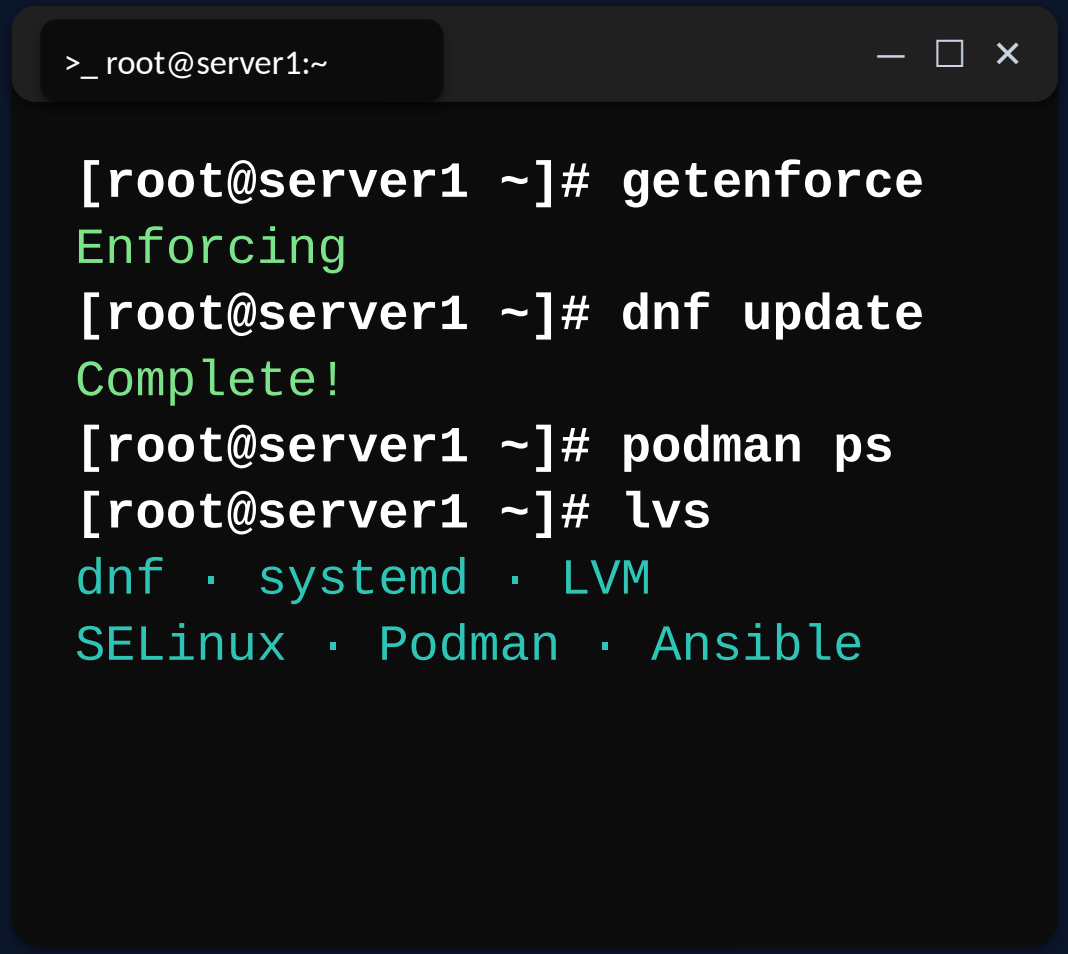


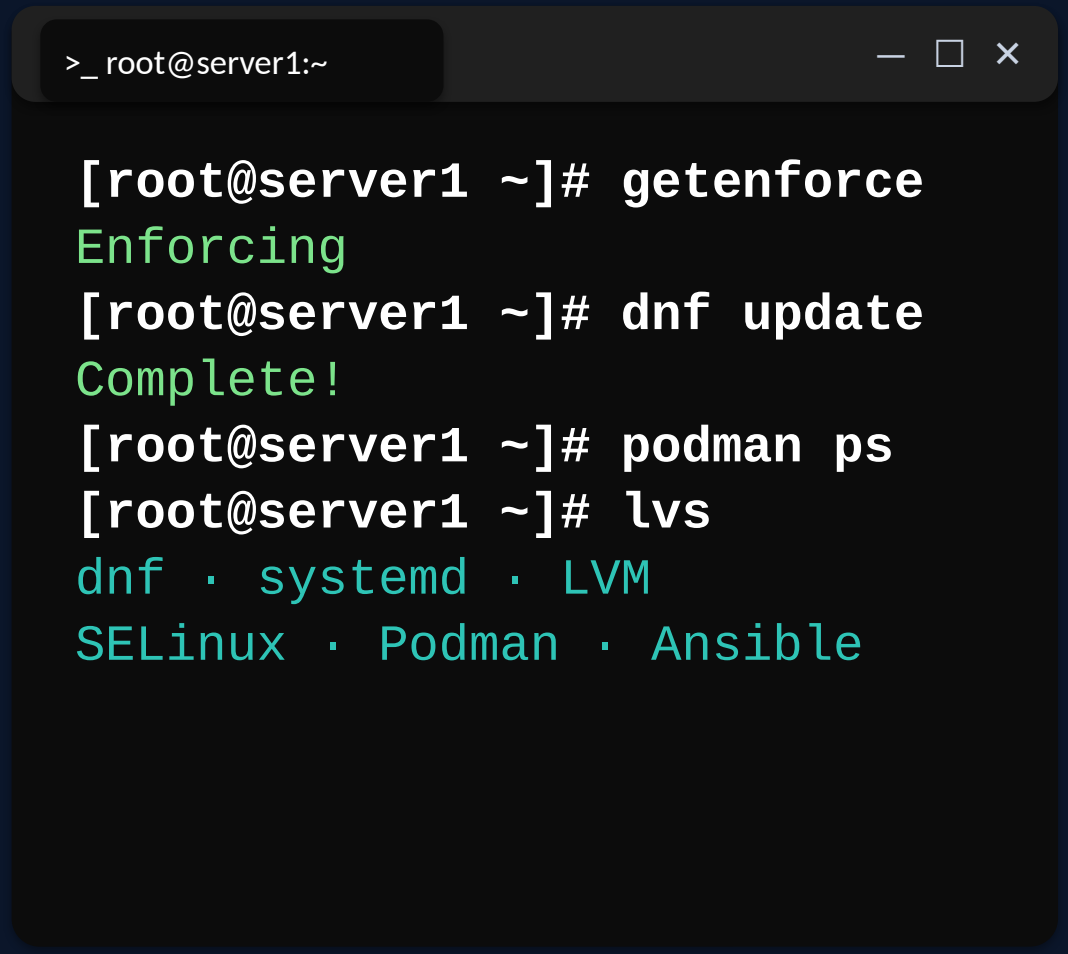
FULL COURSE · 12 LESSONS

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

dnf, systemd, LVM, firewalld, SELinux, Podman & Ansible

Full Course · RHEL 10.3 & 9.9 (Nov 2026)



LESSON 1 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

RHEL 9 and 10: Versions, Lifecycle, Free Subscription & Lab

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 1

- What RHEL is and where it runs
- The family: Fedora, CentOS Stream, RHEL, Rocky and AlmaLinux
- RHEL 9 vs RHEL 10: what changed
- Versions and the 10-year lifecycle
- The no-cost Red Hat Developer subscription
- Hardware requirements and our lab: server1 and server2
- How this course is organized

What Is RHEL?

Topic	Answer
Vendor	Red Hat, part of IBM since 2019
What you pay for	Subscription: updates, security fixes, certification and support
Source code	Open source; built from CentOS Stream
Where it runs	Bare metal, VMs, containers, AWS, Azure, Google Cloud, edge
Architectures	x86-64, ARM 64-bit, IBM Power, IBM Z
Certifications	RHCSA and RHCE are among the most respected Linux certifications

The Red Hat Family

Distribution	Role	Cost
Fedora	Upstream: new features first, new release every 6 months	Free
CentOS Stream	The next RHEL minor release, developed in the open	Free
RHEL	The supported enterprise product	Subscription
Rocky Linux	Community rebuild compatible with RHEL	Free
AlmaLinux	Community rebuild compatible with RHEL	Free
Oracle Linux	Oracle's RHEL-compatible distribution	Free / support

RHEL 9 vs RHEL 10

Area	RHEL 9 (2022)	RHEL 10 (2025)
Kernel	5.14	6.12
x86 CPU level	x86-64-v2	x86-64-v3 (about 2013+ CPUs)
Desktop	GNOME on Wayland or Xorg	GNOME on Wayland only; RDP for remote desktop
Packages	dnf 4 with modules	dnf 4, no modules; Application Streams
Security	SELinux, crypto policies	Adds post-quantum crypto, MCP server
New ways to deploy	Image builder	Image mode (bootc) is fully supported

Versions and Lifecycle

Version	Latest minor	Full support ends	Maintenance ends
RHEL 10	10.3 (Nov 11, 2026)	May 31, 2030	May 31, 2035
RHEL 9	9.9 (Nov 11, 2026)	May 31, 2027	May 31, 2032
RHEL 8	8.10 (final)	Ended May 2024	May 31, 2029
Minor releases	Every 6 months	During full support	Security fixes only after

The No-Cost Developer Subscription

Item	Details
Where	developers.redhat.com — create a free Red Hat account
What you get	RHEL for development and learning, up to 16 systems
Downloads	DVD (full) ISO and boot ISO for RHEL 10 and RHEL 9
Updates	Full access to Red Hat repositories after registering
Not for	Production business servers — those need a paid subscription
Alternative	Rocky Linux or AlmaLinux: free and compatible

Hardware Requirements

Item	Minimum	Recommended for the lab
CPU	x86-64-v3 (RHEL 10) or ARM 64	2 vCPUs per VM
RAM	2 GB (server, no GUI)	4 GB for server1, 2 GB for server2
Disk	10 GB	30 GB system disk + two 5 GB extra disks
Firmware	UEFI or BIOS	UEFI with Secure Boot
Host	Any 64-bit PC or Mac	16 GB RAM on the host
Hypervisor	KVM, VirtualBox, Hyper-V, VMware, UTM	Whatever you already have

Our Lab for This Course

Machine	OS	IP address	Purpose
server1	RHEL 10.3	10.0.0.11/24	Main server; most lessons
server2	RHEL 9.9	10.0.0.12/24	Second server: SSH, NFS, upgrade lesson
Network	lab network	10.0.0.0/24	Gateway 10.0.0.1, DNS 10.0.0.1
Users	root + student	—	We work as student and use sudo
Extra disks	server1	/dev/vdb, /dev/vdc	Storage and LVM in Lesson 7

Your First Look at RHEL

```
[student@server1 ~]$ cat /etc/redhat-release
Red Hat Enterprise Linux release 10.3 (Coughlan)
[student@server1 ~]$ uname -r
6.12.0-...el10.x86_64
[student@server1 ~]$ hostnamectl
[student@server1 ~]$ nproc ; free -h ; df -h /
[student@server1 ~]$ ld.so --help | grep supported     # x86-64-v3 (supported)?
```

How This Course Is Organized

Lessons	Topics
1–3	Versions and lab, installation and Cockpit, command line essentials
4–6	Users and permissions, software with dnf, systemd, boot and logs
7–8	Storage and LVM, networking, firewalld and SSH
9	SELinux — a full lesson
10–11	Apache, NFS, Podman; security, updates and upgrading 9 to 10
12	Scripting, Ansible, image mode, troubleshooting and RHCSA

Summary and Homework

- RHEL: the supported enterprise Linux; Fedora and CentOS Stream come first
- RHEL 10 for new servers; RHEL 9 is still everywhere
- Ten-year lifecycle; minor releases every six months
- Free developer subscription: up to 16 systems
- Homework: create your Red Hat account and download the RHEL 10 DVD ISO
- Homework: create the server1 and server2 VMs (no install yet)

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

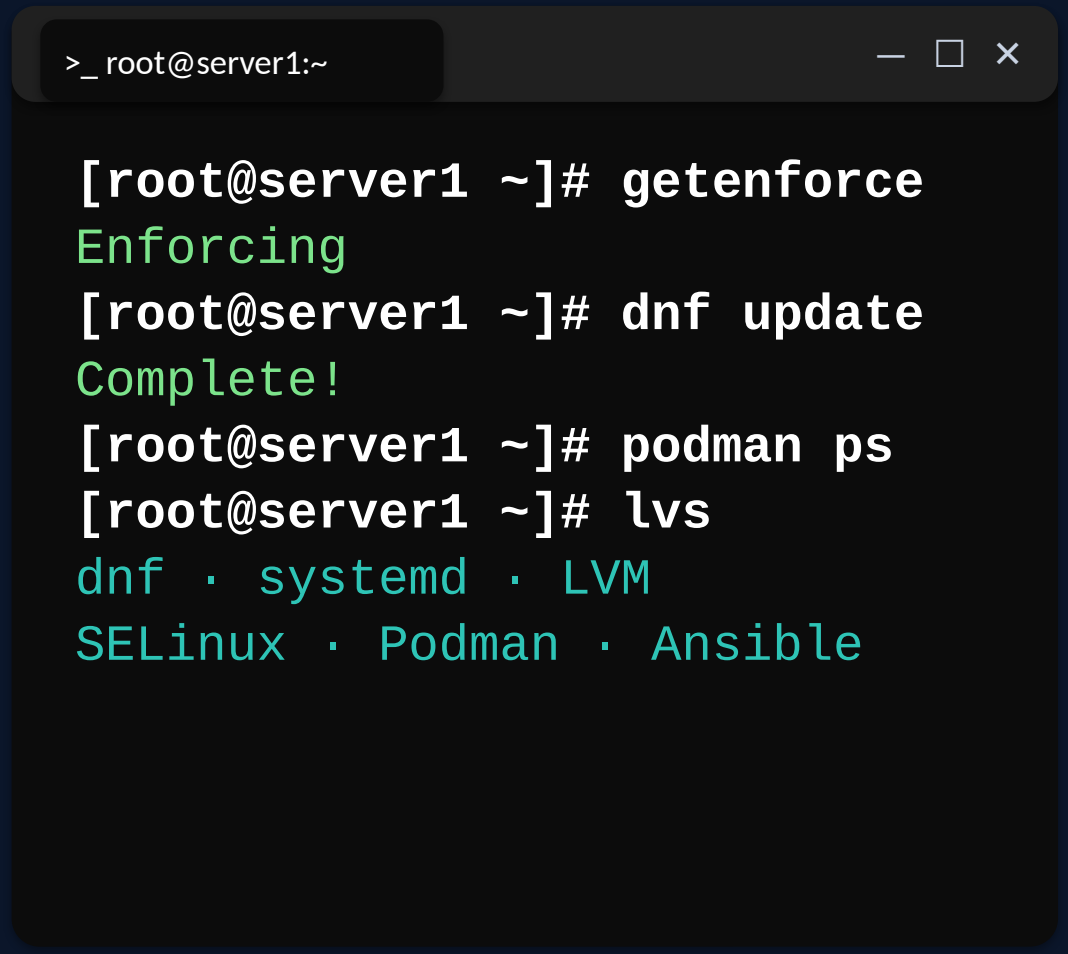
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 2 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Installing RHEL, Registering & the Web Console (Cockpit)

