Mise en Place d'un Serveur de Log Dockerisé

Encadré par:

Mr. Mourad MELLITI

Réalisé & Présenté par:

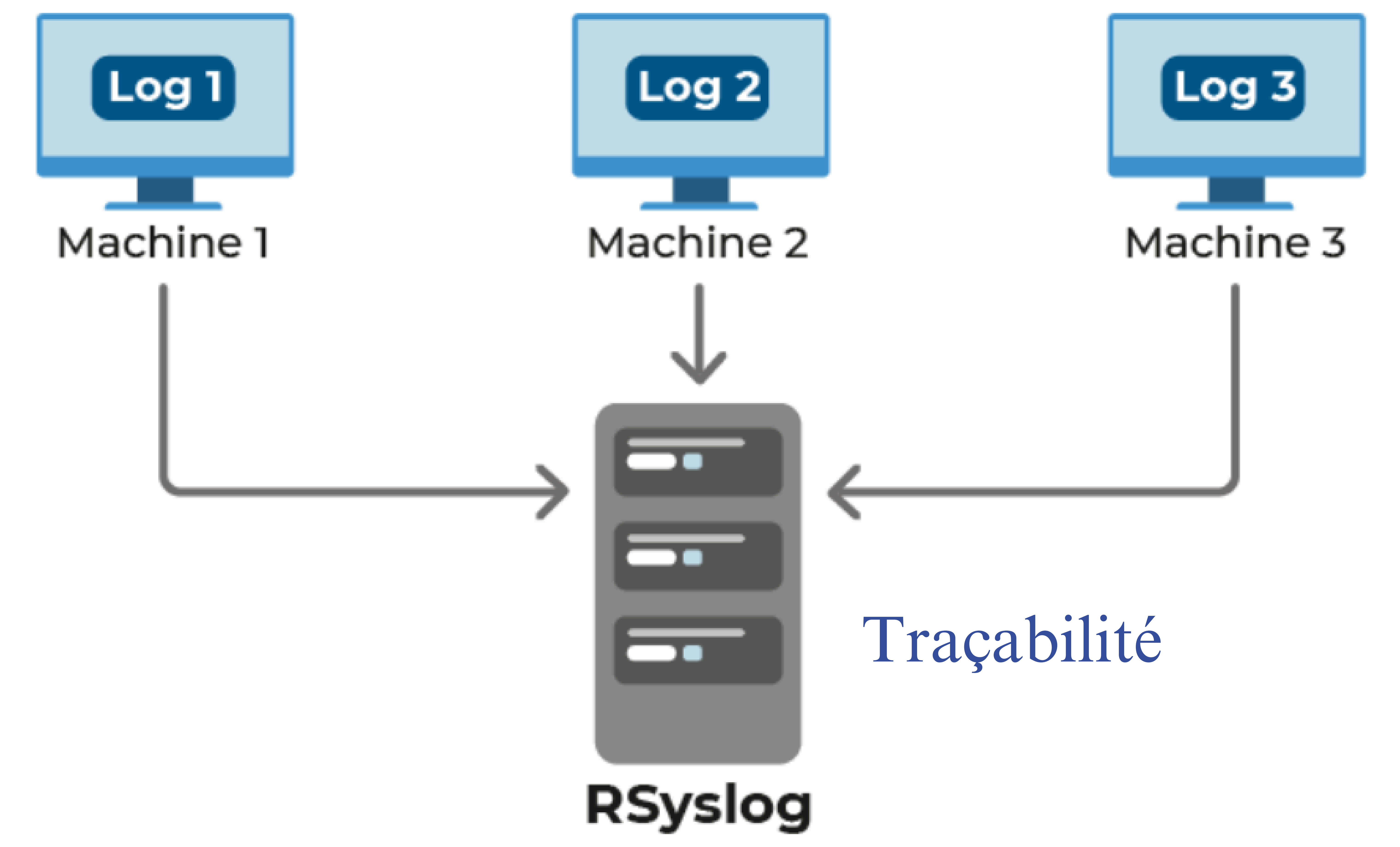
Jihen BOUKHADHRA & Tasnim MAAMOURI

ENICARTHAGE | 2023-2024

Plan

- Introduction
- Rsyslog et Docker
- Étapes de mise en place du serveur rsyslog dockernisé
- Défis
- Conclusion

Introduction



Traçabilité

Rsyslog

- C’est un serveur conçu pour surveiller les périphériques réseaux et systèmes afin d'envoyer des messages de notification et de journalisation.
- **Rsyslog** est la dernière version et elle est la plus utilisée.

