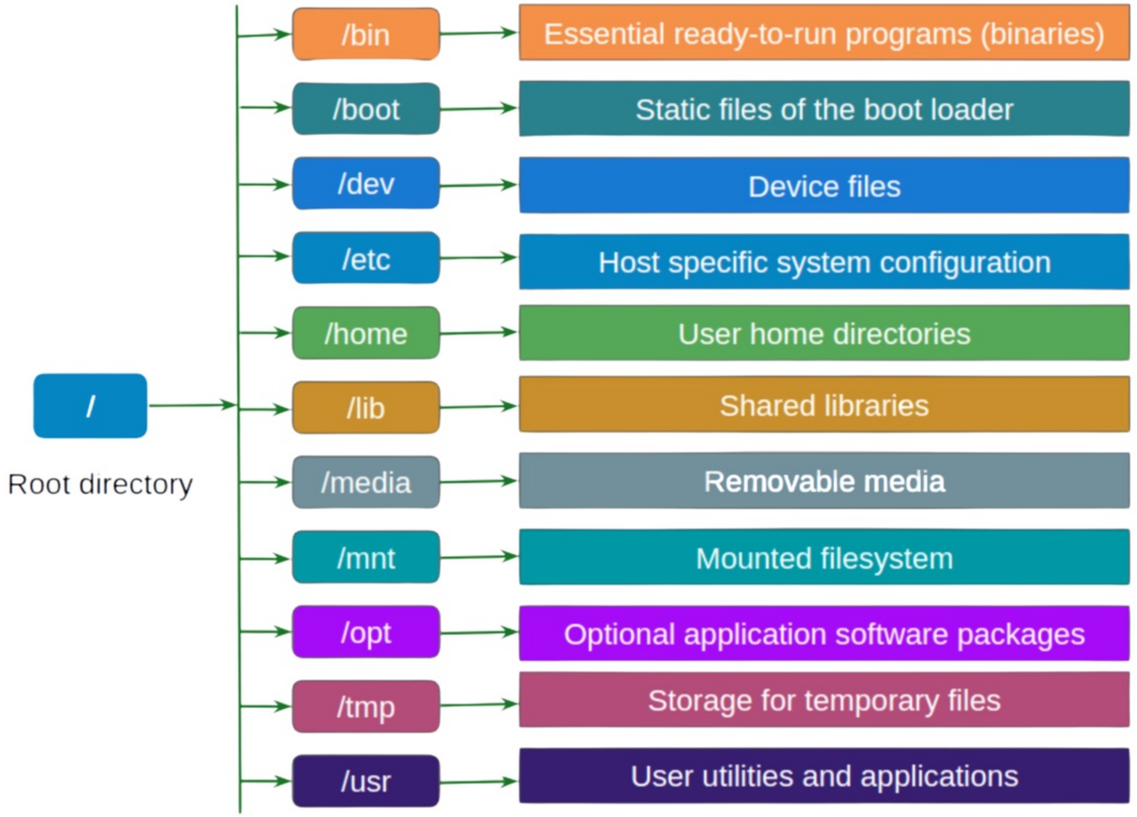


image source: https://www.enablegeek.com/tutorial/work-file-system-in-linux/

File Permissions & Ownership

- Read (r), Write (w), Execute (x)
- Changing permissions: chmod, chown
- Command Example:
- chmod 755 script.sh
- chown user:group file.txt

- **Key Takeaways:** Permissions control security at the file level.

File Permissions Explain

Binary	Octal	String Representation	Permissions
000	0 (0+0+0)	---	No Permission
001	1 (0+0+1)	--x	Execute
010	2 (0+2+0)	-w-	Write
011	3 (0+2+1)	-wx	Write + Execute
100	4 (4+0+0)	r--	Read
101	5 (4+0+1)	r-x	Read + Execute
110	6 (4+2+0)	rw-	Read + Write
111	7 (4+2+1)	rwx	Read + Write + Execute