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 2

- The installation ISOs and the Anaconda installer
- Step by step: language, disks, network, users, software
- Server with GUI vs Server vs Minimal
- Registering with subscription-manager or rhc
- First tasks: updates, hostname, time zone
- The web console: Cockpit on port 9090
- Kickstart: automated installation

Installation Media

ISO	Size	Use it when
DVD (binary) ISO	about 8–12 GB	Normal choice: everything included, works offline
Boot ISO	about 1 GB	Small; packages come from the network after registration
Cloud images (qcow2, AMI)	small	Ready-made VMs and cloud instances
Image mode (bootc) images	container image	Deploy RHEL as a container image (Lesson 12)
USB stick	—	Write the ISO with Fedora Media Writer or dd

The Anaconda Installer

Screen	What to choose in the lab
Language and keyboard	English (United States), or your own
Installation destination	The 30 GB disk, automatic partitioning (LVM)
Network and host name	Enable the NIC; host name server1.lab.local
Time and date	Your time zone; network time on
Software selection	Server (no GUI) for server2; Server with GUI for server1
Root account	Locked or strong password; no root SSH login
User creation	student, with administrator rights (wheel group)

Software Selection

Base environment	What you get	Best for
Server with GUI	GNOME desktop + server tools	Learning, admin workstations
Server	Command line + common server tools	Most real servers
Minimal Install	The smallest working system	Containers hosts, security, cloud
Workstation	Desktop for daily work	Developer laptops
Add-ons	Extra package groups	Add later with dnf group install

Register the System

```
[student@server1 ~]$ sudo rhc connect -u <your-Red-Hat-login>
# or the classic tool:
[student@server1 ~]$ sudo subscription-manager register --username <your-Red-Hat-login>
[student@server1 ~]$ sudo subscription-manager status
[student@server1 ~]$ sudo dnf repolist
repo id                                    repo name
rhel-10-for-x86_64-appstream-rpms          Red Hat Enterprise Linux 10 ... AppStream
rhel-10-for-x86_64-baseos-rpms             Red Hat Enterprise Linux 10 ... BaseOS
```

First Tasks After Installation

```
[student@server1 ~]$ sudo dnf -y update            # install all updates
[student@server1 ~]$ sudo reboot                   # if the kernel was updated

[student@server1 ~]$ sudo hostnamectl set-hostname server1.lab.local
[student@server1 ~]$ timedatectl                   # time zone and NTP status
[student@server1 ~]$ sudo timedatectl set-timezone America/Los_Angeles
[student@server1 ~]$ ip -brief address             # check the IP address
[student@server1 ~]$ sudo systemctl enable --now cockpit.socket
```

The Web Console: Cockpit

Feature	What you can do
Address	https://server1.lab.local:9090 — sign in as student
Overview	CPU, memory, disk and network graphs; updates; reboot
Logs and services	Read the journal; start, stop and enable services
Storage and networking	Disks, LVM, file systems; interfaces, firewall
Accounts and terminal	Create users; a full terminal in the browser
Add-ons	cockpit-podman, cockpit-machines, cockpit-files

Install Cockpit and Add-ons

```
[student@server2 ~]$ sudo dnf install -y cockpit cockpit-podman cockpit-files
[student@server2 ~]$ sudo systemctl enable --now cockpit.socket
[student@server2 ~]$ sudo firewall-cmd --add-service=cockpit --permanent
[student@server2 ~]$ sudo firewall-cmd --reload
[student@server2 ~]$ ss -tlnp | grep 9090
LISTEN 0  4096   *:9090   *:*
# From server1 Cockpit: Add new host > server2.lab.local
```

Kickstart: Automated Installation

```
# /root/anaconda-ks.cfg is created on every installed system
lang en_US.UTF-8
keyboard us
timezone America/Los_Angeles --utc
network --bootproto=static --ip=10.0.0.13 --hostname=server3.lab.local
rootpw --lock
user --name=student --groups=wheel --password=<hash> --iscrypted
autopart --type=lvm
%packages
@^server-product-environment
%end
```

Installing RHEL 9 on server2

Item	Difference from RHEL 10
Installer	Same Anaconda; screens look almost identical
CPU	Runs on older x86-64-v2 CPUs
Desktop	Xorg session still available
Remote GUI	VNC available; RHEL 10 uses RDP
Register and update	Same commands: rhc, subscription-manager, dnf
Repositories	rhel-9-for-x86_64-baseos-rpms and appstream-rpms

Summary and Homework

- Install from the DVD ISO with Anaconda; automatic LVM partitioning
- Create student as an administrator (wheel); lock root
- Register with rhc connect or subscription-manager, then dnf update
- Cockpit: https://host:9090 for browser-based management
- Homework: install RHEL 10 on server1 and RHEL 9 on server2
- Homework: register both, update, enable Cockpit, read anaconda-ks.cfg

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

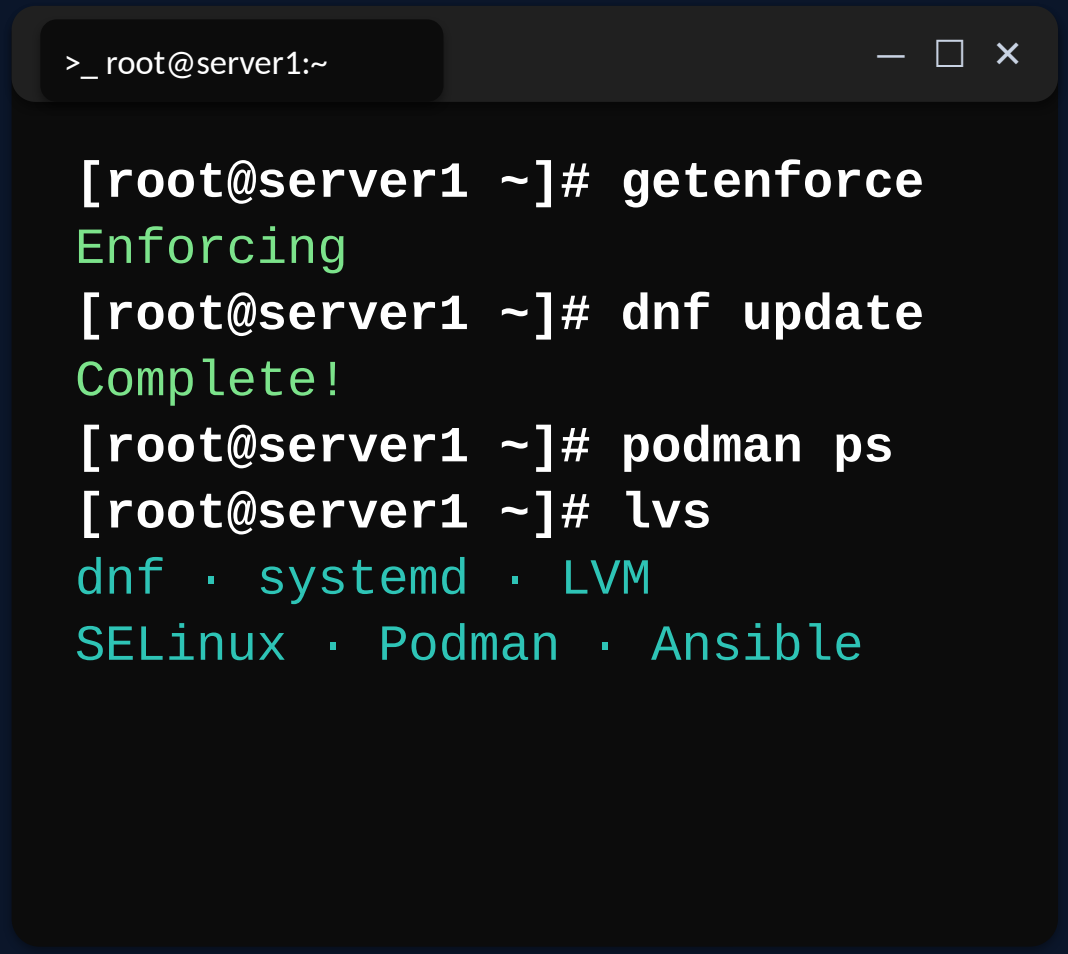
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 3 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Command Line Essentials: Files, Help, Editing & Pipes

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 3

- The bash shell, the prompt and command syntax
- The file system hierarchy: /etc, /var, /home and more
- Navigating and managing files and directories
- Getting help: man, --help, /usr/share/doc
- Editing files with vim and nano
- Redirection, pipes and grep
- Finding files, links and archives with tar

The Shell and the Prompt

Part	Meaning
[student@server1 ~]$	User, host, current directory; $ = normal user
[root@server1 ~]#	# = root, the administrator; be careful
command -options arguments	ls -l /etc — command, option, argument
Tab key	Completes commands and file names
Up arrow, history, Ctrl+R	Repeat and search earlier commands
Ctrl+C / Ctrl+D / clear	Stop a command / log out / clear the screen

The File System Hierarchy

Directory	Contents
/	The root of everything
/etc	System configuration files
/home, /root	Users' home directories; root's home
/var	Changing data: logs (/var/log), web sites, databases
/usr	Installed programs and libraries (/usr/bin, /usr/lib)
/tmp	Temporary files, cleaned automatically
/boot, /dev, /proc, /run	Kernel and boot files; devices; live system info

Navigate and Manage Files

```
[student@server1 ~]$ pwd                        # where am I?
/home/student
[student@server1 ~]$ ls -la /etc | less         # long list, hidden files, page by page
[student@server1 ~]$ cd /var/log ; cd ~ ; cd -   # go there, go home, go back
[student@server1 ~]$ mkdir -p projects/web/html
[student@server1 ~]$ touch notes.txt
[student@server1 ~]$ cp notes.txt projects/ ; cp -r projects backup
[student@server1 ~]$ mv notes.txt todo.txt       # rename or move
[student@server1 ~]$ rm todo.txt ; rm -r backup  # no recycle bin!
```

Getting Help

Command	What it gives you
man ls	The manual page; q to quit, /word to search
man -k password	Search manual page titles for a word
man 5 passwd	Section 5: file formats (section 1 = commands, 8 = admin)
ls --help	A short summary of options
/usr/share/doc	Extra documentation and examples
docs.redhat.com	Official RHEL guides, free, for each version

Editing Files with vim

Keys	Action
vim file.txt	Open (or create) a file in normal mode
i / Esc	Insert mode to type / back to normal mode
:w :q :wq :q!	Save / quit / save and quit / quit without saving
dd yy p u	Delete line / copy line / paste / undo
/text n	Search forward / next match
:%s/old/new/g	Replace everywhere in the file
nano file.txt	A simpler editor: commands shown at the bottom

Redirection and Pipes

```
[student@server1 ~]$ ls /etc > etc-list.txt          # stdout to a file (overwrite)
[student@server1 ~]$ date >> etc-list.txt            # append
[student@server1 ~]$ ls /nothing 2> errors.txt       # stderr to a file
[student@server1 ~]$ dnf list installed &> pkgs.txt  # both stdout and stderr
[student@server1 ~]$ ls /etc | wc -l                 # pipe: count the files
[student@server1 ~]$ ps aux | sort -k3 -nr | head -5 # top 5 CPU users
[student@server1 ~]$ cat /etc/passwd | cut -d: -f1 | sort
[student@server1 ~]$ echo 'hello' | tee copy.txt      # screen AND file
```

Search Text with grep

```
[student@server1 ~]$ grep student /etc/passwd
student:x:1000:1000:student:/home/student:/bin/bash
[student@server1 ~]$ grep -i error /var/log/messages     # ignore case
[student@server1 ~]$ grep -r 'Listen' /etc/httpd/ 2>/dev/null
[student@server1 ~]$ grep -v '^#' /etc/ssh/sshd_config | grep -v '^$'
[student@server1 ~]$ grep -E 'fail(ed|ure)' /var/log/secure
[student@server1 ~]$ grep -c bash /etc/passwd              # count matching lines
```

Find Files and Make Links

```
[student@server1 ~]$ find /etc -name '*.conf' -mtime -7      # changed in 7 days
[student@server1 ~]$ find / -user student -type f 2>/dev/null
[student@server1 ~]$ find /var/log -size +10M
[student@server1 ~]$ find /tmp -name '*.tmp' -delete
[student@server1 ~]$ locate sshd_config        # fast; needs: sudo dnf install plocate
[student@server1 ~]$ ln -s /var/log/messages mylog   # symbolic (soft) link
[student@server1 ~]$ ln notes.txt hardlink.txt       # hard link: same inode
[student@server1 ~]$ ls -li mylog hardlink.txt
```

Archives and Compression with tar

```
[student@server1 ~]$ tar -czvf etc-backup.tar.gz /etc        # create, gzip
[student@server1 ~]$ tar -cJf etc-backup.tar.xz /etc         # create, xz (smaller)
[student@server1 ~]$ tar -tzf etc-backup.tar.gz | head        # list contents
[student@server1 ~]$ tar -xzf etc-backup.tar.gz -C /tmp/restore
[student@server1 ~]$ gzip big.log ; gunzip big.log.gz
[student@server1 ~]$ zip -r site.zip projects/ ; unzip site.zip
[student@server1 ~]$ du -sh /var/log ; ls -lh etc-backup.*
```