Docker

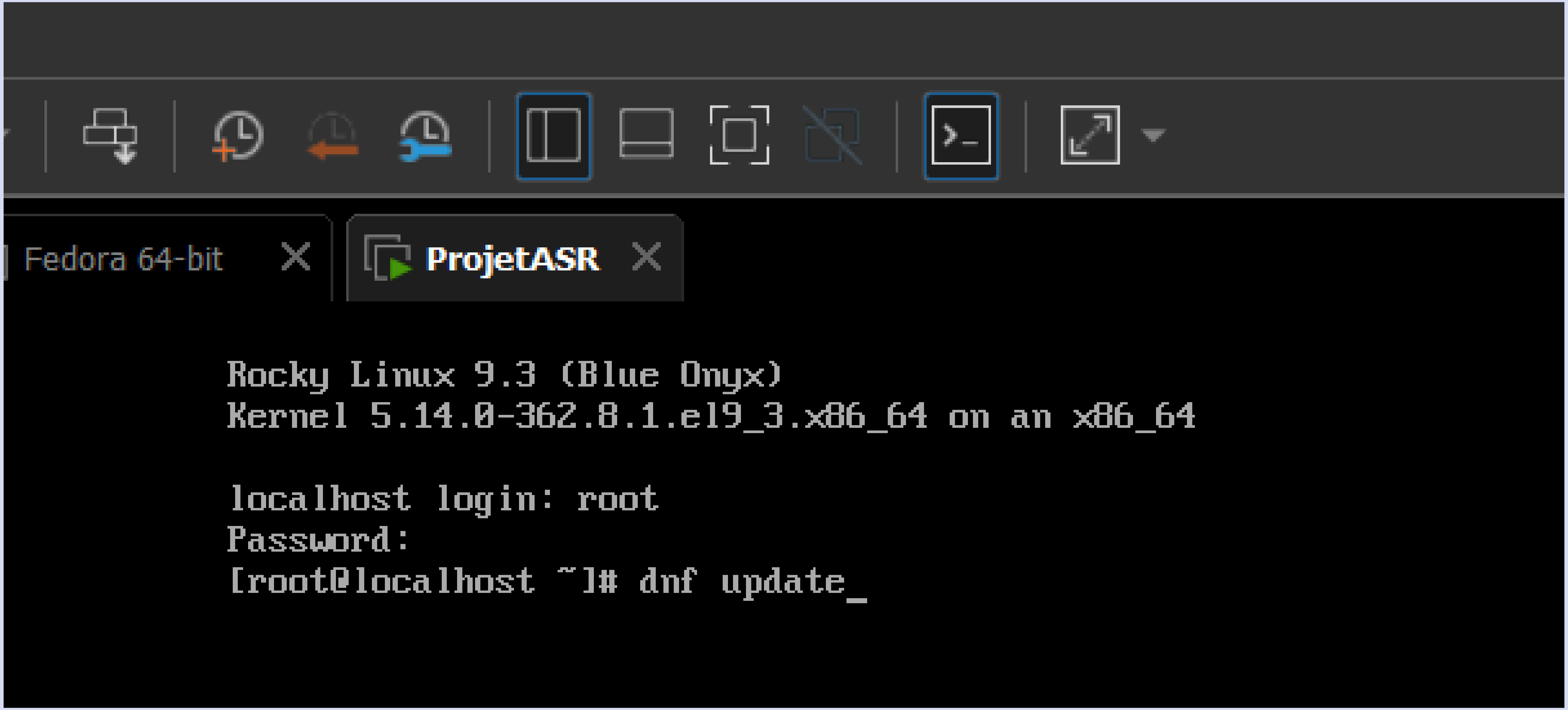
Qu'est-ce qu'un **conteneur**?

- C’est d’un environnement d’exécution léger. C’est une alternative aux méthodes de virtualisation traditionnelles basées sur les VMs.
- Le **Docker** permet d'encapsuler toutes les dépendances relatives au système. On n’a pas besoin d'installer les dépendances car tout est embarqué dans le conteneur.

