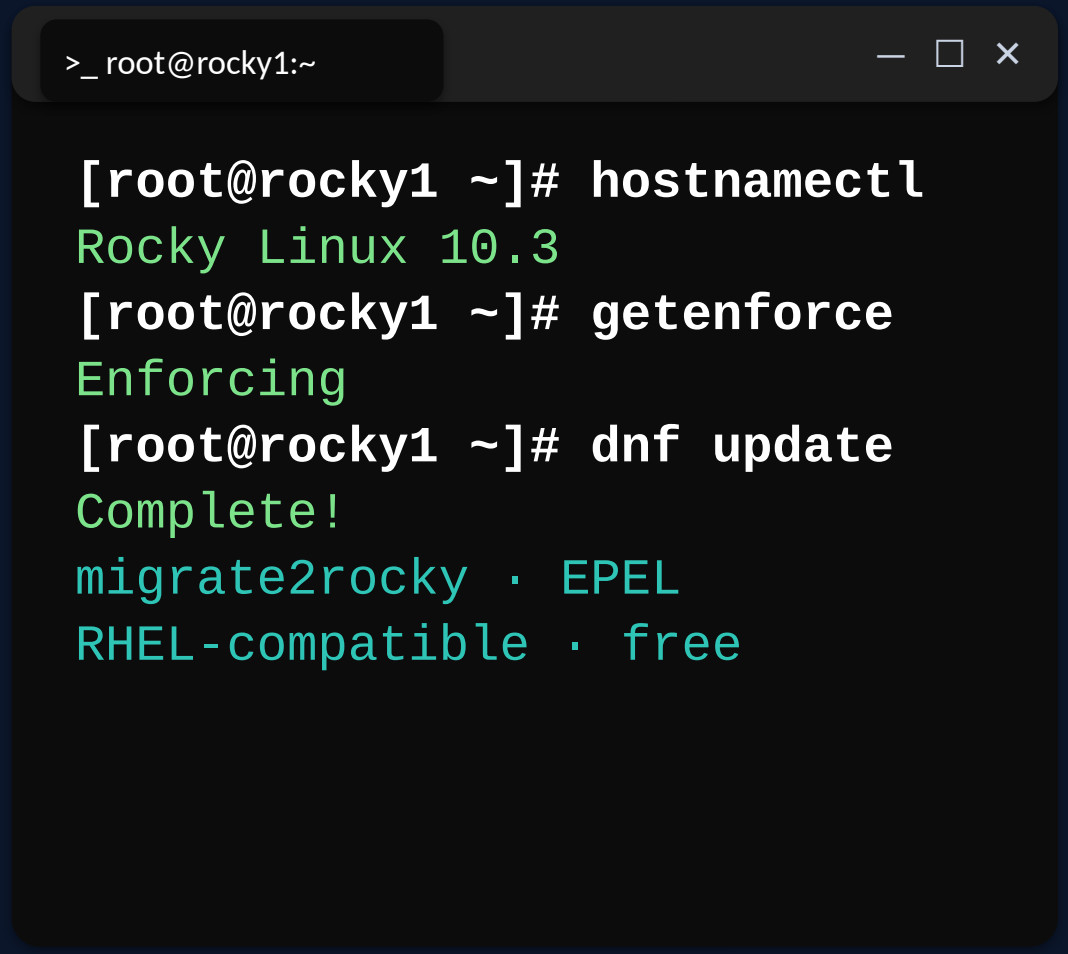


LESSON 10 OF 12

Rocky Linux 9 & 10

Install · Migrate · Differences

Rocky in Containers, the Cloud, WSL & Custom Images

Full Course · Rocky Linux 10.3 & 9.9 (Nov 2026)

What You Will Learn in Lesson 10

- Rocky container images: base, minimal, init
- Building your own images on Rocky
- Rocky in AWS, Azure and Google Cloud
- cloud-init: configuring instances at first boot
- Rocky on Windows with WSL
- Local test labs with Vagrant and qcow2 images
- Custom images with image builder

Rocky Container Images

Image	Use it for
quay.io/rockylinux/rockylinux:10	General base image with dnf
quay.io/rockylinux/rockylinux:10-minimal	Smaller; uses microdnf
quay.io/rockylinux/rockylinux:10-ubi-init	Runs systemd inside the container
docker.io/rockylinux/rockylinux:10	Same images on Docker Hub
Tags	:10, :10.3, :9, :9.9 — pin a minor version for reproducible builds
Free to redistribute	Ship images built on Rocky to anyone