Summary and Homework

- Prompt: $ normal user, # root; Tab and history save time
- /etc config, /var data and logs, /home users, /usr programs
- man, man -k and --help: answers are always on the system
- vim: i to insert, Esc, :wq to save and quit
- Pipes and redirection: | > >> 2> &>
- Homework: practice vimtutor; find all .conf files in /etc; archive /etc with tar

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

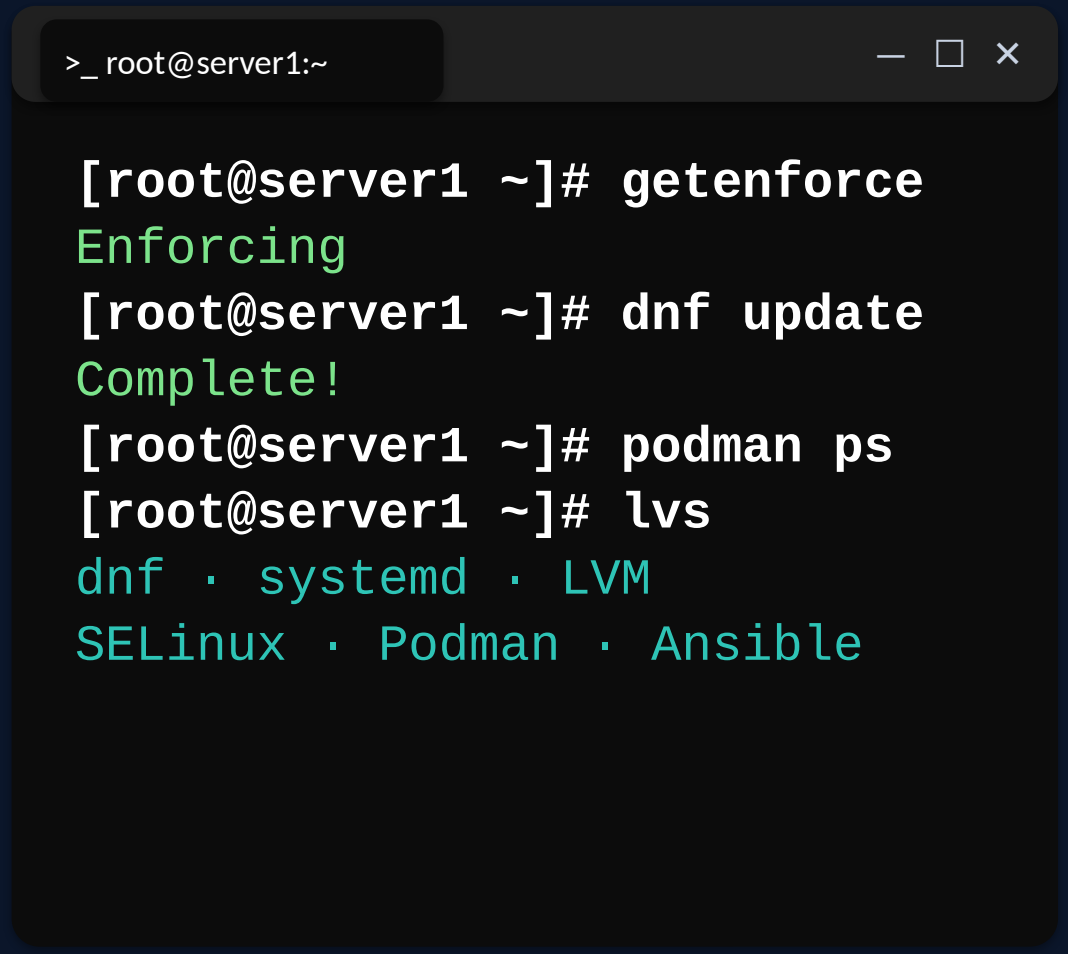
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 4 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Users, Groups, Permissions, ACLs & sudo

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 4

- Users, groups, UIDs and the files behind them
- Creating and managing users and groups
- Passwords and password aging
- File permissions: read, write, execute; chmod and chown
- Special permissions: setgid, sticky bit; umask
- ACLs for fine-grained access
- sudo and the wheel group

Users and Groups: The Basics

Item	Details
root	UID 0: can do anything
System users	UID 1–999: run services such as apache or sshd
Normal users	UID 1000 and up: people
Primary group	Each user has one; new files get this group
Supplementary groups	Extra groups, such as wheel for administrators
id student	uid=1000(student) gid=1000(student) groups=…,10(wheel)

The Files Behind Users

File	Holds
/etc/passwd	name:x:UID:GID:comment:home:shell — readable by everyone
/etc/shadow	Password hashes and aging — root only
/etc/group	group:x:GID:members
/etc/gshadow	Group passwords and admins — root only
/etc/login.defs	Defaults: UID ranges, password aging, home creation
/etc/skel	Files copied into every new home directory

Create and Manage Users and Groups

```
[student@server1 ~]$ sudo groupadd -g 2000 developers
[student@server1 ~]$ sudo useradd -c 'Sara Ahmadi' -G developers sara
[student@server1 ~]$ sudo passwd sara
[student@server1 ~]$ sudo useradd -u 1500 -s /sbin/nologin svc-backup   # no login
[student@server1 ~]$ sudo usermod -aG wheel sara       # -a: append, keep old groups!
[student@server1 ~]$ sudo usermod -L sara ; sudo usermod -U sara   # lock / unlock
[student@server1 ~]$ sudo userdel -r olduser           # delete with home directory
[student@server1 ~]$ id sara ; getent group developers
```

Passwords and Password Aging

```
[student@server1 ~]$ sudo chage -l sara
Last password change                    : Nov 20, 2026
Password expires                        : never
[student@server1 ~]$ sudo chage -M 90 -m 1 -W 7 sara     # max 90 days, warn 7 days before
[student@server1 ~]$ sudo chage -d 0 sara                 # must change at next login
[student@server1 ~]$ sudo chage -E 2027-06-30 contractor  # account expires
[student@server1 ~]$ grep ^PASS_ /etc/login.defs            # defaults for new users
[student@server1 ~]$ cat /etc/security/pwquality.conf       # minlen, complexity
```

File Permissions

Part of ls -l	Meaning
-rw-r----- 1 sara developers report.txt	Type, permissions, owner, group, name
rw- r-- ---	Owner (u) / group (g) / others (o)
r = 4, w = 2, x = 1	Read, write, execute — add them: rw- = 6
On files	r read, w change, x run as a program
On directories	r list, w create/delete files, x enter (cd)
Who wins?	Owner first, then group, then others

chmod and chown

```
[student@server1 ~]$ chmod 640 report.txt            # rw-r-----
[student@server1 ~]$ chmod u+x,g-w,o= script.sh      # symbolic form
[student@server1 ~]$ chmod -R g+rwX /srv/projects    # X: execute on directories only
[student@server1 ~]$ sudo chown sara:developers report.txt
[student@server1 ~]$ sudo chown -R :developers /srv/projects
[student@server1 ~]$ ls -l report.txt
-rw-r----- 1 sara developers 2048 Nov 20 10:00 report.txt
[student@server1 ~]$ stat -c '%A %a %U:%G %n' report.txt
```

Special Permissions and umask

Setting	Effect
setuid (4, u+s)	Program runs as its owner, e.g. /usr/bin/passwd
setgid on a directory (2, g+s)	New files inherit the directory's group — great for teams
sticky bit (1, o+t)	Only the owner can delete their files, e.g. /tmp
chmod 2770 /srv/projects	Team folder: group shares, others locked out
umask 022 / 002 / 077	Removes rights from new files: 644/755, 664/775, 600/700
Set umask	~/.bashrc for a user; /etc/login.defs for defaults

ACLs: Fine-Grained Access

```
[student@server1 ~]$ setfacl -m u:ahmad:rw report.txt       # one extra user
[student@server1 ~]$ setfacl -m g:auditors:r report.txt     # one extra group
[student@server1 ~]$ setfacl -R -m d:g:developers:rwX /srv/projects   # default ACL
[student@server1 ~]$ getfacl report.txt
# file: report.txt
user::rw-
user:ahmad:rw-
group::r--
group:auditors:r--
mask::rw-
[student@server1 ~]$ setfacl -x u:ahmad report.txt ; setfacl -b report.txt   # remove
```

sudo: Safe Administrator Rights

```
[student@server1 ~]$ sudo -l                    # what may I run?
[student@server1 ~]$ sudo -i                    # root shell (logged)
[student@server1 ~]$ sudo visudo -f /etc/sudoers.d/webteam
%webteam  ALL=(root)  /usr/bin/systemctl restart httpd, /usr/bin/systemctl status httpd
ahmad     ALL=(ALL)   ALL

# wheel group already has full sudo in /etc/sudoers:
%wheel    ALL=(ALL)   ALL
[student@server1 ~]$ sudo journalctl _COMM=sudo --since today   # who ran what
```

Summary and Homework

- useradd, usermod -aG, passwd, chage, userdel -r
- rwx = 421; chmod 640, chmod g+s, chown user:group
- setgid team folders; umask controls new file rights
- setfacl/getfacl when one owner and one group are not enough
- sudo + wheel; least privilege with /etc/sudoers.d
- Homework: create group developers, users sara and ahmad, and a setgid team folder /srv/projects

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

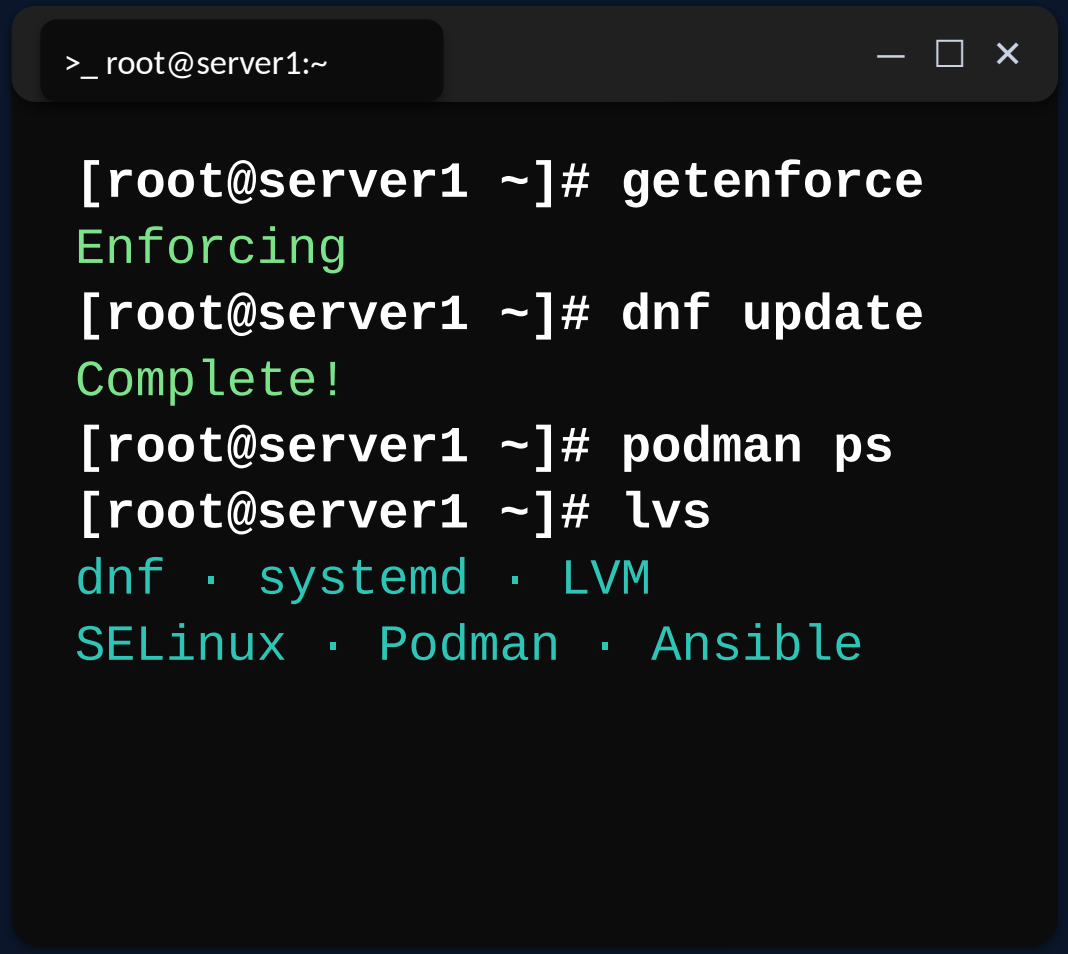
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 5 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Software Management with DNF, RPM, Repos & Flatpak

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 5

- RPM packages and repositories: BaseOS and AppStream
- dnf: search, install, update, remove
- Package groups and environments
- History and undo
- Application Streams: newer versions of languages and databases
- Querying packages with rpm
- Adding repositories: EPEL and your own; Flatpak

Packages and Repositories

Term	Meaning
RPM package	httpd-2.4.63-1.el10.x86_64.rpm — name, version, release, architecture
Repository	A signed collection of packages that dnf downloads from
BaseOS	The core operating system, supported for the whole life of the version
AppStream	Applications, languages, databases and tools
CodeReady Builder	Extra developer packages (enable when needed)
GPG signature	Proves a package really came from Red Hat

Search, Install and Remove

```
[student@server1 ~]$ dnf search nginx
[student@server1 ~]$ dnf info httpd
[student@server1 ~]$ dnf provides /usr/bin/dig       # which package has this file?
bind-utils-9.18...el10.x86_64 : Utilities for querying DNS name servers
[student@server1 ~]$ sudo dnf install -y httpd bind-utils
[student@server1 ~]$ sudo dnf remove nginx
[student@server1 ~]$ sudo dnf reinstall httpd
[student@server1 ~]$ dnf list installed | grep httpd
[student@server1 ~]$ sudo dnf install ./local-tool-1.0.rpm   # a downloaded file
```