Étapes de mise en place du serveur rsyslog dockernisé

1 Préparation de votre environnement Rocky Linux

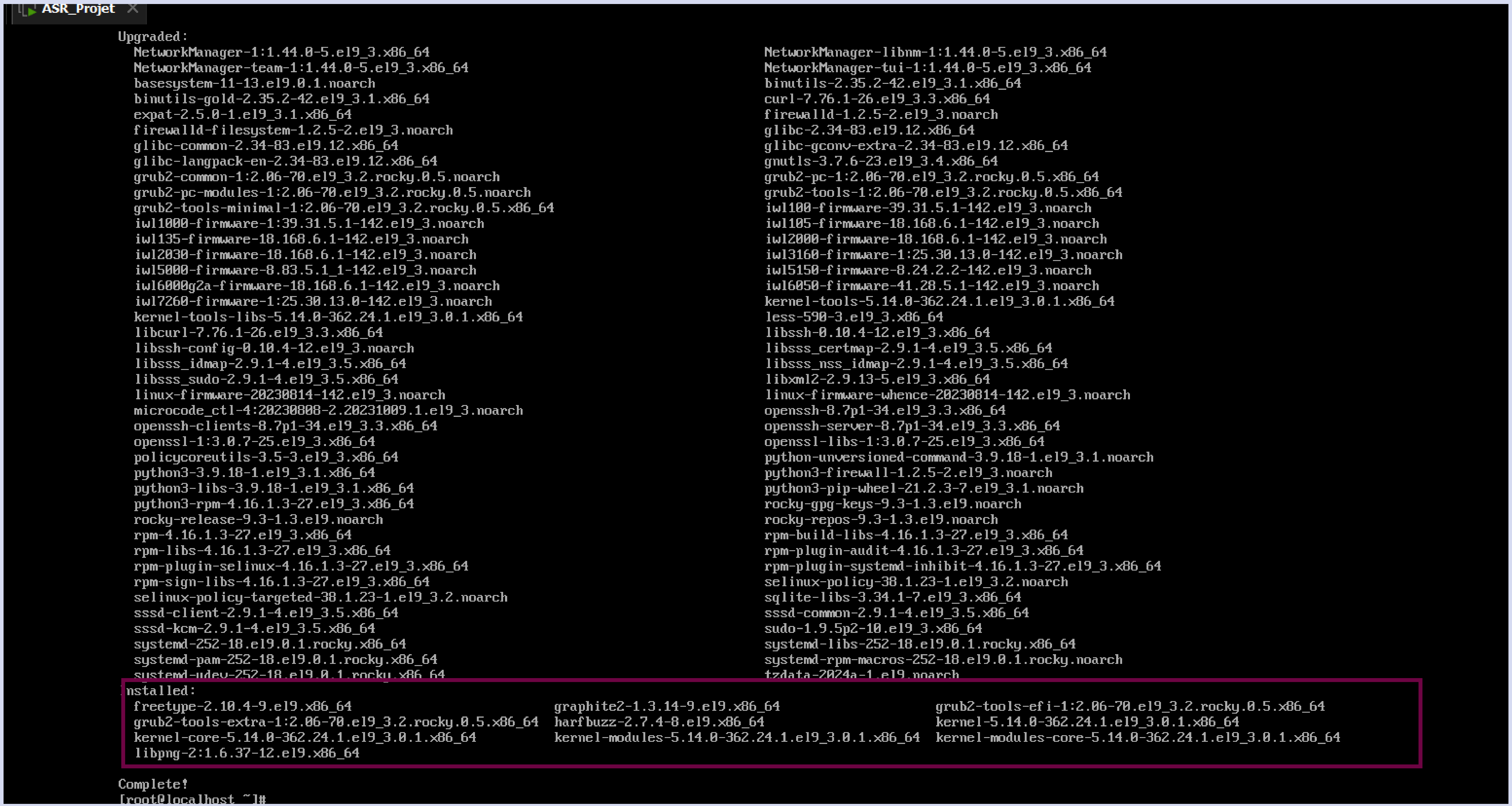
a.Mise à jour système



```
Rocky Linux 9.3 (Blue Onyx)
Kernel 5.14.0-362.8.1.el9_3.x86_64 on an x86_64

localhost login: root
Password:
[root@localhost ~]# dnf update_
```