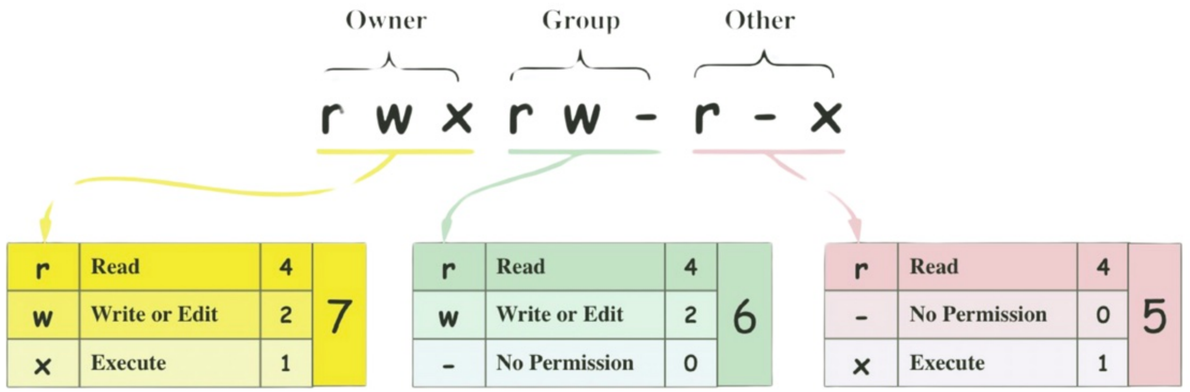


image source: https://x.com/sahnlam/status/1775393641965687034/photo/1

Basic Linux Commands

Navigation: pwd, cd, ls

File Management: cp, mv, rm

Viewing Files: cat, less, head, tail

Example:

- ls -l /home
- cd /etc
- cat /var/log/syslog

User & Group Management

Add User: adduser newuser

Modify Groups: usermod -aG sudo newuser

Key Takeaways: Linux is multi-user by design—permissions matter!

Remote Access with SSH

Secure login: ssh user@host

File transfer: scp file.txt user@host:/path

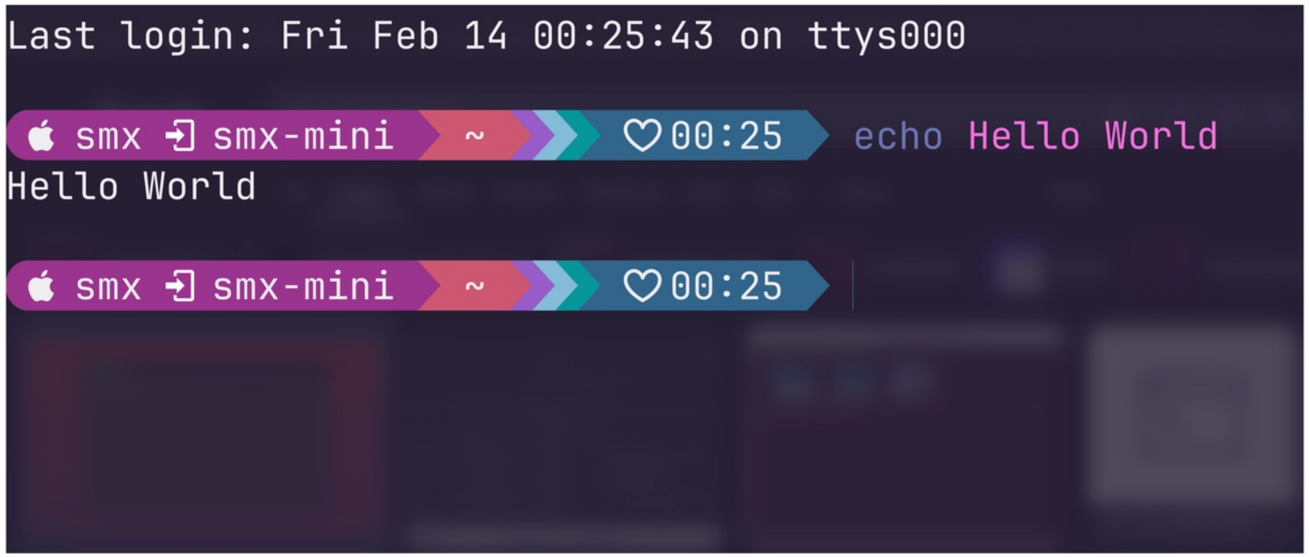
Exercise : Connect to a remote server with puTTY and edit a file with Nano.