Updates

```
[student@server1 ~]$ dnf check-update                  # what is waiting?
[student@server1 ~]$ sudo dnf update -y                # update everything
[student@server1 ~]$ sudo dnf update --security        # security fixes only
[student@server1 ~]$ dnf updateinfo list --security    # advisories (RHSA)
[student@server1 ~]$ sudo dnf update openssl           # one package
[student@server1 ~]$ dnf needs-restarting -r           # is a reboot needed?
[student@server1 ~]$ sudo dnf install dnf-automatic
[student@server1 ~]$ sudo systemctl enable --now dnf-automatic.timer
```

Groups, History and Undo

```
[student@server1 ~]$ dnf group list
[student@server1 ~]$ dnf group info 'Development Tools'
[student@server1 ~]$ sudo dnf group install -y 'Development Tools'
[student@server1 ~]$ dnf history
ID | Command line              | Date and time    | Action(s) | Altered
 7 | group install Developmen.. | 2026-11-20 10:42 | Install   |   38
 6 | install -y httpd bind-u..  | 2026-11-20 10:30 | Install   |    9
[student@server1 ~]$ dnf history info 7
[student@server1 ~]$ sudo dnf history undo 7           # remove those 38 packages
```

Application Streams

Item	RHEL 9	RHEL 10
Idea	Several versions of a language or database	Same idea
How	Modules: dnf module enable nodejs:22	Versioned packages, e.g. nodejs24
Examples	nodejs, php, postgresql, ruby modules	python3.12, postgresql18, nodejs24
List	dnf module list	dnf list --available 'postgresql*'
Life cycle	Each stream has its own support dates	Check the Application Streams life cycle page

Query Packages with rpm

```
[student@server1 ~]$ rpm -qa | wc -l                    # how many packages?
[student@server1 ~]$ rpm -qi httpd                      # information
[student@server1 ~]$ rpm -ql httpd | head               # files in the package
[student@server1 ~]$ rpm -qc httpd                      # its configuration files
[student@server1 ~]$ rpm -qf /etc/ssh/sshd_config      # which package owns this file?
openssh-server-9.9p1-...el10.x86_64
[student@server1 ~]$ rpm -V openssh-server              # what changed since install?
S.5....T.  c /etc/ssh/sshd_config
[student@server1 ~]$ rpm -q --changelog openssl | head
```

Repositories: EPEL and Your Own

```
[student@server1 ~]$ dnf repolist --all
[student@server1 ~]$ sudo subscription-manager repos \
      --enable codeready-builder-for-rhel-10-x86_64-rpms
[student@server1 ~]$ sudo dnf install \
      https://dl.fedoraproject.org/pub/epel/epel-release-latest-10.noarch.rpm
[student@server1 ~]$ sudo dnf install htop

[student@server1 ~]$ sudo vim /etc/yum.repos.d/internal.repo
[internal]
name=Company packages
baseurl=http://10.0.0.11/repo/
gpgcheck=1
gpgkey=http://10.0.0.11/repo/RPM-GPG-KEY-internal
```

Flatpak and Desktop Apps

```
[student@server1 ~]$ flatpak remotes
Name    Options
rhel    system,oci
[student@server1 ~]$ flatpak search firefox
[student@server1 ~]$ flatpak install rhel org.mozilla.firefox
[student@server1 ~]$ flatpak list --app
[student@server1 ~]$ flatpak update
# Optional, community apps:
[student@server1 ~]$ flatpak remote-add --if-not-exists flathub \
      https://dl.flathub.org/repo/flathub.flatpakrepo
```

Summary and Homework

- BaseOS + AppStream; packages are GPG-signed RPMs
- dnf search, info, provides, install, remove, update
- dnf history undo is your safety net
- rpm -qa, -qi, -ql, -qc, -qf, -V answer questions
- EPEL for extras (unsupported); Flatpak for desktop apps
- Homework: install httpd and Development Tools, find who owns /etc/ssh/sshd_config, undo the group install

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

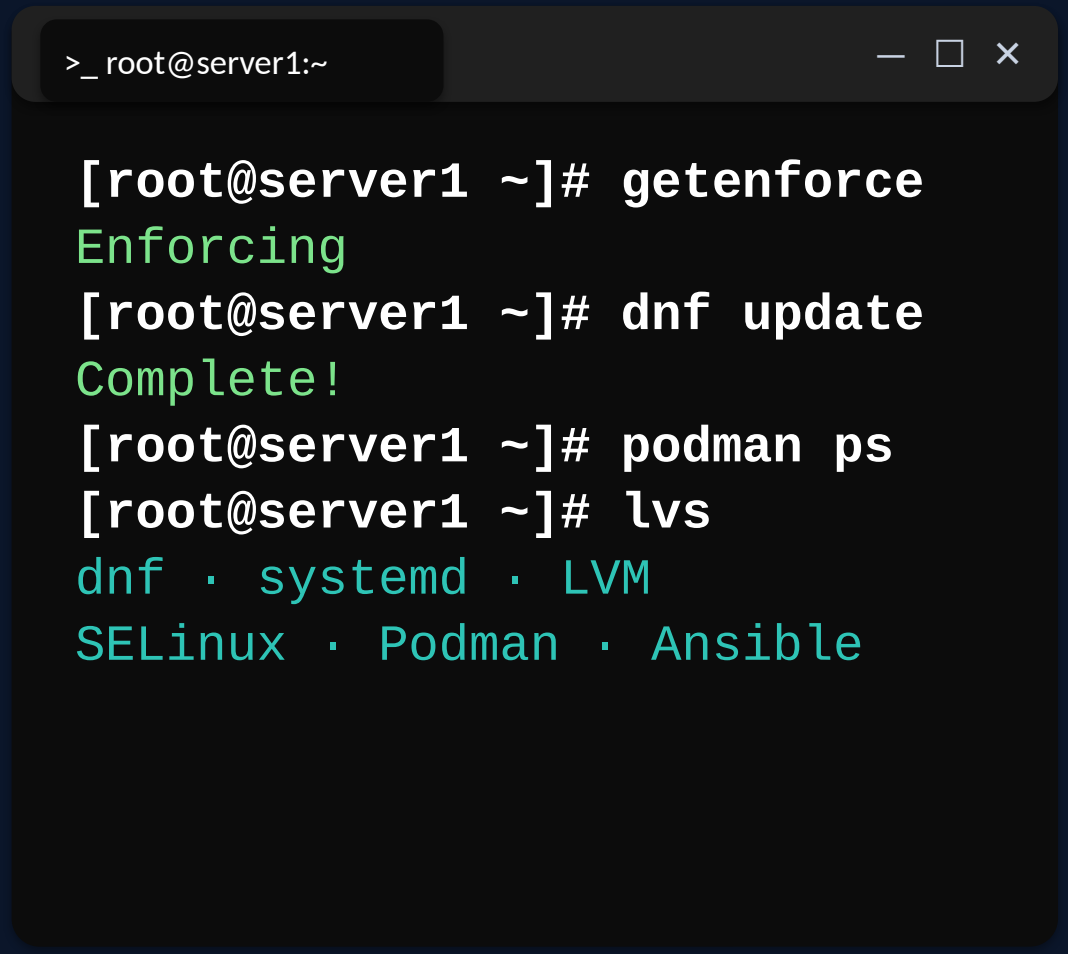
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 6 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

systemd, Boot, Logs, Processes & Scheduling

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 6

- systemd units and systemctl
- The boot process: firmware, GRUB, kernel, systemd, targets
- Recovering a lost root password
- Logs: journalctl and /var/log
- Processes: ps, top, kill, nice
- Performance profiles with tuned
- Scheduling: cron, at and systemd timers

systemd and Units

Unit type	Example	Purpose
service	sshd.service	A daemon (background program)
socket	cockpit.socket	Starts a service when a connection arrives
target	multi-user.target	A group of units: a system state
timer	dnf-automatic.timer	Runs a service on a schedule
mount	home.mount	A mounted file system
Unit files	/usr/lib/systemd/system	Your changes go in /etc/systemd/system

Managing Services with systemctl

```
[student@server1 ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
     Active: active (running) since Fri 2026-11-20 09:12:01 PST
[student@server1 ~]$ sudo systemctl start httpd ; sudo systemctl stop httpd
[student@server1 ~]$ sudo systemctl restart httpd ; sudo systemctl reload httpd
[student@server1 ~]$ sudo systemctl enable --now httpd     # start now and at boot
[student@server1 ~]$ systemctl is-enabled httpd ; systemctl is-active httpd
[student@server1 ~]$ systemctl list-units --type=service --state=failed
[student@server1 ~]$ sudo systemctl mask cups               # can never be started
```

Your Own Settings: Drop-ins and Unit Files

```
[student@server1 ~]$ sudo systemctl edit httpd          # creates a drop-in override
[Service]
Restart=on-failure
RestartSec=5
[student@server1 ~]$ systemctl cat httpd                # unit + overrides

# /etc/systemd/system/hello.service — your own service
[Unit]
Description=Hello web app
After=network-online.target
[Service]
ExecStart=/usr/local/bin/hello --port 8080
User=svc-hello
[Install]
WantedBy=multi-user.target
```

The Boot Process

Step	What happens
1. Firmware	UEFI or BIOS tests hardware; Secure Boot checks signatures
2. Boot loader	GRUB 2 shows the menu and loads the kernel and initramfs
3. Kernel	Starts, uses the initramfs to find the root file system
4. systemd	PID 1 reads the default target and starts units in parallel
5. Target	multi-user.target (text) or graphical.target (desktop)
Commands	systemctl get-default / set-default; systemd-analyze blame

Reset a Lost Root Password

```
# 1. Reboot; in the GRUB menu press e on the kernel entry
# 2. At the end of the line starting with 'linux' add:   rd.break
# 3. Press Ctrl+X to boot into the initramfs shell
switch_root:/# mount -o remount,rw /sysroot
switch_root:/# chroot /sysroot
sh-5.2# passwd root
sh-5.2# touch /.autorelabel                # SELinux must relabel the files
sh-5.2# exit ; exit                        # the system relabels and reboots

# Protect GRUB in production:  sudo grub2-setpassword
```

Reading Logs with journalctl

```
[student@server1 ~]$ journalctl -u sshd --since '1 hour ago'
[student@server1 ~]$ journalctl -p err -b              # errors since this boot
[student@server1 ~]$ journalctl -f                     # follow live, like tail -f
[student@server1 ~]$ journalctl -b -1                  # the previous boot
[student@server1 ~]$ journalctl _UID=1000 -o verbose
[student@server1 ~]$ sudo mkdir -p /var/log/journal    # keep logs across reboots
[student@server1 ~]$ sudo less /var/log/messages ; sudo tail /var/log/secure
[student@server1 ~]$ logger 'backup finished'          # write your own message
```

Processes

```
[student@server1 ~]$ ps aux --sort=-%mem | head -5
[student@server1 ~]$ ps -ef --forest | less          # parent/child tree
[student@server1 ~]$ pgrep -a httpd
[student@server1 ~]$ top                              # press 1, M, P, k, q
[student@server1 ~]$ kill 4321 ; kill -9 4321         # TERM (polite) / KILL (forced)
[student@server1 ~]$ pkill -u sara                    # all of sara's processes
[student@server1 ~]$ nice -n 10 tar -czf big.tgz /data ; sudo renice -n -5 -p 4321
[student@server1 ~]$ sleep 300 &  ;  jobs  ;  fg %1   # background jobs
```

Performance Profiles with tuned

```
[student@server1 ~]$ tuned-adm active
Current active profile: virtual-guest
[student@server1 ~]$ tuned-adm list
balanced  powersave  throughput-performance  virtual-guest ...
[student@server1 ~]$ tuned-adm recommend
virtual-guest
[student@server1 ~]$ sudo tuned-adm profile throughput-performance
[student@server1 ~]$ uptime ; vmstat 2 5 ; free -h     # load and memory
```

Scheduling: cron, at and timers

```
[student@server1 ~]$ crontab -e                          # your personal jobs
# min hour day month weekday  command
30 2 * * *   /home/student/bin/backup.sh                # every day at 02:30
*/15 8-17 * * 1-5  /usr/local/bin/check-site            # every 15 min, office hours
[student@server1 ~]$ crontab -l ; sudo crontab -u sara -l
[student@server1 ~]$ echo 'systemctl restart httpd' | sudo at 23:00   # once
[student@server1 ~]$ atq
[student@server1 ~]$ systemctl list-timers               # systemd timers
```

Summary and Homework

- systemctl enable --now, status, restart; overrides with systemctl edit
- Boot: firmware → GRUB 2 → kernel → systemd → target
- Root password reset: rd.break, chroot, passwd, touch /.autorelabel
- journalctl -u, -p err, -b, -f; persistent journal
- ps, top, kill, nice; tuned-adm profile; cron, at, timers
- Homework: reset root's password on server2; schedule a daily tar backup of /etc