1 Préparation de votre environnement Rocky Linux

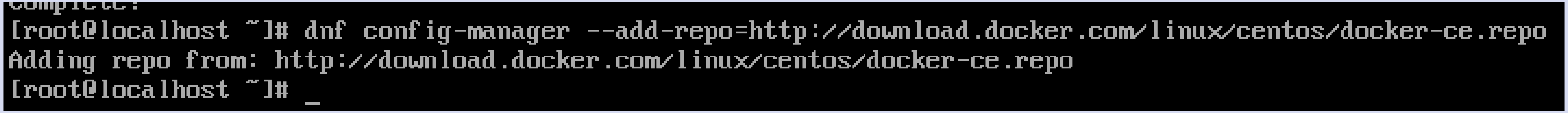
b.Output de la mise à jour système



```
Upgraded:
  NetworkManager-1:1.44.0-5.el9_3.x86_64                                         NetworkManager-libnm-1:1.44.0-5.el9_3.x86_64
  NetworkManager-team-1:1.44.0-5.el9_3.x86_64                                    NetworkManager-tui-1:1.44.0-5.el9_3.x86_64
  basesystem-11-13.el9.0.1.noarch                                                binutils-2.35.2-42.el9_3.1.x86_64
  binutils-gold-2.35.2-42.el9_3.1.x86_64                                         curl-7.76.1-26.el9_3.3.x86_64
  expat-2.5.0-1.el9_3.1.x86_64                                                   firewalld-1.2.5-2.el9_3.noarch
  firewalld-filesystem-1.2.5-2.el9_3.noarch                                      glibc-2.34-83.el9.12.x86_64
  glibc-common-2.34-83.el9.12.x86_64                                             glibc-gconv-extra-2.34-83.el9.12.x86_64
  glibc-langpack-en-2.34-83.el9.12.x86_64                                        gnutls-3.7.6-23.el9_3.4.x86_64
  grub2-common-1:2.06-70.el9_3.2.rocky.0.5.noarch                                grub2-pc-1:2.06-70.el9_3.2.rocky.0.5.x86_64
  grub2-pc-modules-1:2.06-70.el9_3.2.rocky.0.5.noarch                            grub2-tools-1:2.06-70.el9_3.2.rocky.0.5.x86_64
  grub2-tools-minimal-1:2.06-70.el9_3.2.rocky.0.5.x86_64                         iwl100-firmware-39.31.5.1-142.el9_3.noarch
  iwl1000-firmware-1:39.31.5.1-142.el9_3.noarch                                  iwl105-firmware-18.168.6.1-142.el9_3.noarch
  iwl135-firmware-18.168.6.1-142.el9_3.noarch                                    iwl2000-firmware-18.168.6.1-142.el9_3.noarch
  iwl2030-firmware-18.168.6.1-142.el9_3.noarch                                   iwl3160-firmware-1:25.30.13.0-142.el9_3.noarch
  iwl5000-firmware-8.83.5.1_1-142.el9_3.noarch                                   iwl5150-firmware-8.24.2.2-142.el9_3.noarch
  iwl6000g2a-firmware-18.168.6.1-142.el9_3.noarch                                iwl6050-firmware-41.28.5.1-142.el9_3.noarch
  iwl7260-firmware-1:25.30.13.0-142.el9_3.noarch                                 kernel-tools-5.14.0-362.24.1.el9_3.0.1.x86_64
  kernel-tools-libs-5.14.0-362.24.1.el9_3.0.1.x86_64                             less-590-3.el9_3.x86_64
  libcurl-7.76.1-26.el9_3.3.x86_64                                               libssh-0.10.4-12.el9_3.x86_64
  libssh-config-0.10.4-12.el9_3.noarch                                           libsss_certmap-2.9.1-4.el9_3.5.x86_64
  libsss_idmap-2.9.1-4.el9_3.5.x86_64                                            libsss_nss_idmap-2.9.1-4.el9_3.5.x86_64
  libsss_sudo-2.9.1-4.el9_3.5.x86_64                                             libxml2-2.9.13-5.el9_3.x86_64
  linux-firmware-20230814-142.el9_3.noarch                                       linux-firmware-whence-20230814-142.el9_3.noarch
  microcode_ctl-4:20230808-2.20231009.1.el9_3.noarch                             openssh-8.7p1-34.el9_3.3.x86_64
  openssh-clients-8.7p1-34.el9_3.3.x86_64                                        openssh-server-8.7p1-34.el9_3.3.x86_64
  openssl-1:3.0.7-25.el9_3.x86_64                                                openssl-libs-1:3.0.7-25.el9_3.x86_64
  policycoreutils-3.5-3.el9_3.x86_64                                             python-unversioned-command-3.9.18-1.el9_3.1.noarch
  python3-3.9.18-1.el9_3.1.x86_64                                                python3-firewall-1.2.5-2.el9_3.noarch
  python3-libs-3.9.18-1.el9_3.1.x86_64                                           python3-pip-wheel-21.2.3-7.el9_3.1.noarch
  python3-rpm-4.16.1.3-27.el9_3.x86_64                                           rocky-gpg-keys-9.3-1.3.el9.noarch
  rocky-release-9.3-1.3.el9.noarch                                               rocky-repos-9.3-1.3.el9.noarch
  rpm-4.16.1.3-27.el9_3.x86_64                                                   rpm-build-libs-4.16.1.3-27.el9_3.x86_64
  rpm-libs-4.16.1.3-27.el9_3.x86_64                                              rpm-plugin-audit-4.16.1.3-27.el9_3.x86_64
  rpm-plugin-selinux-4.16.1.3-27.el9_3.x86_64                                    rpm-plugin-systemd-inhibit-4.16.1.3-27.el9_3.x86_64
  rpm-sign-libs-4.16.1.3-27.el9_3.x86_64                                         selinux-policy-38.1.23-1.el9_3.2.noarch
  selinux-policy-targeted-38.1.23-1.el9_3.2.noarch                               sqlite-libs-3.34.1-7.el9_3.x86_64
  sssd-client-2.9.1-4.el9_3.5.x86_64                                             sssd-common-2.9.1-4.el9_3.5.x86_64
  sssd-kcm-2.9.1-4.el9_3.5.x86_64                                                sudo-1.9.5p2-10.el9_3.x86_64
  systemd-252-18.el9.0.1.rocky.x86_64                                            systemd-libs-252-18.el9.0.1.rocky.x86_64
  systemd-pam-252-18.el9.0.1.rocky.x86_64                                        systemd-rpm-macros-252-18.el9.0.1.rocky.noarch
  systemd-udev-252-18.el9.0.1.rocky.x86_64                                       tzdata-2024a-1.el9.noarch
Installed:
  freetype-2.10.4-9.el9.x86_64                          graphite2-1.3.14-9.el9.x86_64                    grub2-tools-efi-1:2.06-70.el9_3.2.rocky.0.5.x86_64
  grub2-tools-extra-1:2.06-70.el9_3.2.rocky.0.5.x86_64  harfbuzz-2.7.4-8.el9.x86_64                      kernel-5.14.0-362.24.1.el9_3.0.1.x86_64
  kernel-core-5.14.0-362.24.1.el9_3.0.1.x86_64          kernel-modules-5.14.0-362.24.1.el9_3.0.1.x86_64  kernel-modules-core-5.14.0-362.24.1.el9_3.0.1.x86_64
  libpng-2:1.6.37-12.el9.x86_64

Complete!
[root@localhost ~]#
```

