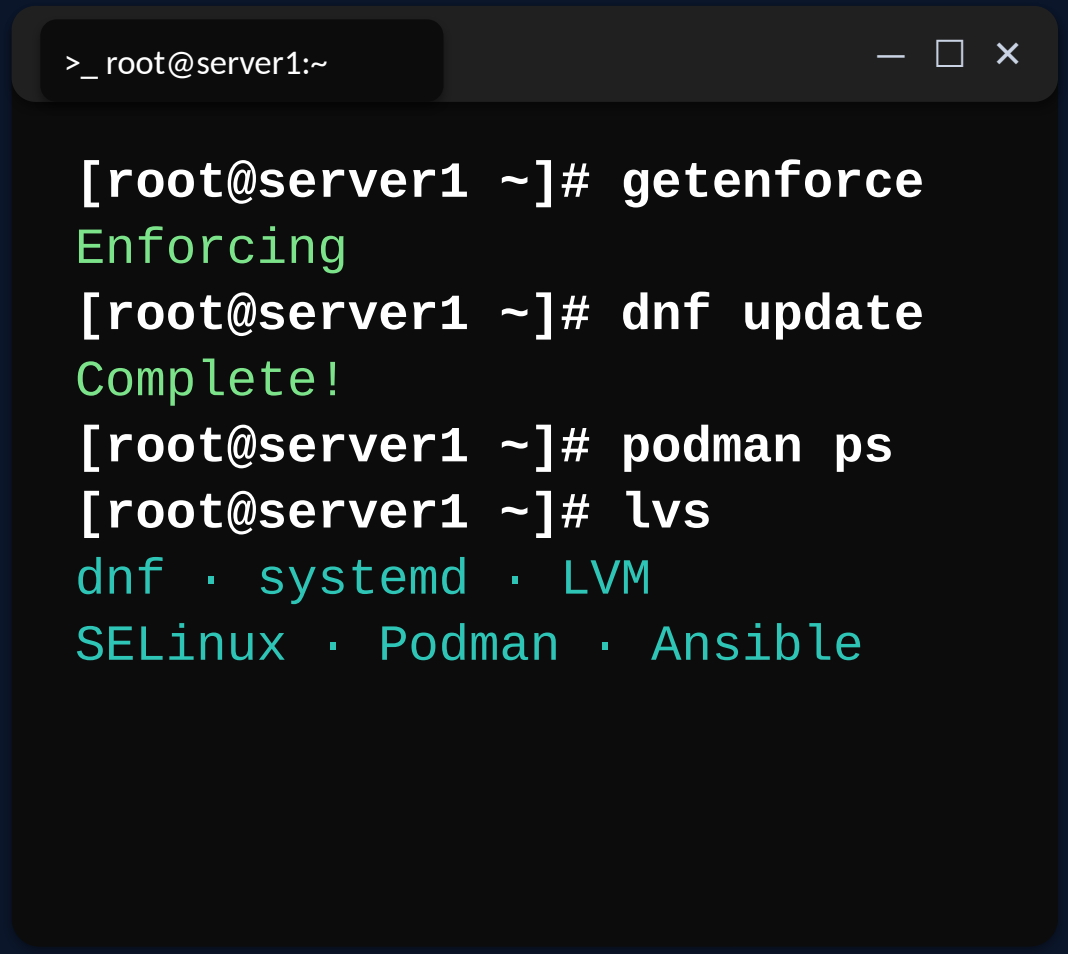


LESSON 10 OF 12

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.