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

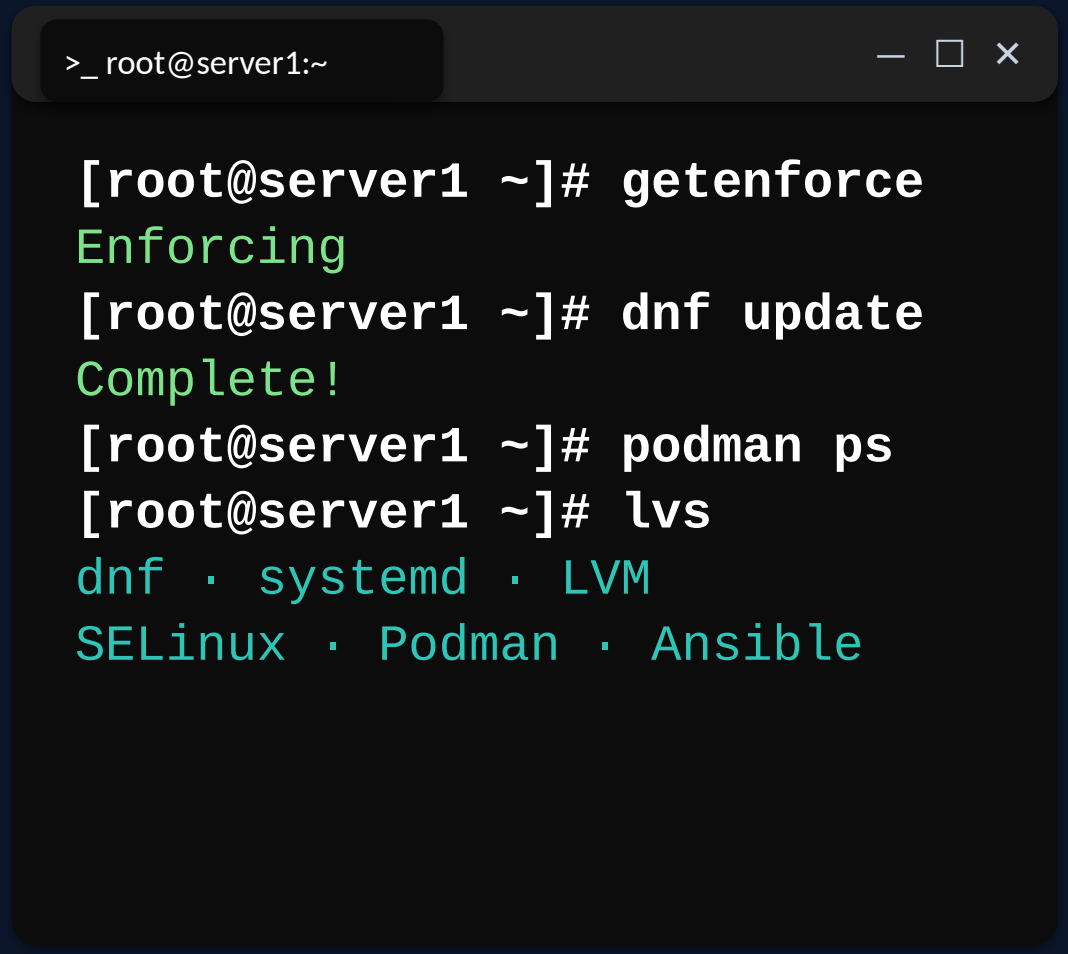
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 7 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Storage: Partitions, XFS, LVM, Swap & Stratis

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 7

- Disks, partitions and device names
- Partitioning with parted: GPT
- File systems: XFS and ext4; mounting
- Permanent mounts in /etc/fstab with UUIDs
- LVM: physical volumes, volume groups, logical volumes
- Growing a file system while it is in use
- Swap space; Stratis; LUKS encryption

Find Your Disks

```
[student@server1 ~]$ lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
vda           252:0    0   30G  0 disk
├─vda1        252:1    0  600M  0 part /boot/efi
├─vda2        252:2    0    1G  0 part /boot
└─vda3        252:3    0 28.4G  0 part
  ├─rhel-root 253:0    0 25.4G  0 lvm  /
  └─rhel-swap 253:1    0    3G  0 lvm  [SWAP]
vdb           252:16   0    5G  0 disk
vdc           252:32   0    5G  0 disk
[student@server1 ~]$ lsblk -f ; df -hT ; sudo blkid
```

Partition a Disk with parted

```
[student@server1 ~]$ sudo parted /dev/vdb mklabel gpt
[student@server1 ~]$ sudo parted /dev/vdb mkpart data xfs 1MiB 2GiB
[student@server1 ~]$ sudo parted /dev/vdb mkpart lvm1 2GiB 100%
[student@server1 ~]$ sudo parted /dev/vdb set 2 lvm on
[student@server1 ~]$ sudo parted /dev/vdb print
Number  Start   End     Size    File system  Name  Flags
 1      1049kB  2147MB  2146MB               data
 2      2147MB  5368MB  3221MB               lvm1  lvm
[student@server1 ~]$ sudo udevadm settle ; lsblk /dev/vdb
```

Create File Systems and Mount Them

```
[student@server1 ~]$ sudo mkfs.xfs -L data /dev/vdb1      # XFS: the RHEL default
[student@server1 ~]$ sudo mkdir /data
[student@server1 ~]$ sudo mount /dev/vdb1 /data
[student@server1 ~]$ df -h /data
Filesystem      Size  Used Avail Use% Mounted on
/dev/vdb1       2.0G   47M  1.9G   3% /data
[student@server1 ~]$ sudo umount /data
# ext4 is also supported:  sudo mkfs.ext4 /dev/vdX1
# XFS can grow but cannot shrink; ext4 can do both (offline)
```

Permanent Mounts in /etc/fstab

```
[student@server1 ~]$ sudo blkid /dev/vdb1
/dev/vdb1: LABEL="data" UUID="5b2e7c1a-..." TYPE="xfs"
[student@server1 ~]$ sudo vim /etc/fstab
# device                                mount  type  options   dump fsck
UUID=5b2e7c1a-...                       /data  xfs   defaults  0 0
[student@server1 ~]$ sudo systemctl daemon-reload
[student@server1 ~]$ sudo mount -a          # test: mounts everything in fstab
[student@server1 ~]$ findmnt /data
# A mistake here can stop the boot: ALWAYS test with mount -a
```

LVM: Logical Volume Manager

Layer	What it is	Commands
Physical volume (PV)	A disk or partition given to LVM	pvcreate, pvs
Volume group (VG)	A pool of space from one or more PVs	vgcreate, vgextend, vgs
Logical volume (LV)	A 'partition' cut from the pool	lvcreate, lvextend, lvs
File system	XFS or ext4 on the LV	mkfs.xfs, xfs_growfs
Why	Grow online, add disks, snapshots	RHEL installs on LVM by default

Build LVM Step by Step

```
[student@server1 ~]$ sudo pvcreate /dev/vdb2 /dev/vdc
[student@server1 ~]$ sudo vgcreate vgdata /dev/vdb2 /dev/vdc
[student@server1 ~]$ sudo lvcreate -n lvweb -L 3G vgdata
[student@server1 ~]$ sudo lvcreate -n lvlogs -l 50%FREE vgdata
[student@server1 ~]$ sudo mkfs.xfs /dev/vgdata/lvweb
[student@server1 ~]$ sudo mkdir /srv/web
[student@server1 ~]$ echo '/dev/vgdata/lvweb /srv/web xfs defaults 0 0' | sudo tee -a /etc/fstab
[student@server1 ~]$ sudo systemctl daemon-reload ; sudo mount -a
[student@server1 ~]$ sudo pvs ; sudo vgs ; sudo lvs
```

Grow a File System Online

```
[student@server1 ~]$ df -h /srv/web
/dev/mapper/vgdata-lvweb  3.0G  2.8G  200M  94% /srv/web
[student@server1 ~]$ sudo lvextend -r -L +1G /dev/vgdata/lvweb    # -r grows the FS too
  Size of logical volume vgdata/lvweb changed from 3.00 GiB to 4.00 GiB.
[student@server1 ~]$ df -h /srv/web
/dev/mapper/vgdata-lvweb  4.0G  2.8G  1.2G  70% /srv/web

# Pool full? Add a disk first:
[student@server1 ~]$ sudo vgextend vgdata /dev/vdd
# Without -r:  sudo xfs_growfs /srv/web   (ext4: sudo resize2fs /dev/vgdata/lvweb)
```

Swap Space

```
[student@server1 ~]$ swapon --show ; free -h
[student@server1 ~]$ sudo lvcreate -n lvswap -L 1G vgdata
[student@server1 ~]$ sudo mkswap /dev/vgdata/lvswap
[student@server1 ~]$ echo '/dev/vgdata/lvswap none swap defaults 0 0' | sudo tee -a /etc/fstab
[student@server1 ~]$ sudo systemctl daemon-reload ; sudo swapon -a
[student@server1 ~]$ swapon --show
NAME       TYPE      SIZE USED PRIO
/dev/dm-1  partition   3G   0B   -2
/dev/dm-4  partition   1G   0B   -3
```

Stratis and LUKS Encryption

Tool	What it does	Example
Stratis	Pool-based storage on XFS: thin provisioning, snapshots	stratis pool create pool1 /dev/vdd
Stratis FS	File systems grow automatically inside the pool	stratis fs create pool1 docs
Stratis mount	fstab needs x-systemd.requires=stratisd.service	see man stratis; mount -a to test
LUKS	Encrypts a whole partition or LV	cryptsetup luksFormat /dev/vdb1
Open LUKS	Unlock, then mkfs and mount the mapper device	cryptsetup open /dev/vdb1 secret
At boot	/etc/crypttab, or Clevis with TPM 2.0 / Tang	Laptops: encrypt during installation

Summary and Homework

- lsblk, blkid, df -hT show disks, UUIDs and space
- parted mklabel gpt / mkpart; mkfs.xfs; mount
- fstab with UUIDs; always test with mount -a
- LVM: pvcreate → vgcreate → lvcreate; lvextend -r grows online
- mkswap + swapon; Stratis pools; LUKS encryption
- Homework: build vgdata from vdb2 + vdc, create lvweb, mount at /srv/web, then grow it by 1 GB

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

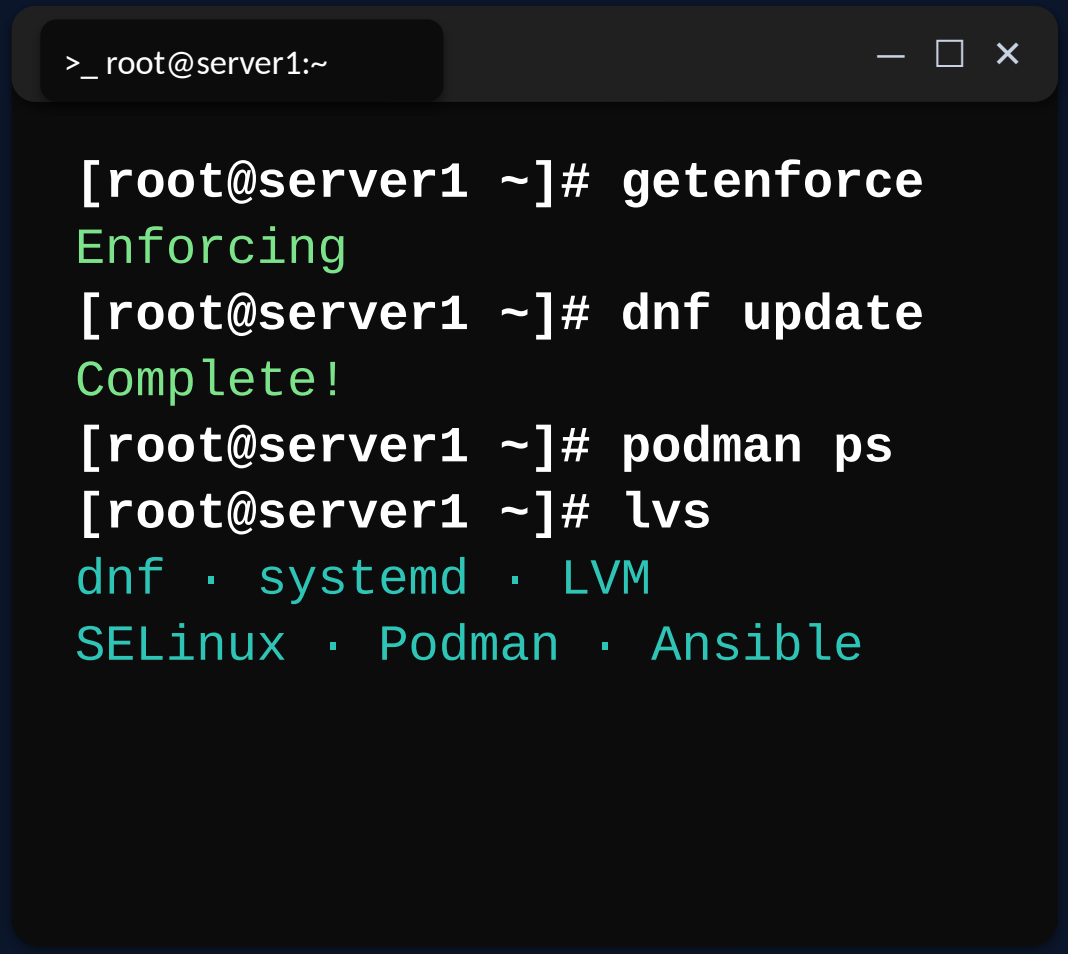
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 8 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Networking, firewalld & Secure SSH

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 8

- Network basics: interfaces, IP, gateway, DNS
- Checking the network: ip, ss, ping, dig
- NetworkManager with nmcli: static IP, DNS, hostname
- Name resolution: /etc/hosts and resolv.conf
- firewalld: zones, services and ports
- SSH: keys, secure configuration, copying files
- Troubleshooting step by step

Check the Network

```
[student@server1 ~]$ ip -br addr
lo        UNKNOWN  127.0.0.1/8 ::1/128
enp1s0    UP       10.0.0.11/24 fe80::5054:ff:fe12:3456/64
[student@server1 ~]$ ip route                    # default via 10.0.0.1
[student@server1 ~]$ ping -c 3 10.0.0.1          # gateway reachable?
[student@server1 ~]$ dig +short redhat.com        # DNS working?
[student@server1 ~]$ ss -tulpn                   # listening ports + programs
[student@server1 ~]$ tracepath redhat.com
[student@server1 ~]$ curl -I https://www.mywebuniversity.com
```

NetworkManager Concepts

Term	Meaning
Device	A network interface, e.g. enp1s0
Connection (profile)	Settings for a device: IP, gateway, DNS, autoconnect
Files	/etc/NetworkManager/system-connections/*.nmconnection
Tools	nmcli (command line), nmtui (text menus), Cockpit, GNOME
RHEL 9 and 10	Keyfile format; old ifcfg files are no longer used in RHEL 10
Rule	Never edit while NetworkManager runs — use nmcli

Static IP with nmcli

```
[student@server1 ~]$ nmcli device status
[student@server1 ~]$ nmcli connection show
NAME    UUID        TYPE      DEVICE
enp1s0  9c2e...     ethernet  enp1s0
[student@server1 ~]$ sudo nmcli con mod enp1s0 ipv4.method manual \
      ipv4.addresses 10.0.0.11/24 ipv4.gateway 10.0.0.1 \
      ipv4.dns '10.0.0.1 1.1.1.1' ipv4.dns-search lab.local
[student@server1 ~]$ sudo nmcli con up enp1s0
[student@server1 ~]$ sudo nmcli con add con-name lab2 type ethernet ifname enp2s0 \
      ipv4.method manual ipv4.addresses 192.168.50.11/24
[student@server1 ~]$ sudo hostnamectl set-hostname server1.lab.local
```