Build Your Own Image on Rocky

```
# Containerfile
FROM quay.io/rockylinux/rockylinux:10-minimal
RUN microdnf -y install python3 python3-pip && microdnf clean all
WORKDIR /app
COPY app.py requirements.txt ./
RUN pip3 install --no-cache-dir -r requirements.txt
USER 1001
EXPOSE 8000
CMD ["python3", "app.py"]

[student@rocky1 ~]$ podman build -t myapp:1.0 .
[student@rocky1 ~]$ podman run -d -p 8000:8000 --name myapp myapp:1.0
[student@rocky1 ~]$ podman image inspect myapp:1.0 --format '{{.Size}}'
```

Rocky in the Public Cloud

Cloud	How to find official images
AWS	Marketplace: 'Rocky Linux' published by the Rocky Enterprise Software Foundation
Azure	Marketplace: Rocky Linux images from RESF; also via az vm image list
Google Cloud	Public images: project rocky-linux-cloud (gcloud compute images list)
Others	Oracle Cloud, DigitalOcean, Linode, Vultr, Hetzner and more
Default user	Usually 'rocky' with your SSH key; root login disabled
Tip	Check the publisher — use official RESF images only

Launch Rocky in the Cloud

```
# Google Cloud
$ gcloud compute images list --project rocky-linux-cloud --no-standard-images
$ gcloud compute instances create web1 --zone us-west1-b \
    --image-project rocky-linux-cloud --image-family <rocky-linux-10-family-from-list>

# Azure
$ az vm image list --publisher resf --all --output table

# AWS: find the AMI ID for your region in the Marketplace listing, then:
$ aws ec2 run-instances --image-id <rocky-10-ami-id> --instance-type t3.small --key-name mykey
$ ssh rocky@<public-ip>
```

cloud-init: Configure at First Boot

```
#cloud-config
hostname: web1
users:
  - name: student
    groups: wheel
    sudo: ALL=(ALL) NOPASSWD:ALL
    ssh_authorized_keys: ["ssh-ed25519 AAAA... student@laptop"]
package_upgrade: true
packages: [httpd, firewalld]
runcmd:
  - systemctl enable --now httpd firewalld
  - firewall-cmd --permanent --add-service=http && firewall-cmd --reload
  - echo 'Hello from cloud-init on Rocky' > /var/www/html/index.html
```

Rocky on Windows with WSL

```
# 1. Download the Rocky WSL image from rockylinux.org/download (Containers / WSL)
PS C:\> wsl --import Rocky10 C:\WSL\Rocky10 .\<Rocky-10-WSL-image-file>
PS C:\> wsl -d Rocky10
[root@PC ~]# dnf -y update ; dnf -y install sudo passwd vim
[root@PC ~]# useradd -G wheel student ; passwd student
[root@PC ~]# printf '[user]\ndefault=student\n' > /etc/wsl.conf
PS C:\> wsl --terminate Rocky10 ; wsl -d Rocky10      # now you are student
PS C:\> wsl -l -v
```

Local Labs: Vagrant and Cloud Images

Tool	How
Vagrant	vagrant init rockylinux/9 ; vagrant up ; vagrant ssh (check boxes for 10)
Generic cloud qcow2	Download Rocky-10-GenericCloud from the download page
KVM + cloud-init	virt-install --import --cloud-init user-data=web.yaml …
Cockpit Machines	Create VMs from the Rocky ISO or qcow2 in the browser
Why	Create, test and destroy servers in minutes
Tip	Snapshot before each lesson; restore to practice again

Custom Images with image builder

```
[student@rocky1 ~]$ sudo dnf install -y image-builder          # CLI, generally available in 10.3
[student@rocky1 ~]$ cat web.toml
name = "rocky-web"
version = "1.0.0"
[[packages]]
name = "httpd"
[customizations.services]
enabled = ["httpd"]
[[customizations.user]]
name = "student"
groups = ["wheel"]
[student@rocky1 ~]$ image-builder list | grep rocky        # distributions and image types
[student@rocky1 ~]$ sudo image-builder build qcow2 --blueprint web.toml
```

Summary and Homework

- Official images: quay.io/rockylinux/rockylinux:10 (also -minimal, -ubi-init)
- Clouds: official RESF images in AWS, Azure, Google Cloud and more
- cloud-init user data turns a plain image into a ready server
- WSL: wsl --import a Rocky image on Windows for learning
- Vagrant, qcow2 + KVM, and image builder for labs and golden images
- Homework: build myapp on Rocky minimal; boot a GenericCloud qcow2 with your cloud-init file

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.