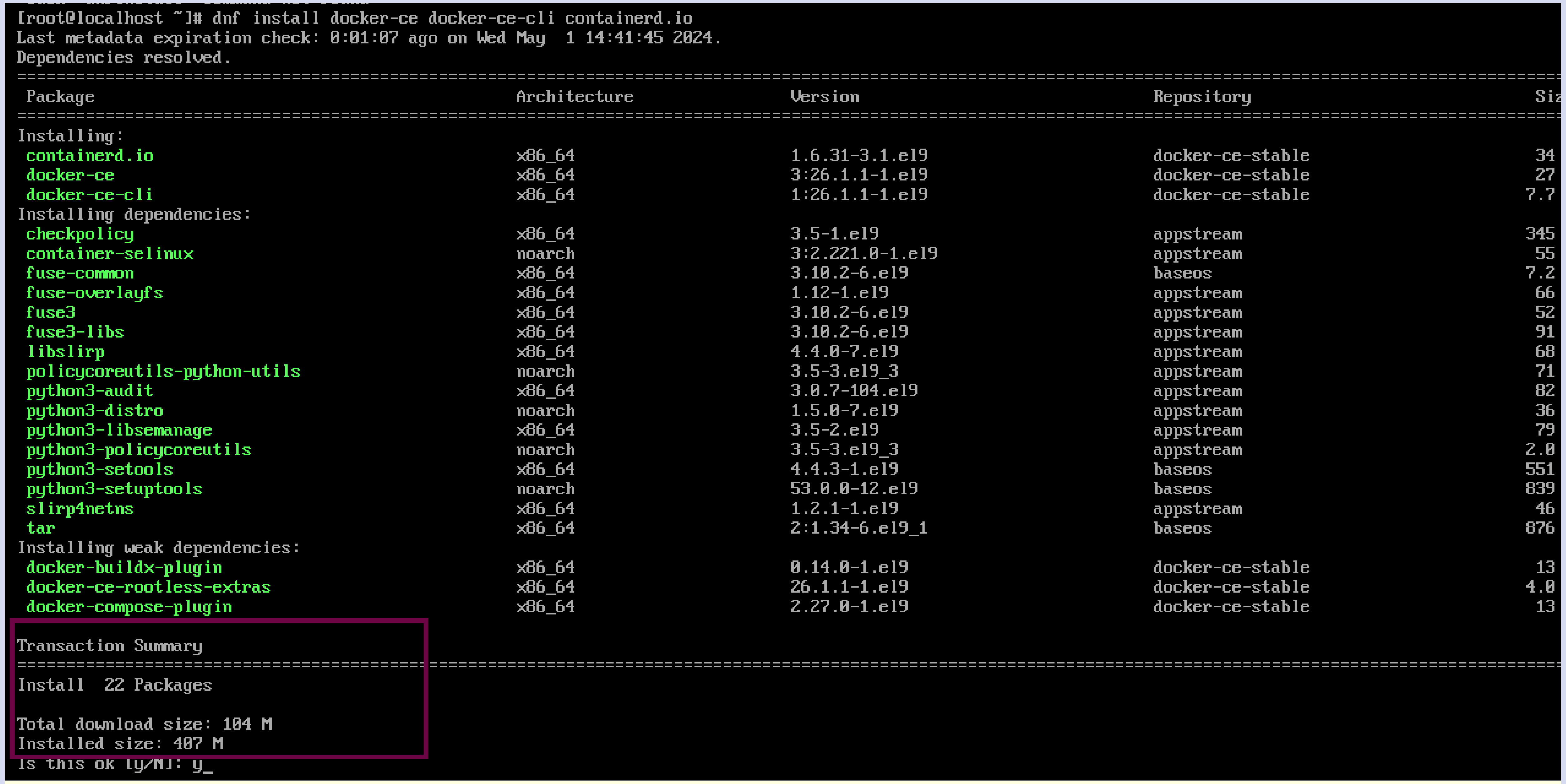
2 Installation du Docker

a. Ajout du Docker Repository



```
Complete!
[root@localhost ~]# dnf config-manager --add-repo=http://download.docker.com/linux/centos/docker-ce.repo
Adding repo from: http://download.docker.com/linux/centos/docker-ce.repo
[root@localhost ~]# _
```

b.Installation du Docker Packages



```
[root@localhost ~]# dnf install docker-ce docker-ce-cli containerd.io
Last metadata expiration check: 0:01:07 ago on Wed May  1 14:41:45 2024.
Dependencies resolved.
================================================================================================================================================================
 Package                                            Architecture                Version                              Repository                             Siz
================================================================================================================================================================
Installing:
 containerd.io                                      x86_64                      1.6.31-3.1.el9                       docker-ce-stable                         34
 docker-ce                                          x86_64                      3:26.1.1-1.el9                       docker-ce-stable                         27
 docker-ce-cli                                      x86_64                      1:26.1.1-1.el9                       docker-ce-stable                        7.7
Installing dependencies:
 checkpolicy                                        x86_64                      3.5-1.el9                            appstream                               345
 container-selinux                                  noarch                      3:2.221.0-1.el9                      appstream                                55
 fuse-common                                        x86_64                      3.10.2-6.el9                         baseos                                  7.2
 fuse-overlayfs                                     x86_64                      1.12-1.el9                           appstream                                66
 fuse3                                              x86_64                      3.10.2-6.el9                         appstream                                52
 fuse3-libs                                         x86_64                      3.10.2-6.el9                         appstream                                91
 libslirp                                           x86_64                      4.4.0-7.el9                          appstream                                68
