Linux Basics For Cybersecurity

A comprehensive guide

14th / June 2025

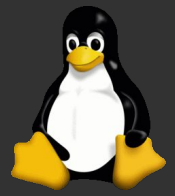
Presented by

Shivendra Chauhan

Shivendra0309@gmail.com

Content

Why Linux? The Philosophy



🔓 Open Source

Source code is free to audit, modify, and distribute

🧠 Unix Philosophy

"Do one thing and do it well" — modularity & pipelines

🛠️ Customizability

From embedded devices to servers to supercomputers

🧬 Evolution

Built by a global community, for real-world needs

Linus Torvalds — The G.O.A.T of Modern Computing

"Talk is cheap, show me the code."

```
From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: What would you like to see most in minix?
Summary: small poll for my new operating system
Message-ID:
Date: 25 Aug 91 20:57:08 GMT
Organization: University of Helsinki


Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and
professional like gnu) for 386(486) AT clones.  This has been brewing
since april, and is starting to get ready.  I'd like any feedback on
things people like/dislike in minix, as my OS resembles it somewhat
(same physical layout of the file-system (due to practical reasons)
among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work.
This implies that I'll get something practical within a few months, and
I'd like to know what features most people would want.  Any suggestions
are welcome, but I won't promise I'll implement them :-)

              Linus (torvalds@kruuna.helsinki.fi)

PS.  Yes - it's free of any minix code, and it has a multi-threaded fs.
It is NOT protable (uses 386 task switching etc), and it probably never
will support anything other than AT-harddisks, as that's all I have :-(.

 Judging from the post, 0.01 wasn't actually out yet, but it's close.
 I'd guess the first version went out in the middle of September -91. I
 got some responses to this (most by mail, which I haven't saved), and
 I even got a few mails asking to be beta-testers for linux. After that
 just a few general answers to quesions on the net:
```

Linus Benedict Torvalds, a Finnish software engineer, created the Linux kernel in 1991 as a personal project — just to improve his UNIX-like operating system experience.

What started as a humble post on an internet forum exploded into a global movement. Today, Linux powers:

96% of the world's top 1 million servers

100% of supercomputers

Android smartphones, firewalls, routers, and every hacker's toolkit.

Linux Architecture Explained

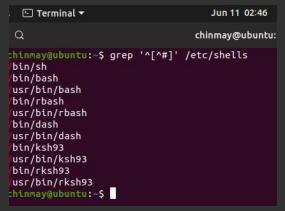
(Car Analogy)

Packages
=> **Fuel**

Apps & dependencies

Shell (Bash)
=> **Steering Wheel**

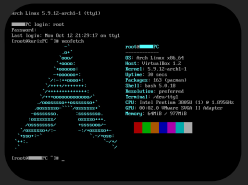
User control interface

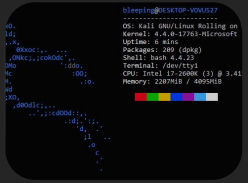


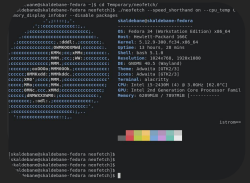
Kernel
=> **Engine**

Core, controls hardware, process mgmt

Distros
=> **Car brands**

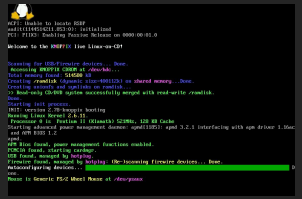






Init System
=> **Ignition**

Boot sequence (e.g., systemd)



📁 Linux Filesystem Hierarchy

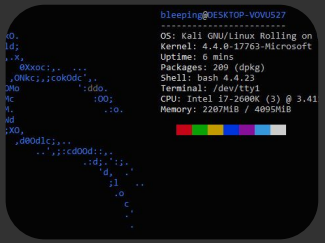
```
/
├── bin        # Essential user commands (e.g., ls, cp, mv)
├── etc        # Configuration files
│   ├── passwd     # User account info
│   ├── shadow     # Encrypted user passwords
│   └── network/   # Network configuration
├── home       # User directories
│   └── shivendra/ # Your personal files and configs
├── lib        # Shared libraries for binaries in /bin and /sbin
│   └── systemd/   # Libraries used by systemd init system
├── root       # Home directory for root user (superuser)
├── usr        # User-installed applications and files
│   ├── bin/       # Non-essential user commands
│   ├── lib/       # Libraries for /usr/bin
│   └── share/     # Architecture-independent data (man pages, docs)
├── var        # Variable data files
│   ├── log/       # System logs (e.g., auth.log, syslog)
│   └── cache/     # Cached application data
```

Virtual Machines

Benefit	Explanation
💣 Sandbox	Test malware, exploits safely
🔁 Snapshots	Revert any damage
🧪 Experimentation	Break it without fear
📦 Portability	Easily move/test images