Linux Shell

- A command-line interface for interacting with OS
- Enables users to execute commands, run scripts, and automate tasks
- Supports various shell environments like Bash, Zsh, and more

Linux Prompt

- The text-based interface where users enter commands
- Displays useful information (e.g., username, current directory)
- Highly customizable to enhance user experience



Hello World

- **Use the command:** echo "Hello World"
- Displays the text "Hello World" in the terminal.
- A basic example to test your shell environment.

sudo

- Execute commands with elevated privileges.
- Provides temporary administrative rights for specific tasks.
- Enhances security by avoiding constant root login.
- Configurable through the /etc/sudoers file.

Editors

- **Nano:** User-friendly and ideal for beginners.
- **Vim:** Powerful, extensible, and highly configurable.
- **Emacs:** Feature-rich and customizable for advanced users.
- **Graphical Options:** Gedit, Kate, and more for GUI-based editing.

First Edit /etc/apt/sources.list

- Contains repository sources for APT.
- Essential for managing package installation and updates.
- Edit using a text editor with root privileges (e.g., sudo nano /etc/apt/sources.list).
- Run sudo apt update after changes to refresh package lists.

Package Management

APT (Debian & Ubuntu): sudo apt install apache2

YUM (CentOS): sudo yum install http

Key Takeaways: Package managers simplify software installation and updates.