Name Resolution

```
[student@server1 ~]$ cat /etc/hosts
127.0.0.1   localhost
10.0.0.11   server1.lab.local server1
10.0.0.12   server2.lab.local server2
[student@server1 ~]$ cat /etc/resolv.conf      # written by NetworkManager
search lab.local
nameserver 10.0.0.1
[student@server1 ~]$ grep ^hosts /etc/nsswitch.conf
hosts:      files dns myhostname
[student@server1 ~]$ getent hosts server2      # uses the same order as programs
```

firewalld: Zones, Services, Ports

Concept	Meaning
Zone	A trust level for a network; public is the default
Service	A named set of ports, e.g. http = 80/tcp, ssh = 22/tcp
Port	A single port/protocol, e.g. 8080/tcp
Runtime vs permanent	Runtime is lost at reload; --permanent is saved
Rich rules	Allow or deny by source address, logging, limits
Backend	nftables in RHEL 9 and 10

Using firewall-cmd

```
[student@server1 ~]$ sudo firewall-cmd --state ; sudo firewall-cmd --get-active-zones
[student@server1 ~]$ sudo firewall-cmd --list-all
public (active)
  interfaces: enp1s0
  services: cockpit dhcpv6-client ssh
[student@server1 ~]$ sudo firewall-cmd --permanent --add-service={http,https}
[student@server1 ~]$ sudo firewall-cmd --permanent --add-port=8080/tcp
[student@server1 ~]$ sudo firewall-cmd --permanent --remove-service=dhcpv6-client
[student@server1 ~]$ sudo firewall-cmd --permanent --add-rich-rule=\
      'rule family=ipv4 source address=10.0.0.0/24 service name=mysql accept'
[student@server1 ~]$ sudo firewall-cmd --reload ; sudo firewall-cmd --list-services
```

SSH Keys

```
[student@server1 ~]$ ssh-keygen -t ed25519 -C 'student@server1'
Your public key has been saved in /home/student/.ssh/id_ed25519.pub
[student@server1 ~]$ ssh-copy-id student@server2
[student@server1 ~]$ ssh student@server2              # no password now
[student@server1 ~]$ ssh server2 'uptime; df -h /'    # run a remote command
[student@server1 ~]$ scp report.txt server2:/tmp/
[student@server1 ~]$ rsync -avz projects/ server2:projects/
[student@server1 ~]$ sftp server2
```

A Secure SSH Server

```
[student@server2 ~]$ sudo vim /etc/ssh/sshd_config.d/50-security.conf
PermitRootLogin no
PasswordAuthentication no          # keys only (test your key first!)
MaxAuthTries 3
AllowGroups wheel sshusers
[student@server2 ~]$ sudo sshd -t                    # test the syntax
[student@server2 ~]$ sudo systemctl reload sshd

# A different port needs SELinux and firewall changes (Lesson 9):
[student@server2 ~]$ sudo semanage port -a -t ssh_port_t -p tcp 2222
```

Troubleshooting the Network

Step	Check	Command
1	Is the link up and the address right?	ip -br addr; nmcli device status
2	Can I reach the gateway?	ping -c3 10.0.0.1; ip route
3	Does DNS work?	dig redhat.com; getent hosts server2
4	Is the service listening?	ss -tlnp | grep :80; systemctl status httpd
5	Is the firewall open?	firewall-cmd --list-all
6	Is SELinux blocking it?	ausearch -m AVC -ts recent (Lesson 9)
7	What do the logs say?	journalctl -u NetworkManager -u httpd

Summary and Homework

- ip, ss, ping, dig, getent: look before you change
- nmcli con mod … ipv4.method manual; nmcli con up
- /etc/hosts → DNS (order in nsswitch.conf)
- firewall-cmd --permanent --add-service, then --reload
- ssh-keygen, ssh-copy-id; PermitRootLogin no
- Homework: static IPs on both servers, open http on server1, key-only SSH to server2

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

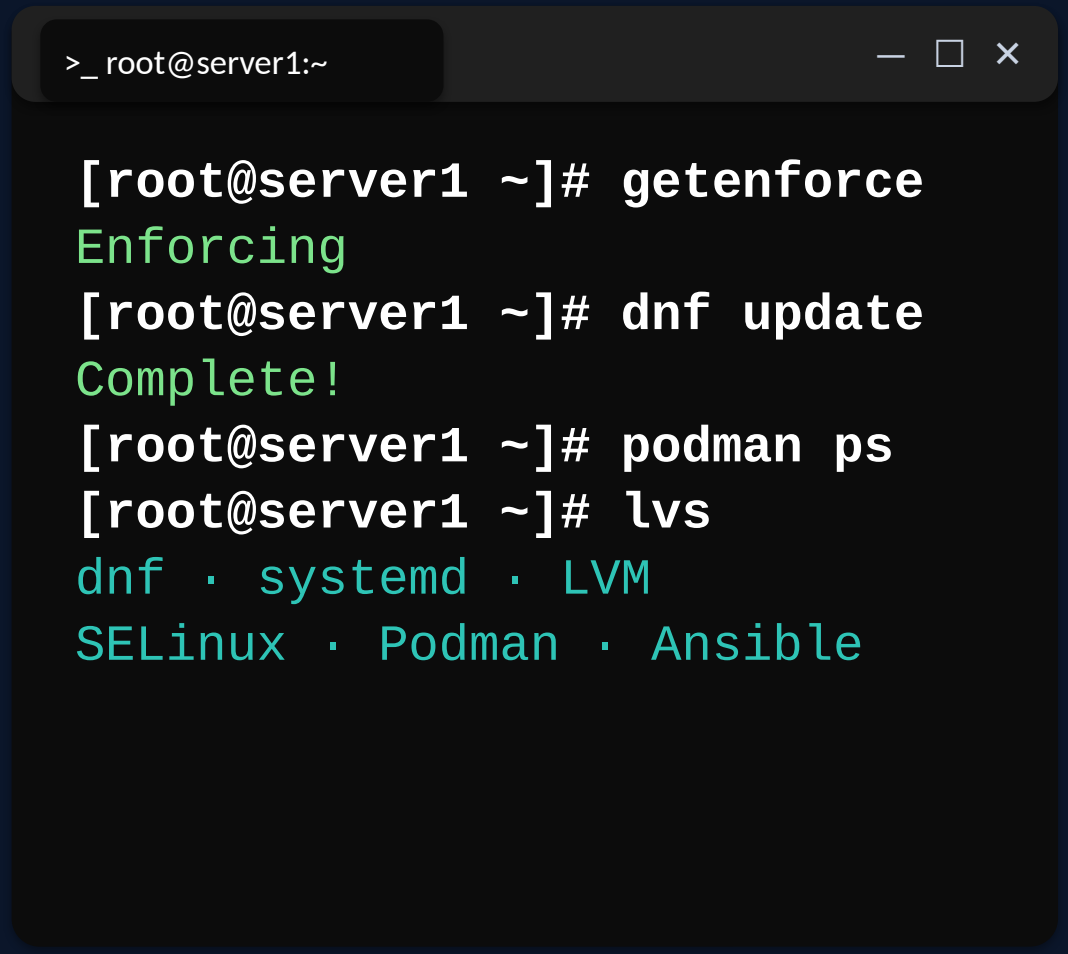
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 9 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

SELinux: Modes, Contexts, Booleans & Troubleshooting

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 9

- Why SELinux: mandatory access control
- Modes: enforcing, permissive, disabled
- Contexts: user, role, type, level
- Fixing file contexts: semanage fcontext and restorecon
- Booleans: switches for common needs
- Ports: semanage port
- Troubleshooting AVC denials: ausearch, sealert, audit2why

Why SELinux?

	Normal permissions (DAC)	SELinux (MAC)
Who decides	The owner of the file	A central policy from Red Hat
Based on	User, group, rwx	Labels (types) on processes and files
root	Bypasses everything	Is also limited by the policy
If a web server is hacked	Attacker gets all of apache's file access	Attacker is confined to web content types
Default in RHEL	Always on	Enforcing, targeted policy

Modes

```
[student@server1 ~]$ getenforce
Enforcing
[student@server1 ~]$ sestatus
SELinux status:                 enabled
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
[student@server1 ~]$ sudo setenforce 0      # permissive until reboot (testing only)
[student@server1 ~]$ sudo setenforce 1      # back to enforcing
[student@server1 ~]$ grep ^SELINUX= /etc/selinux/config
SELINUX=enforcing
```

Contexts: Labels on Everything

```
[student@server1 ~]$ ls -Z /var/www/html/index.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/index.html
[student@server1 ~]$ ls -Zd /home/student
unconfined_u:object_r:user_home_dir_t:s0 /home/student
[student@server1 ~]$ ps -eZ | grep httpd
system_u:system_r:httpd_t:s0     1234 ?  00:00:01 httpd
[student@server1 ~]$ id -Z
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
# user : role : TYPE : level   — in the targeted policy, the TYPE matters most
```

The Classic Problem: Moved Files

```
[student@server1 ~]$ echo 'Hello from MyWebUniversity' > index.html
[student@server1 ~]$ sudo mv index.html /var/www/html/     # mv keeps the OLD label
[student@server1 ~]$ ls -Z /var/www/html/index.html
unconfined_u:object_r:user_home_t:s0 /var/www/html/index.html
[student@server1 ~]$ curl http://localhost/
<title>403 Forbidden</title>
[student@server1 ~]$ sudo restorecon -Rv /var/www/html
Relabeled /var/www/html/index.html from ...user_home_t:s0 to ...httpd_sys_content_t:s0
[student@server1 ~]$ curl http://localhost/
Hello from MyWebUniversity
```

Custom Locations: semanage fcontext

```
# Serve a web site from /srv/web (our LVM volume from Lesson 7)
[student@server1 ~]$ sudo semanage fcontext -a -t httpd_sys_content_t '/srv/web(/.*)?'
[student@server1 ~]$ sudo restorecon -Rv /srv/web
[student@server1 ~]$ ls -Zd /srv/web
system_u:object_r:httpd_sys_content_t:s0 /srv/web
[student@server1 ~]$ sudo semanage fcontext -l -C       # your local rules
[student@server1 ~]$ sudo semanage fcontext -l | grep '/var/www'
# chcon changes a label temporarily — a relabel or restorecon will undo it
```

Booleans: Policy Switches

```
[student@server1 ~]$ getsebool -a | grep httpd | head
httpd_can_network_connect --> off
httpd_can_sendmail --> off
httpd_enable_homedirs --> off
[student@server1 ~]$ sudo semanage boolean -l | grep httpd_can_network_connect
httpd_can_network_connect (off, off)  Allow httpd to can network connect
[student@server1 ~]$ sudo setsebool -P httpd_can_network_connect on   # -P = permanent
[student@server1 ~]$ sudo setsebool -P httpd_enable_homedirs on
[student@server1 ~]$ sudo semanage boolean -l -C        # booleans you changed
```

Ports: semanage port

```
[student@server1 ~]$ sudo semanage port -l | grep -w http_port_t
http_port_t     tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
# Run Apache on port 8888:
[student@server1 ~]$ sudo semanage port -a -t http_port_t -p tcp 8888
[student@server1 ~]$ sudo firewall-cmd --permanent --add-port=8888/tcp ; sudo firewall-cmd --reload
[student@server1 ~]$ sudo systemctl restart httpd

# SSH on port 2222 (from Lesson 8):
[student@server2 ~]$ sudo semanage port -a -t ssh_port_t -p tcp 2222
[student@server2 ~]$ sudo semanage port -l -C
```

Troubleshooting Denials

```
[student@server1 ~]$ sudo ausearch -m AVC -ts recent
type=AVC msg=audit(...): avc:  denied  { read } for  pid=1234 comm="httpd"
  name="index.html" scontext=system_u:system_r:httpd_t:s0
  tcontext=unconfined_u:object_r:user_home_t:s0 tclass=file permissive=0
[student@server1 ~]$ sudo ausearch -m AVC -ts recent | audit2why
[student@server1 ~]$ sudo dnf install -y setroubleshoot-server
[student@server1 ~]$ sudo journalctl -t setroubleshoot --since '10 min ago'
[student@server1 ~]$ sudo sealert -a /var/log/audit/audit.log
... If you want to fix the label: # /sbin/restorecon -v /var/www/html/index.html
```

A Calm SELinux Method

Step	Do this
1	Confirm: does it work in permissive mode? (setenforce 0, test, setenforce 1)
2	Read the denial: ausearch -m AVC -ts recent, or sealert
3	Wrong file label? semanage fcontext -a + restorecon -Rv
4	Feature switch? setsebool -P <boolean> on
5	Non-standard port? semanage port -a -t <type> -p tcp <port>
6	Test again in enforcing mode; never leave it permissive or disabled
Remember	After a root password reset: touch /.autorelabel (Lesson 6)

Summary and Homework

- SELinux = labels + policy; keep it Enforcing
- ls -Z, ps -Z: the TYPE is what matters
- Wrong label → semanage fcontext -a + restorecon -Rv
- Booleans → setsebool -P; ports → semanage port -a
- Denials → ausearch -m AVC, sealert, audit2why
- Homework: serve /srv/web with Apache on port 8888 — fix the label, port and firewall, in enforcing mode