 policycoreutils-python-utils                       noarch                      3.5-3.el9_3                          appstream                                71
 python3-audit                                      x86_64                      3.0.7-104.el9                        appstream                                82
 python3-distro                                     noarch                      1.5.0-7.el9                          appstream                                36
 python3-libsemanage                                x86_64                      3.5-2.el9                            appstream                                79
 python3-policycoreutils                            noarch                      3.5-3.el9_3                          appstream                               2.0
 python3-setools                                    x86_64                      4.4.3-1.el9                          baseos                                  551
 python3-setuptools                                 noarch                      53.0.0-12.el9                        baseos                                  839
 slirp4netns                                        x86_64                      1.2.1-1.el9                          appstream                                46
 tar                                                x86_64                      2:1.34-6.el9_1                       baseos                                  876
Installing weak dependencies:
 docker-buildx-plugin                               x86_64                      0.14.0-1.el9                         docker-ce-stable                         13
 docker-ce-rootless-extras                          x86_64                      26.1.1-1.el9                         docker-ce-stable                        4.0
 docker-compose-plugin                              x86_64                      2.27.0-1.el9                         docker-ce-stable                         13

Transaction Summary
================================================================================================================================================================
Install  22 Packages

Total download size: 104 M
Installed size: 407 M
Is this ok [y/N]: y_
```

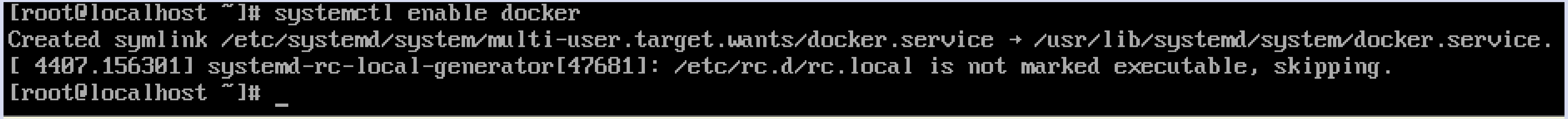
docker-ce: Il est le package principal

docker-ce-cli: Il fournit l'interface en ligne de commande

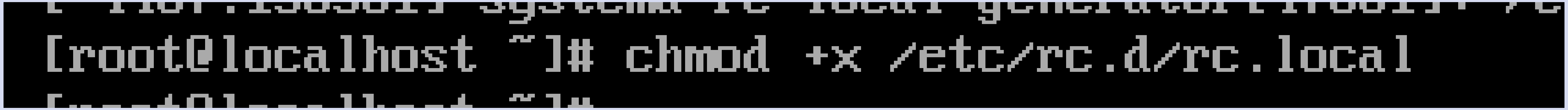
containerd.io: Il installe le moteur d'exécution du conteneur “containerd”

