Linux & VM Quick Handbook

Comprehensive Guide to Linux Administration, Virtualization & DevOps Tools

Author
Chinmaya Dalai

Connect With Me

LinkedIn: linkedin.com/in/chinmayadalai

GitHub: github.com/chikuchinmaya

December 2025

Contents

1 Prerequisites for Windows Tools

1.1 Installing Chocolatey

Follow the official installation guide at `https://chocolatey.org/docs/installation`.

1.2 Essential Tool Installation

```
choco install virtualbox --version=7.1.4 -y
choco install vagrant --version=2.4.3 -y
choco install git -y
choco install corretto17jdk -y
choco install maven -y
choco install awscli -y
choco install intellijidea-community -y
choco install vscode -y
choco install sublimetext3 -y
```

2 Virtual Machine Setup

2.1 Manual VM Setup (VirtualBox)

1. Enable virtualization (VT-x/AMD-V) in BIOS/UEFI and boot Windows.
2. Open VirtualBox → Click "New" → Provide name → Select OS type.
3. Assign CPU/RAM and disk size as required.
4. Go to Settings → Storage → Add OS ISO file (CentOS Stream 9 or Ubuntu).
5. Go to Settings → Network → Adapter 2 → Enable → Set to "Bridged Adapter".
6. Select your network adapter (Wi-Fi or Ethernet).
7. Settings → System → Pointing Device → Change to "USB Tablet".
8. Start VM, complete OS installation, remove ISO, and restart.

Note: For Ubuntu, download ISO from `https://releases.ubuntu.com/jammy/`

2.2 VM Setup with Vagrant (Automated)

Vagrant simplifies VM creation by reading a `Vagrantfile` and managing VM boxes automatically.

2.2.1 Quick Start

```
# Create directories for VMs
mkdir -p vagrant-vms/centos
mkdir -p vagrant-vms/ubuntu

# Navigate to centos directory
cd vagrant-vms/centos

# Initialize vagrant with CentOS Stream 9
vagrant init eurolinux-vagrant/centos-stream-9

# Or for Ubuntu 22.04
vagrant init ubuntu/jammy64
```

Find available boxes at `https://portal.cloud.hashicorp.com/vagrant/discover`

2.2.2 Vagrant Commands

Common Vagrant Commands

```
vagrant up              # Start and provision VM
vagrant box list        # List downloaded boxes
vagrant status          # Check VM status
vagrant ssh             # SSH into VM
vagrant halt            # Stop VM
vagrant reload          # Reboot VM with config changes
vagrant destroy         # Delete VM (keeps box)
vagrant global-status   # Show all Vagrant environments
```

2.2.3 Network Configuration

Inside Linux VM:

```
ip addr show    # Show network details
ip a            # Shortened version
```

On Windows host:

```
ipconfig        # Show network details
```

3 Linux Fundamentals

3.1 Important Linux Directories

Directory	Purpose
`/root, /home/username`	Home directories
`/bin, /usr/bin, /usr/local/bin`	User executables
`/media, /mnt`	Mount points
`/etc`	Configuration files
`/tmp`	Temporary files
`/boot`	Kernel and bootloader
`/var, /srv`	Server/state data
`/proc, /sys`	System information
`/lib, /usr/lib, /usr/local/lib`	Shared libraries

3.2 Popular Linux Distributions

3.2.1 Desktop Linux

Ubuntu, Linux Mint, Arch Linux, Fedora, openSUSE, Debian

3.2.2 Server Linux

Red Hat Enterprise Linux, Ubuntu Server, CentOS/AlmaLinux/Rocky, SUSE Linux Enterprise

3.3 Basic Linux Commands

Essential Commands

```
pwd                # Print working directory
mkdir              # Create directory
cd                 # Change directory
ls                 # List files/directories
touch              # Create empty file
cat                # Show file contents
whoami             # Show current user
uptime             # System uptime
free -m            # Memory usage in MB
cat /etc/os-release  # OS details
ip addr show       # Network details
ip a               # Shorthand
sudo -i            # Become root
sudo su            # Switch to root
```

3.3.1 Path Types

- **Absolute Path:** Starts with `/`, complete path from root (e.g., `/home/user/documents`)
- **Relative Path:** Path from current directory (e.g., `./documents` or `../`)

4 Text Editing with Vim

4.1 Installation

```
sudo yum install vim -y
```

4.2 Opening a File

```
vim filename.txt
```

4.3 Vim Modes

Vim operates in three modes:

1. **Command (Normal) Mode:** Default mode for navigation and commands.
2. **Insert Mode:** For editing text. Enter with `i`.
3. **Extended (Colon) Mode:** For saving, quitting, searching. Enter with :

4.4 Essential Commands

4.4.1 Command Mode Navigation

```
gg              # Go to start of file
G               # Go to end of file
w, b            # Move forward/backward by word
5w, 3b          # Move n words
u               # Undo last change
Ctrl+r          # Redo
yy              # Copy line
5yy             # Copy 5 lines
p               # Paste below cursor
P               # Paste above cursor
dw              # Delete word
x               # Delete character
dd              # Delete line
5dd             # Delete 5 lines
/word           # Search for word
```

4.4.2 Extended Mode (Colon Commands)

```
:w              # Save
:q              # Quit
:wq or :x       # Save and quit
:w!             # Force save
:q!             # Quit without saving
```

```
:20              # Go to line 20
:set nu          # Show line numbers
:set nonu        # Hide line numbers
:%s/old/new/     # Replace first per line
:%s/old/new/g    # Replace all occurrences
```

5 Text Processing & Filtering

5.1 grep - Search Text

```
grep text file                    # Case-sensitive search
grep -i text file                 # Case-insensitive
grep -iR text *                   # Recursive search
grep -iv text file                # Lines without text
grep -n text file                 # Show line numbers
```