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

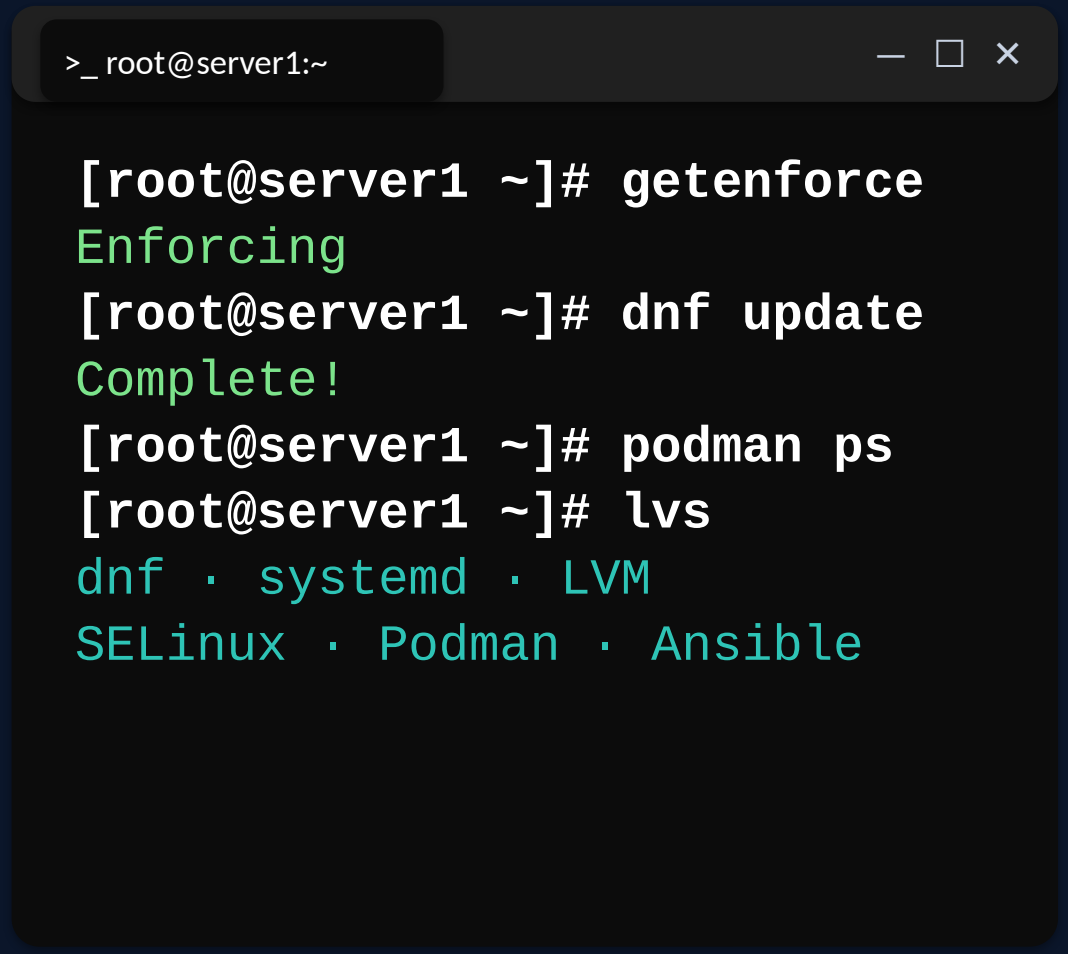
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 10 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Services: Apache, NFS, autofs & Podman Containers

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 10

- Apache (httpd): install, configure, test
- Virtual hosts and HTTPS with mod_ssl
- NFS server on server2; NFS client on server1
- autofs: mount on demand
- Podman: rootless containers, images, volumes
- Running containers as systemd services with Quadlet
- Databases from AppStream: MariaDB and PostgreSQL

Apache Web Server

```
[student@server1 ~]$ sudo dnf install -y httpd
[student@server1 ~]$ sudo systemctl enable --now httpd
[student@server1 ~]$ sudo firewall-cmd --permanent --add-service={http,https}
[student@server1 ~]$ sudo firewall-cmd --reload
[student@server1 ~]$ echo '<h1>Welcome to MyWebUniversity lab</h1>' | \
      sudo tee /var/www/html/index.html
[student@server1 ~]$ curl http://server1.lab.local
<h1>Welcome to MyWebUniversity lab</h1>
# Config: /etc/httpd/conf/httpd.conf  +  /etc/httpd/conf.d/*.conf
# Logs:   /var/log/httpd/access_log, error_log
[student@server1 ~]$ sudo apachectl configtest      # Syntax OK
```

Virtual Hosts

```
# /etc/httpd/conf.d/courses.conf
<VirtualHost *:80>
    ServerName courses.lab.local
    DocumentRoot /srv/web/courses
    ErrorLog /var/log/httpd/courses-error.log
    CustomLog /var/log/httpd/courses-access.log combined
</VirtualHost>
<Directory /srv/web/courses>
    Require all granted
</Directory>
[student@server1 ~]$ sudo mkdir -p /srv/web/courses ; sudo restorecon -Rv /srv/web
[student@server1 ~]$ sudo apachectl configtest && sudo systemctl reload httpd
```

HTTPS with mod_ssl

```
[student@server1 ~]$ sudo dnf install -y mod_ssl
# Lab: a self-signed certificate (production: Let's Encrypt or your CA)
[student@server1 ~]$ sudo openssl req -x509 -newkey rsa:3072 -nodes -days 365 \
      -keyout /etc/pki/tls/private/courses.key -out /etc/pki/tls/certs/courses.crt \
      -subj '/CN=courses.lab.local'
<VirtualHost *:443>
    ServerName courses.lab.local
    DocumentRoot /srv/web/courses
    SSLEngine on
    SSLCertificateFile /etc/pki/tls/certs/courses.crt
    SSLCertificateKeyFile /etc/pki/tls/private/courses.key
</VirtualHost>
[student@server1 ~]$ curl -k https://courses.lab.local
```

NFS Server on server2

```
[student@server2 ~]$ sudo dnf install -y nfs-utils
[student@server2 ~]$ sudo mkdir -p /exports/shared ; sudo chmod 2775 /exports/shared
[student@server2 ~]$ sudo vim /etc/exports
/exports/shared   10.0.0.0/24(rw,sync,root_squash)
[student@server2 ~]$ sudo systemctl enable --now nfs-server
[student@server2 ~]$ sudo exportfs -rav
exporting 10.0.0.0/24:/exports/shared
[student@server2 ~]$ sudo firewall-cmd --permanent --add-service={nfs,rpc-bind,mountd}
[student@server2 ~]$ sudo firewall-cmd --reload
```

NFS Client and autofs on server1

```
[student@server1 ~]$ showmount -e server2
/exports/shared 10.0.0.0/24
[student@server1 ~]$ sudo mkdir /mnt/shared
[student@server1 ~]$ sudo mount -t nfs server2:/exports/shared /mnt/shared
# Permanent (fstab):  server2:/exports/shared  /mnt/shared  nfs  defaults,_netdev  0 0

[student@server1 ~]$ sudo dnf install -y autofs
[student@server1 ~]$ echo '/nfs  /etc/auto.nfs' | sudo tee /etc/auto.master.d/nfs.autofs
[student@server1 ~]$ echo 'shared  -rw,sync  server2:/exports/shared' | sudo tee /etc/auto.nfs
[student@server1 ~]$ sudo systemctl enable --now autofs
[student@server1 ~]$ ls /nfs/shared        # mounted on first access
```

Podman: Containers on RHEL

Topic	Details
What	Runs OCI containers; Docker-compatible commands
Daemonless	No background daemon; each container is a normal process
Rootless	Normal users run containers safely
Images	registry.redhat.io UBI images (free to redistribute), quay.io, docker.io
Tools	podman, buildah (build), skopeo (copy, inspect)
Security	SELinux labels every container; :Z on volumes

Running Containers

```
[student@server1 ~]$ sudo dnf install -y container-tools
[student@server1 ~]$ podman search --limit 3 ubi10
[student@server1 ~]$ podman pull registry.access.redhat.com/ubi10/ubi
[student@server1 ~]$ podman run -it --rm ubi10/ubi cat /etc/redhat-release
[student@server1 ~]$ mkdir -p ~/site ; echo 'Hello from a container' > ~/site/index.html
[student@server1 ~]$ podman run -d --name web -p 8080:8080 \
      -v ~/site:/var/www/html:Z registry.access.redhat.com/ubi10/httpd-24
[student@server1 ~]$ podman ps ; curl http://localhost:8080
[student@server1 ~]$ podman logs web ; podman exec -it web bash
[student@server1 ~]$ podman stop web ; podman rm web ; podman images
```

Containers as Services: Quadlet

```
[student@server1 ~]$ mkdir -p ~/.config/containers/systemd
[student@server1 ~]$ vim ~/.config/containers/systemd/web.container
[Container]
Image=registry.access.redhat.com/ubi10/httpd-24
PublishPort=8080:8080
Volume=%h/site:/var/www/html:Z
[Install]
WantedBy=default.target
[student@server1 ~]$ systemctl --user daemon-reload
[student@server1 ~]$ systemctl --user start web
[student@server1 ~]$ sudo loginctl enable-linger student     # run without a login
```

Databases from AppStream

Database	Install and start	Notes
MariaDB	dnf install mariadb-server; systemctl enable --now mariadb	Run mariadb-secure-installation
MySQL	dnf install mysql-server; systemctl enable --now mysqld	MySQL 8.4 LTS
PostgreSQL	dnf install postgresql-server; postgresql-setup --initdb	Newer versions as postgresql18 etc.
Firewall	firewall-cmd --add-service=mysql / postgresql	Open only to app servers
SELinux	Default ports and paths are already labeled	Custom data directory? semanage fcontext
Containers	podman run … rhel10/postgresql-16	Quick for development

Summary and Homework

- httpd + conf.d virtual hosts; apachectl configtest; mod_ssl for HTTPS
- NFS: /etc/exports + exportfs -rav; firewall nfs, rpc-bind, mountd
- Client: mount -t nfs, fstab _netdev, or autofs on demand
- Podman: rootless, :Z volumes, UBI images; Quadlet for services
- Databases from AppStream; open ports carefully
- Homework: courses.lab.local over HTTPS on server1; NFS share from server2 via autofs; a Quadlet web container

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

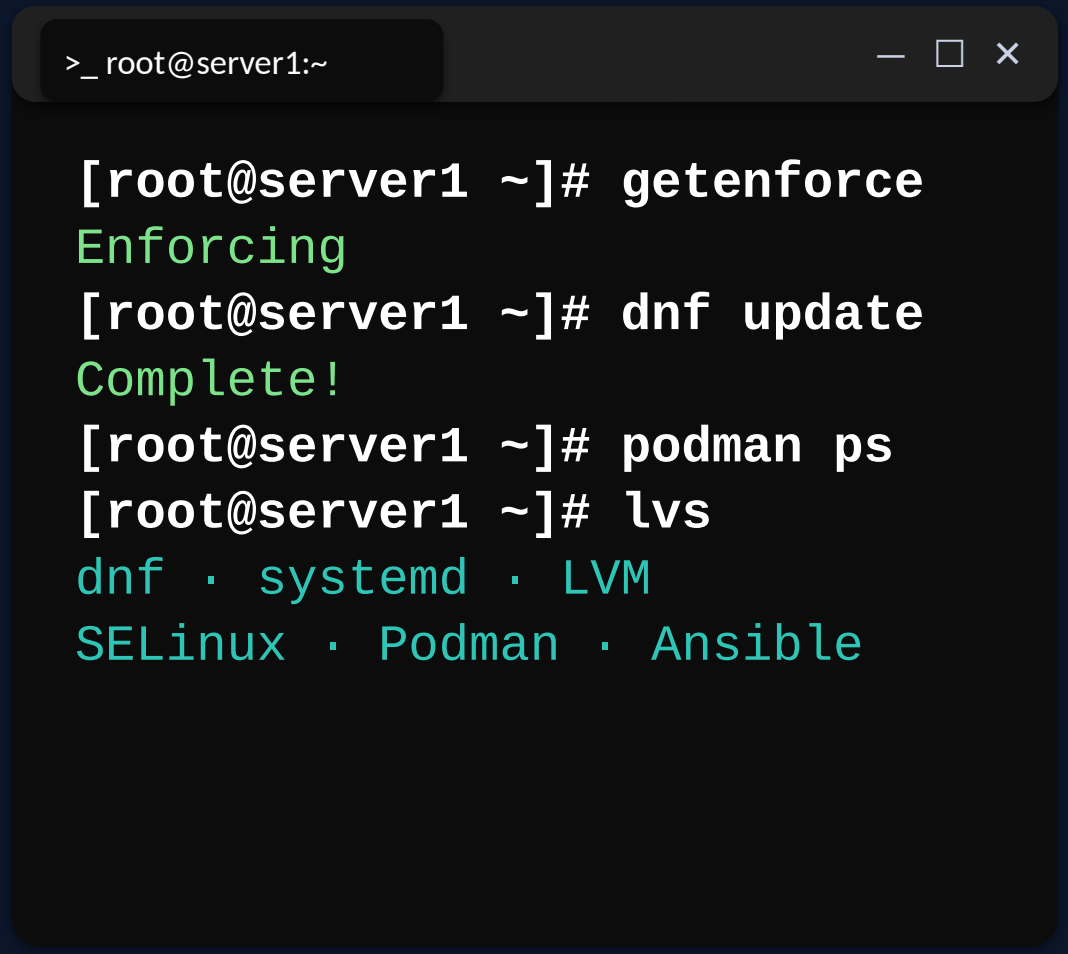
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 11 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Security, Updates, Time, Backup & Upgrading 9 to 10

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 11

- A hardening checklist for every server
- System-wide crypto policies and post-quantum crypto
- Compliance scanning with OpenSCAP; fapolicyd
- Accurate time with chrony
- Patch management and Red Hat Lightspeed (Insights)
- Backup and restore: tar, rsync, ReaR
- In-place upgrade RHEL 9 → 10 with Leapp

Hardening Checklist

Area	Do this
Software	Install only what you need (Minimal / Server); remove the rest
Accounts	Lock root; sudo + wheel; strong pwquality; remove old users
SSH	Keys only; PermitRootLogin no; AllowGroups (Lesson 8)
Network	firewalld default zone with only needed services (ss -tulpn)
SELinux	Enforcing, always (Lesson 9)
Updates and logs	Security updates weekly; persistent journal; central logging
Storage	LUKS for laptops and sensitive data; separate /var/log if possible

System-Wide Crypto Policies

```
[student@server1 ~]$ update-crypto-policies --show
DEFAULT
[student@server1 ~]$ sudo update-crypto-policies --set FUTURE    # stricter
[student@server1 ~]$ sudo update-crypto-policies --set DEFAULT:NO-SHA1
[student@server1 ~]$ ls /usr/share/crypto-policies/policies/
DEFAULT.pol  FIPS.pol  FUTURE.pol  LEGACY.pol ...
# Applies to OpenSSL, GnuTLS, NSS, OpenSSH, Kerberos, Java ...
# RHEL 10: post-quantum algorithms (ML-KEM, ML-DSA) enabled in TLS and SSH
[student@server1 ~]$ sudo fips-mode-setup --check    # RHEL 9; RHEL 10: enable FIPS at install
```