3 Activation et Lancement du Docker

#systemctl enable docker

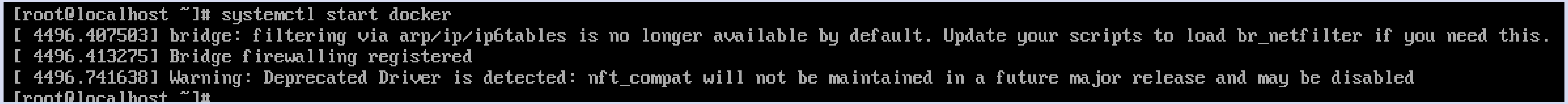


```
[root@localhost ~]# systemctl enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[ 4407.156301] systemd-rc-local-generator[47681]: /etc/rc.d/rc.local is not marked executable, skipping.
[root@localhost ~]# _
```

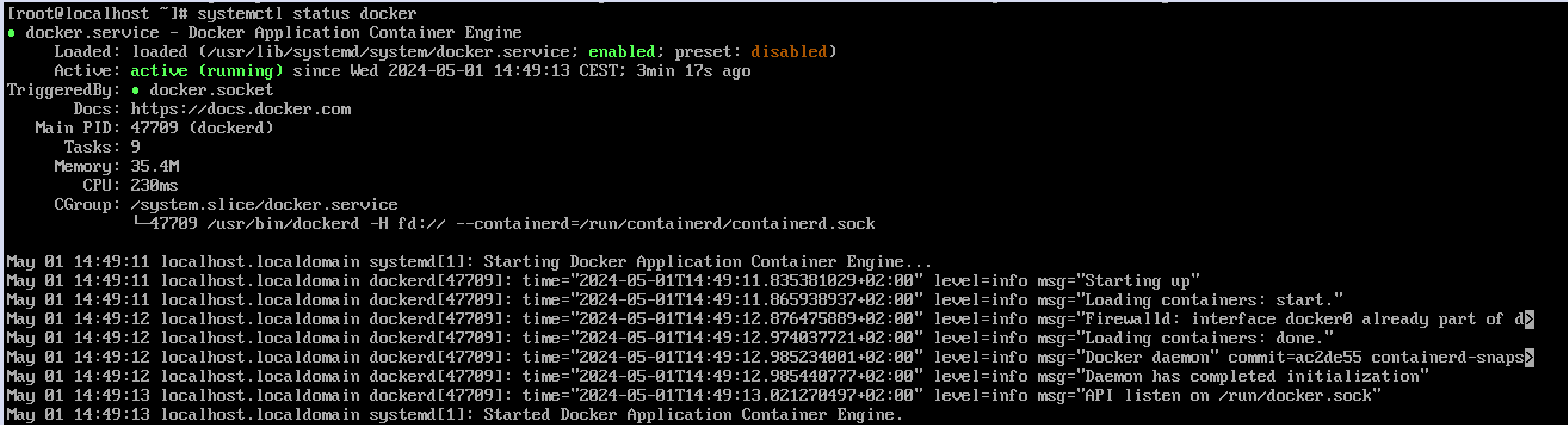


```
[root@localhost ~]# chmod +x /etc/rc.d/rc.local
[root@localhost ~]#
```