Repositories / Dependencies

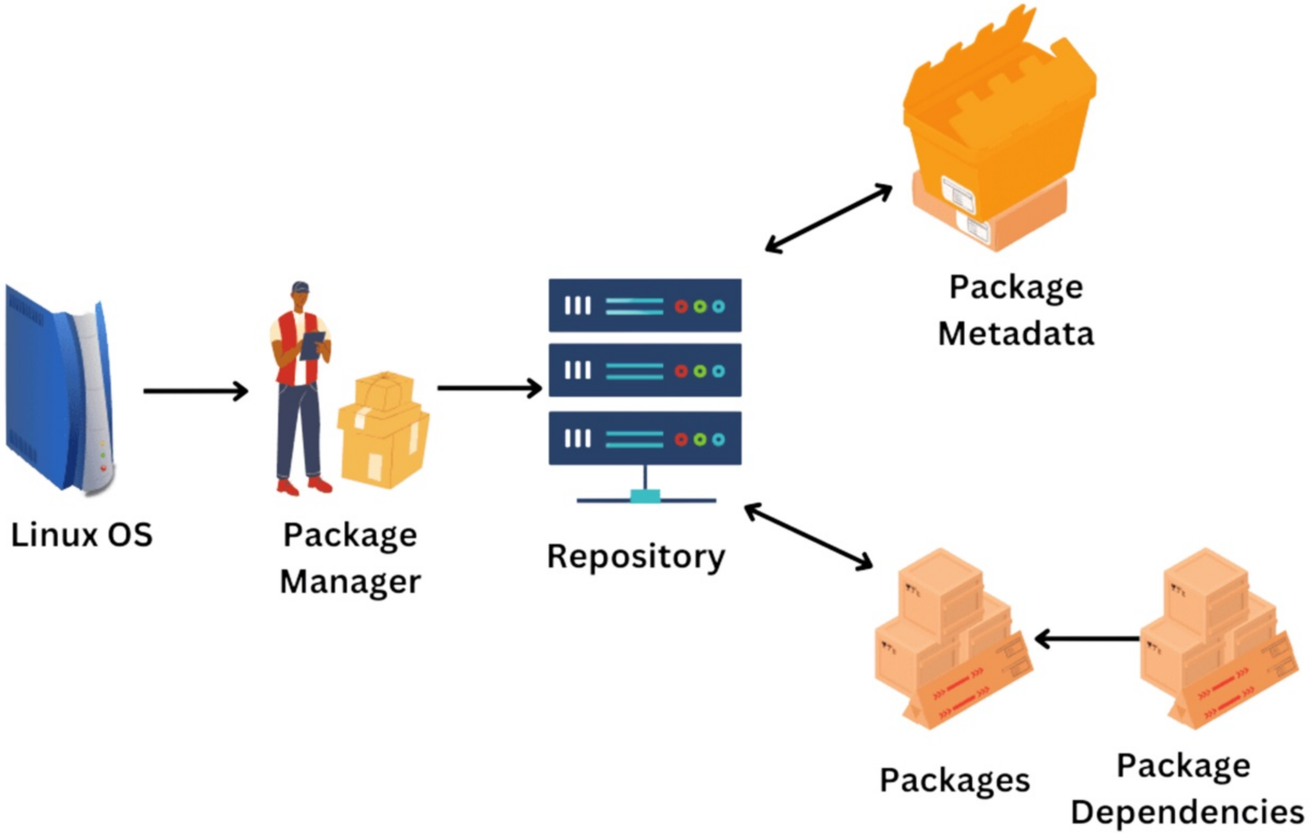


image source: https://www.enablegeek.com/tutorial/work-file-system-in-linux/

Networking Basics

Check IP: ip a

Test connection: ping google.com

View ports: ss -tuln

Exercise : Ping a server, verify open ports.

Process Management

View Processes: ps aux, top, htop

Kill Process: kill PID

Terminate: CTRL+Z

Key Takeaways: Every service runs as a process—monitor them regularly.

System Control (systemctl)

- systemctl start <service_name>
- systemctl stop <service_name>
- systemctl restart <service_name>
- systemctl status <service_name)

Basic Shell Scripting

Simple script example:

```
#!/bin/bash
echo "Hello, Linux World!"
```

Make executable: chmod +x script.sh

Exercise Prompt: Write a script to automate file backups.

Diagnostics & Help Command

Manual & Help Command

View Manual: man <command>

Command Example: man ls

Key Takeaways: Man pages are an essential reference for Linux commands.

Log & Troubleshooting

View Logs: tail -f /var/log/syslog

Diagnose Issues: dmesg | less

Key Takeaways: Logs are your best friend for troubleshooting.

Final Recap

- Core Linux Concepts Reviewed
- Key Commands Highlighted
- Hands-on Practice Reinforced
- Keep exploring Linux—practice makes perfect!

Resources

Introduction to Linux Filesystem for Windows Users : https://www.redhat.com/en/blog/linux-filesystem-windows

Linux Tools for Windows Users : https://hackmd.io/@pmitev/Linux4WinUsers

Hostinger Tutorial: 60 Essential Linux Commands : https://www.hostinger.com/tutorials/linux-commands

Basic Linux Commands : https://www.redhat.com/en/blog/basic-linux-commands

Ultimate Guide to Linux for Windows Users : https://www.dedoimedo.com/computers/ultimate-linux-guide-for-windows-users.html

How to Learn Linux as a Windows User (5 Steps) : https://installtekz.com/how-to-learn-linux-windows-user-guide/

Top 50+ Essential Linux Commands You Must Know : https://www.digitalocean.com/community/tutorials/linux-commands

DistroWatch Linux Ranking: https://distrowatch.com

Command Cheat sheet: https://www.geeksforgeeks.org/linux-commands-cheat-sheet/

RACKSYNC CO., LTD.

RACKSYNC Co., LTD. specializes in automation and smart solutions of all scales. We are experts in designing, implementing, and monitoring sophisticated automation systems. Our team of specialists provides comprehensive consulting services and technical implementation for both residential and commercial projects. Beyond automation, we offer full-cycle Software as a Service (SaaS) development, helping businesses transform their operations through custom digital solutions. With our deep expertise in IoT, home automation, and enterprise systems, we deliver reliable and innovative solutions tailored to each client's unique requirements.

Address : 135/5 Moo.1, Khuntalae Sub-District, Muang District, Surat Thani, 84100

devops@racksync.com

https://racksync.com

FB : https://www.fb.com/racksync

Tel : 08 5880 8885