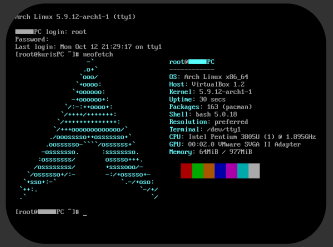
Host OS : For eg Windows

Hypervisor

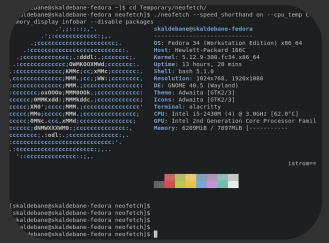


Kali Linux
8gb Ram, 80gb SSD

Debian Linux
8 gb Ram, 50 gb SSD



Arch Linux
1gb Ram, 20gb SSD



Fedora Linux
16gb Ram, 100gb SSD

Arch Linux Installation (With Script)

```
Set/Modify the below options

Use ESC to skip


> Select Archinstall language          SET: English
  Select keyboard layout               SET: us
  Select mirror region                 SET: []
  Select harddrives
  Select bootloader                    SET: systemd-bootctl
  Use swap                             SET: True
  Specify hostname                     SET: archlinux
  Set root password                    SET: None
  Specify superuser account
  Specify user account                 SET: []
  Specify profile                      SET: None
  Select audio                         SET: None
  Select kernels                       SET: ['linux']
  Additional packages to install       SET: []
  Configure network                    SET: Not configured, unavailable unless setup manually
  Select timezone                      SET: UTC
  Set automatic time sync (NTP)        SET: True
  Additional repositories to enable    SET: []

  Save configuration
  Install (2 config(s) missing)
  Abort
(Press "/" to search)
```

Linux Commands: Basics

Type	Command	Use
Files	`ls cd cp mv rm touch nano`	Manage files
Processes	`ps top kill htop`	Process mgmt
Permissions	`chmod chown umask`	File access
Networking	`ping netstat ss ifconfig ip a`	Basic networking
Package	`apt pacman yum dnf`	Install/remove software
System	`df du uptime whoami uname -a`	Info gathering

Linux Commands: Hacking-Oriented Commands

Tool	Command	Purpose
Nmap	`nmap -A scanme.nmap.org`	Recon on public test server
Netcat	`nc -lvnp 4444`	Open listener for reverse shell
Tcpdump	`sudo tcpdump -i any -c 10`	Capture 10 packets on all IFs
Whois	`whois google.com`	Whois lookup (domain info)
Curl	`curl -I https://nmap.org`	View server response headers
Find	`find / -name id_rsa 2>/dev/null`	Search for private keys
Wget	`wget http://testphp.vulnweb.com`	Download a test site (for safe scans)
Traceroute	`traceroute google.com`	Track path to a domain
Host	`host github.com`	DNS lookup

🖥️ Shell Scripts: The Power of Automation

🔹 What is a Shell Script?

A **Shell Script** is a plain text file containing a sequence of Linux commands. Instead of typing commands one by one, scripts **automate** tasks — from system monitoring to hacking automation.

🚀 Why Use Scripts?

- ✅ Automate repetitive tasks
- 🔁 Schedule jobs (e.g., backups, updates)
- 🛠️ Chain multiple commands together
- ⚙️ Build your own Linux tools

Element	Example	Purpose
Comments	`# This is a comment`	Explains code
Variables	`NAME="Shivendra"`	Store data
Conditionals	`if [ $a -gt $b ]; then ...`	Logic checks
Loops	`for i in 1 2 3; do ... done`	Repetition
Functions	`myFunc() { echo "Run"; }`	Modular code
Execution	`./script.sh` `bash script.sh`	Run the script

Note:Use `chmod +x script.sh` to make it executable.

Automate Backups
Personal Use Case

```
#!/bin/bash
# 📦 Backup important files to a folder with date stamp

SRC="$HOME/Documents"
DEST="$HOME/Backups"
DATE=$(date +%F)
mkdir -p "$DEST"
tar -czf "$DEST/backup-$DATE.tar.gz" "$SRC"

echo "✅ Backup of $SRC completed at $DEST/backup-$DATE.tar.gz"
```

Recon Script
Professional Use Case

```
#!/bin/bash
# 🌐 Basic Recon Script for google.com

TARGET="google.com"
echo "🔍 Running OSINT Recon on $TARGET"
echo "--------------------------------"

whois $TARGET | grep -Ei 'OrgName|Country|Registrar|Address'
nslookup $TARGET
dig $TARGET ANY +noall +answer
host $TARGET
curl -I https://$TARGET

echo "✅ Recon Complete"
```

Hacker-ish Script
Hackers Use Case

```
#!/bin/bash
# ⚠️ Simple Port Scanner (educational purpose only)

TARGET="scanme.nmap.org"
echo "🔎 Scanning open ports on $TARGET..."

for PORT in {20..25}; do
  (echo > /dev/tcp/$TARGET/$PORT) >/dev/null 2>&1 && \
  echo "✅ Port $PORT is OPEN" || \
  echo "❌ Port $PORT is CLOSED"
done
```

Resources

*All Resources are given in an attached package folder

Thank you

Ready for what's next?

Let's talk

Shivendra Chauhan
Shivendra0309@gmail