5.2 Other Useful Commands

```
less file              # View file page by page (q to quit)
more file              # Simpler version of less
head file              # First 10 lines
head -20 file          # First 20 lines
tail file              # Last 10 lines
tail -f file           # Follow file (live logs)
wc -l file             # Count lines
```

5.3 cut and awk

Extract specific fields from structured text:

```
# Using cut with delimiter ':'
cut -d: -f1 /etc/passwd          # Show first field (usernames)

# Using awk
awk -F: '{ print \$1 }' /etc/passwd     # Same with awk
awk -F: '{ print \$1 , \$3 }' /etc/passwd  # First and third field
```

5.4 Search and Replace

5.4.1 Using Vim

```
:%s/old/new/          # Replace first per line
:%s/old/new/g         # Replace all in file
```

5.4.2 Using sed

```
sed 's/old/new/g' file                    # Show changes (no file
   mod)
sed -i 's/old/new/g' file                 # Edit file in-place
sed -i 's/old/new/g' *.conf               # Multiple files
```

5.5 Piping & Redirection

5.5.1 Output Redirection

```
command > file              # Overwrite file with stdout
command >> file             # Append stdout
command 2>> file            # Append stderr
command &>> file            # Append both stdout and stderr
command > /dev/null         # Discard output
```

5.5.2 Pipes

```
ls | wc -l                      # Count directory entries
ls | grep host                  # Filter entries with "host"
tail /var/log/messages | grep -i error   # Find errors in logs
ps aux | grep java              # Find Java processes
```

6 Finding Files

6.1 find Command

```
find /etc -name 'host*'          # Find files starting with "host"
find /etc -type f -name 'host*' # Files only (not directories)
find / -name '*.conf'            # Find all config files
find / -type d -name 'logs'      # Directories only
find / -size +100M               # Files larger than 100MB
```

6.2 locate Command

```
sudo yum install mlocate -y
sudo updatedb                  # Update database
locate filename                 # Fast search using database
```

Note: `locate` uses a pre-indexed database, so it's faster than `find` but may show outdated results if `updatedb` hasn't run recently.

7 User and Group Management

7.1 Understanding /etc/passwd

Example entry:

```
root:x:0:0:root:/root:/bin/bash
vagrant:x:1000:1000::/home/vagrant:/bin/bash
```

Format: `username:password_hash:UID:GID:full_name:home_dir:login_shell`

- `x` – Password stored in `/etc/shadow`
- `UID` – User ID (0 = root)
- `GID` – Group ID

7.2 User Management Commands

User Operations

```
useradd username                  # Add user
groupadd groupname                # Add group
usermod -aG groupname username    # Add user to group
passwd username                   # Set password
su - username                     # Switch user
id username                       # Show user IDs and groups
last                              # Login history
lsof -u username                  # Files opened by user
userdel username                  # Delete user (keep home)
userdel -r username               # Delete user with home dir
groupdel groupname                # Delete group
```

8 File Permissions

8.1 Understanding Permissions

```
ls -l
-rw-r--r--. 1 root root 2060 Dec  9 20:07 file.txt
drwxr-xr-x. 2 root root   31 Dec  9 20:15 directory
```

Symbol	Meaning
-	Regular file
d	Directory
l	Symbolic link
r (4)	Read permission
w (2)	Write permission
x (1)	Execute permission

Permission layout: `[file type][user][group][others]`

8.2 Changing Permissions (Symbolic)

```
chmod ugo+r file            # Add read to all
chmod o-w file              # Remove write from others
chmod u+x file              # Add execute to user
chmod -R 755 directory      # Recursive change
```

8.3 Changing Permissions (Numeric)

Permission values: read=4, write=2, execute=1

```
chmod 644 file              # -rw-r--r--
chmod 755 file              # -rwxr-xr-x
chmod 700 directory         # -rwx------
chmod 770 directory         # -rwxrwx---
chmod -R 755 /path          # Recursive
```

8.4 Changing Ownership

```
chown user file             # Change owner
chown user:group file       # Change owner and group
chown -R user:group dir     # Recursive
chgrp group file            # Change group only
```

8.4.1 Example: Multi-user Access

```
# Make directory owned by ansible, readable by devops group
chown -R ansible:devops /opt/devopsdir/
chmod 770 /opt/devopsdir/

# Now:
# - User 'ansible' has full access (rwx)
# - Users in 'devops' group have full access (rwx)
# - Others have no access (---)
```

9 Quick Reference Cheatsheet

9.1 Directory Navigation

```
pwd                 # Current directory
cd /path            # Change to absolute path
cd ~                # Home directory
cd -                # Previous directory
cd ..               # Parent directory
ls -la              # Detailed listing with hidden files
```

9.2 File Operations

```
touch file          # Create empty file
rm file             # Delete file
rm -r directory     # Delete directory recursively
cp source dest      # Copy
mv source dest      # Move/rename
```

9.3 System Information

```
uname -a            # System info
lsb_release -a      # OS version
df -h               # Disk usage
du -sh directory    # Directory size
ps aux              # Running processes
top                 # Real-time processes
```

Additional Resources

- Bash Manual
- Vim Official Website
- Vagrant Documentation
- VirtualBox
- Chocolatey Package Manager

Author Notes

This handbook is a comprehensive quick reference for Linux system administration, virtualization, and DevOps tools. It covers essential commands, file permissions, user management, and practical examples for both beginners and intermediate users.

For more information and updates, visit:

- **LinkedIn:** `https://www.linkedin.com/in/chinmaya-dalai`
- **GitHub:** `https://github.com/chinmaya-dalai`

Last Updated: December 2025