Compliance Scanning with OpenSCAP

```
[student@server1 ~]$ sudo dnf install -y openscap-scanner scap-security-guide
[student@server1 ~]$ oscap info /usr/share/xml/scap/ssg/content/ssg-rhel10-ds.xml | grep Id:
  Id: xccdf_org.ssgproject.content_profile_cis
  Id: xccdf_org.ssgproject.content_profile_stig
[student@server1 ~]$ sudo oscap xccdf eval --profile cis \
      --report /tmp/cis-report.html \
      /usr/share/xml/scap/ssg/content/ssg-rhel10-ds.xml
[student@server1 ~]$ firefox /tmp/cis-report.html        # pass / fail for each rule
# Remediate: --remediate, or generate an Ansible playbook with oscap xccdf generate fix
```

More Security Tools

Tool	What it does
fapolicyd	Allows only trusted programs to run (application allow-listing)
AIDE	Detects changed files: aide --init, then aide --check
auditd	Records security events: ausearch, aureport, audit rules
usbguard	Blocks unknown USB devices
Lightspeed (Insights)	Red Hat's analysis: vulnerabilities, advisories, risks
MCP server for RHEL (10.3)	Lets an AI assistant read system data for diagnostics, with controlled access

Accurate Time with chrony

```
[student@server1 ~]$ timedatectl
               Local time: Fri 2026-11-20 10:15:02 PST
System clock synchronized: yes
              NTP service: active
[student@server1 ~]$ chronyc sources -v
^* time.cloudflare.com     3   6   377    35   +214us[ +301us] +/-   12ms
[student@server1 ~]$ sudo vim /etc/chrony.conf
server 10.0.0.1 iburst              # your own time source
pool 2.rhel.pool.ntp.org iburst
[student@server1 ~]$ sudo systemctl restart chronyd ; chronyc tracking
```

Patch Management

Practice	How
Know what is missing	dnf updateinfo summary; Lightspeed vulnerability service
Test first	Patch test servers, wait, then production
Schedule	Security weekly; full updates monthly; maintenance windows
Automate	dnf-automatic, Ansible, or Red Hat Satellite
Reboot when needed	dnf needs-restarting -r; kernel live patching (kpatch) for critical servers
Roll back	dnf history undo; LVM or VM snapshots before big changes

Backup and Restore

```
# Files: tar with ACLs and SELinux labels
[student@server1 ~]$ sudo tar --xattrs --acls --selinux -czpf /backup/etc-$(date +%F).tgz /etc
[student@server1 ~]$ sudo tar --xattrs --acls --selinux -xzpf /backup/etc-2026-11-20.tgz -C /restore

# Copy to another server, only the changes:
[student@server1 ~]$ sudo rsync -aAXv --delete /srv/web/ backup@server2:/backup/web/

# Disaster recovery: Relax-and-Recover (ReaR) makes a bootable rescue image
[student@server1 ~]$ sudo dnf install -y rear ; sudo rear -v mkbackup
# Rule: a backup is only real after you have TESTED a restore
```

In-Place Upgrade: RHEL 9.9 → 10.3 with Leapp

```
[student@server2 ~]$ sudo dnf update -y && sudo reboot          # be on the latest 9.9
[student@server2 ~]$ sudo dnf install -y leapp-upgrade
[student@server2 ~]$ sudo leapp preupgrade --target 10.3        # report only, no changes
[student@server2 ~]$ sudo less /var/log/leapp/leapp-report.txt
Risk Factor: high (inhibitor)  ... fix every inhibitor, then run again
[student@server2 ~]$ sudo leapp upgrade --target 10.3
[student@server2 ~]$ sudo reboot                                # upgrade runs during boot
[student@server2 ~]$ cat /etc/redhat-release
Red Hat Enterprise Linux release 10.3 (Coughlan)
```

Summary and Homework

- Hardening: minimal software, keys, firewall, SELinux, updates
- update-crypto-policies; OpenSCAP scans against CIS / STIG
- chrony for accurate time: chronyc sources, tracking
- Patch with a plan: test, schedule, automate, roll back
- tar --acls --selinux, rsync -aAX, ReaR; test your restores
- Homework: run an OpenSCAP CIS report; upgrade server2 from 9.9 to 10.3 with Leapp (after a snapshot)

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

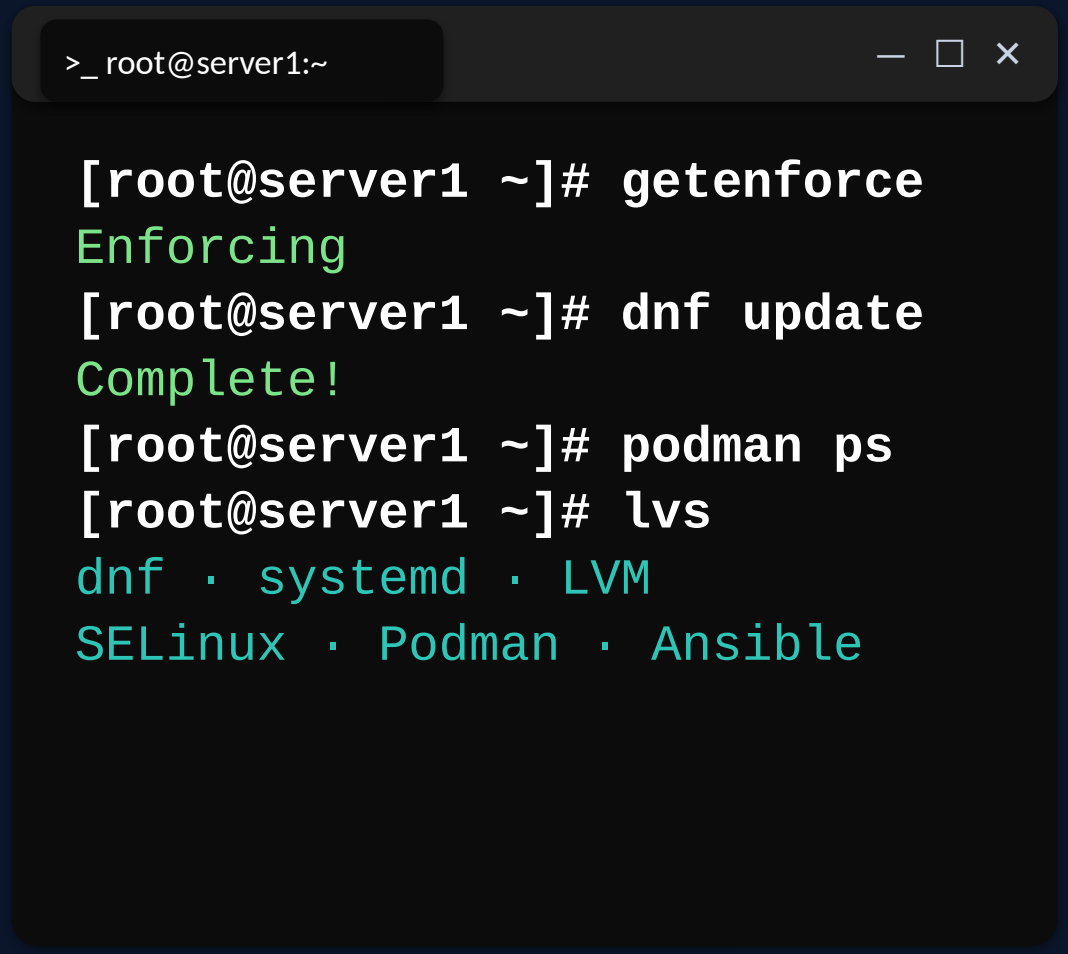
Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.



LESSON 12 OF 12

Red Hat Enterprise Linux

RHEL 9 & 10 · Admin · SELinux

Scripting, Ansible, Image Mode, Troubleshooting & RHCSA

Full Course · RHEL 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 12

- Bash scripting: variables, conditions, loops, exit codes
- A real script: a server health report
- Ansible: inventory, ad hoc commands, playbooks
- RHEL system roles
- Image mode (bootc): RHEL as a container image
- A troubleshooting method from boot to application
- The RHCSA and RHCE exams, and what to learn next

Bash Scripting Basics

```
#!/bin/bash
# /usr/local/bin/adduser-batch.sh — create users from a file
set -euo pipefail
file=${1:?usage: $0 users.txt}
if [[ ! -r $file ]]; then
    echo "Cannot read $file" >&2 ; exit 1
fi
while read -r user group; do
    if id "$user" &>/dev/null; then
        echo "skip: $user exists"
    else
        useradd -G "$group" "$user" && echo "added: $user ($group)"
    fi
done < "$file"
```

Script Building Blocks

Element	Example
Variables	name=web01 ; echo "Host: $name" ; today=$(date +%F)
Arguments	$1 $2 … ; $# count ; $@ all ; $0 script name
Exit codes	$? — 0 = success; anything else = failure
Tests	[[-f file]] [[-d dir]] [[$a == $b]] ((n > 5))
Loops	for h in server1 server2; do ssh $h uptime; done
Functions	log() { echo "$(date +%T) $*"; }
Check scripts	bash -n script.sh ; shellcheck script.sh (EPEL)

A Real Script: Server Health Report

```
#!/bin/bash
# /usr/local/bin/health.sh — run daily from cron or a systemd timer
echo "== $(hostname) $(date '+%F %T') =="
echo "Uptime: $(uptime -p)"
echo "Load:   $(cut -d' ' -f1-3 /proc/loadavg)"
df -h --output=target,pcent -x tmpfs | awk 'NR>1 && $2+0 > 80 {print "DISK WARNING:", $0}'
free -m | awk '/Mem/ {printf "Memory used: %d%%\n", $3*100/$2}'
systemctl --failed --no-legend | sed 's/^/FAILED: /'
echo "SELinux: $(getenforce)   Updates waiting: $(dnf -q check-update | grep -c .)"
journalctl -p err --since yesterday --no-pager | tail -5
```

Ansible: Inventory and Ad Hoc Commands

```
[student@server1 ~]$ sudo dnf install -y ansible-core rhel-system-roles
[student@server1 ~]$ mkdir ~/ansible && cd ~/ansible
[student@server1 ansible]$ cat inventory
[web]
server1.lab.local
[files]
server2.lab.local
[student@server1 ansible]$ cat ansible.cfg
[defaults]
inventory = ./inventory
[privilege_escalation]
become = true
[student@server1 ansible]$ ansible all -m ping
[student@server1 ansible]$ ansible files -m dnf -a 'name=htop state=present'
```

A Playbook

```
# web.yml
- name: Configure web servers
  hosts: web
  tasks:
    - name: Install Apache
      ansible.builtin.dnf: { name: httpd, state: latest }
    - name: Home page
      ansible.builtin.copy:
        content: "Managed by Ansible on {{ inventory_hostname }}\n"
        dest: /var/www/html/index.html
    - name: Start and enable
      ansible.builtin.service: { name: httpd, state: started, enabled: true }
    - name: Open firewall
      ansible.posix.firewalld: { service: http, permanent: true, immediate: true, state: enabled }
[student@server1 ansible]$ ansible-playbook web.yml
```

RHEL System Roles

Role	Configures
rhel-system-roles.timesync	chrony time sources
rhel-system-roles.firewall	firewalld zones, services, ports
rhel-system-roles.selinux	Modes, booleans, file contexts, ports
rhel-system-roles.storage	Partitions, LVM, file systems, mounts
rhel-system-roles.network	NetworkManager connections
rhel-system-roles.podman	Containers and Quadlet units
Docs and examples	/usr/share/doc/rhel-system-roles/

Image Mode: RHEL as a Container Image

```
# Containerfile
FROM registry.redhat.io/rhel10/rhel-bootc:10.3
RUN dnf -y install httpd && systemctl enable httpd
COPY index.html /var/www/html/

[student@server1 ~]$ podman build -t registry.lab.local/web-os:1.0 .
[student@server1 ~]$ podman push registry.lab.local/web-os:1.0
# Install: bootc-image-builder makes an ISO, qcow2 or cloud image from it
# On a running image-mode host:
[root@web01 ~]# bootc status
[root@web01 ~]# bootc upgrade       # pull the new image, applied at next reboot
[root@web01 ~]# bootc rollback      # go back to the previous image
```

A Troubleshooting Method

Layer	Ask	Tools
Boot	Does it boot? Which target?	GRUB menu, rd.break, journalctl -b -1
Storage	Full disk? Bad fstab line?	df -h, lsblk, mount -a, emergency mode
Network	Address, gateway, DNS?	ip, ping, dig, nmcli (Lesson 8 steps)
Service	Running? Enabled? Errors?	systemctl status, journalctl -u
Security	Firewall? SELinux? Permissions?	firewall-cmd, ausearch -m AVC, ls -lZ
Resources	CPU, memory, I/O?	top, free, vmstat, iostat (sysstat)
Change	What changed recently?	dnf history, journalctl, git log of configs

RHCSA, RHCE and Next Steps

Item	Details
RHCSA (EX200)	Hands-on exam: real tasks on live systems; this course covers its topics
RHCE (EX294)	Automation with Ansible; requires RHCSA
Practice	Do every homework twice; time yourself; use only man pages
Check before booking	Current exam version and objectives on redhat.com
Free learning	Red Hat Developer, docs.redhat.com, our videos and PDFs
Next courses	Rocky Linux 9/10, Fedora family, Debian/Ubuntu at MyWebUniversity

Course Summary

- Lessons 1–3: versions and lab, installation and Cockpit, the command line
- Lessons 4–6: users and permissions, dnf, systemd, boot and logs
- Lessons 7–9: storage and LVM, networking and firewalld, SELinux
- Lessons 10–11: Apache, NFS, Podman; security, backup, upgrading 9 to 10
- Lesson 12: scripting, Ansible, image mode, troubleshooting, RHCSA
- Final project: rebuild server1 from scratch with one Ansible playbook

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.