#systemctl start docker



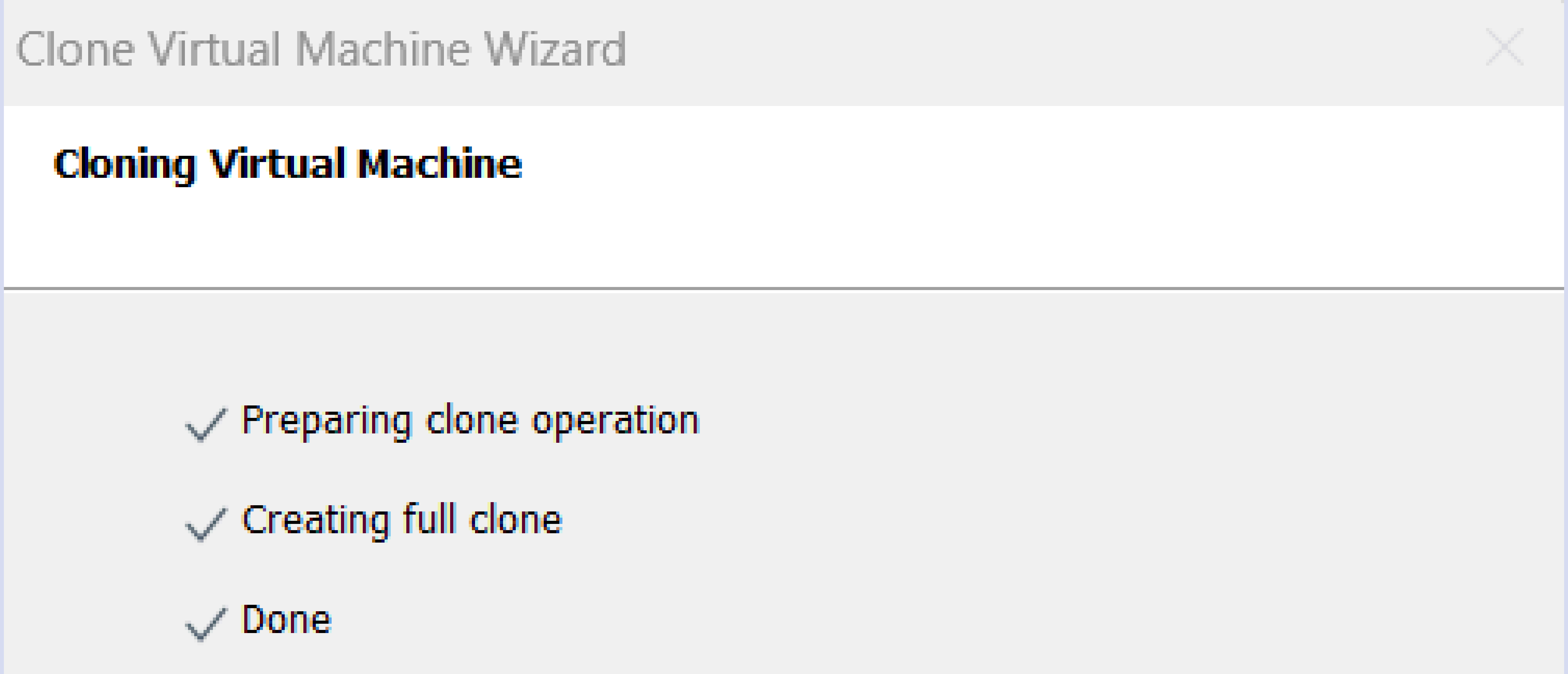
```
[root@localhost ~]# systemctl start docker
[ 4496.407503] bridge: filtering via arp/ip/ip6tables is no longer available by default. Update your scripts to load br_netfilter if you need this.
[ 4496.413275] Bridge firewalling registered
[ 4496.741638] Warning: Deprecated Driver is detected: nft_compat will not be maintained in a future major release and may be disabled
[root@localhost ~]#
```



```
[root@localhost ~]# systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: disabled)
     Active: active (running) since Wed 2024-05-01 14:49:13 CEST; 3min 17s ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 47709 (dockerd)
      Tasks: 9
     Memory: 35.4M
        CPU: 230ms
     CGroup: /system.slice/docker.service
             └─47709 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

May 01 14:49:11 localhost.localdomain systemd[1]: Starting Docker Application Container Engine...
May 01 14:49:11 localhost.localdomain dockerd[47709]: time="2024-05-01T14:49:11.835381029+02:00" level=info msg="Starting up"
May 01 14:49:11 localhost.localdomain dockerd[47709]: time="2024-05-01T14:49:11.865938937+02:00" level=info msg="Loading containers: start."
May 01 14:49:12 localhost.localdomain dockerd[47709]: time="2024-05-01T14:49:12.876475889+02:00" level=info msg="Firewalld: interface docker0 already part of d>
May 01 14:49:12 localhost.localdomain dockerd[47709]: time="2024-05-01T14:49:12.974037721+02:00" level=info msg="Loading containers: done."
May 01 14:49:12 localhost.localdomain dockerd[47709]: time="2024-05-01T14:49:12.985234001+02:00" level=info msg="Docker daemon" commit=ac2de55 containerd-snaps>
May 01 14:49:12 localhost.localdomain dockerd[47709]: time="2024-05-01T14:49:12.985440777+02:00" level=info msg="Daemon has completed initialization"
May 01 14:49:13 localhost.localdomain dockerd[47709]: time="2024-05-01T14:49:13.021270497+02:00" level=info msg="API listen on /run/docker.sock"
May 01 14:49:13 localhost.localdomain systemd[1]: Started Docker Application Container Engine.
```

4 Clonage de la VM

Le clonage d'une machine virtuelle est un processus de duplication d'une instance VM existante. Cela peut être utile dans de nombreux scénarios, notamment pour la sauvegarde, la création de machines virtuelles de test ou de développement à partir d'une configuration existante.



Clone Virtual Machine Wizard

Cloning Virtual Machine

✓ Preparing clone operation

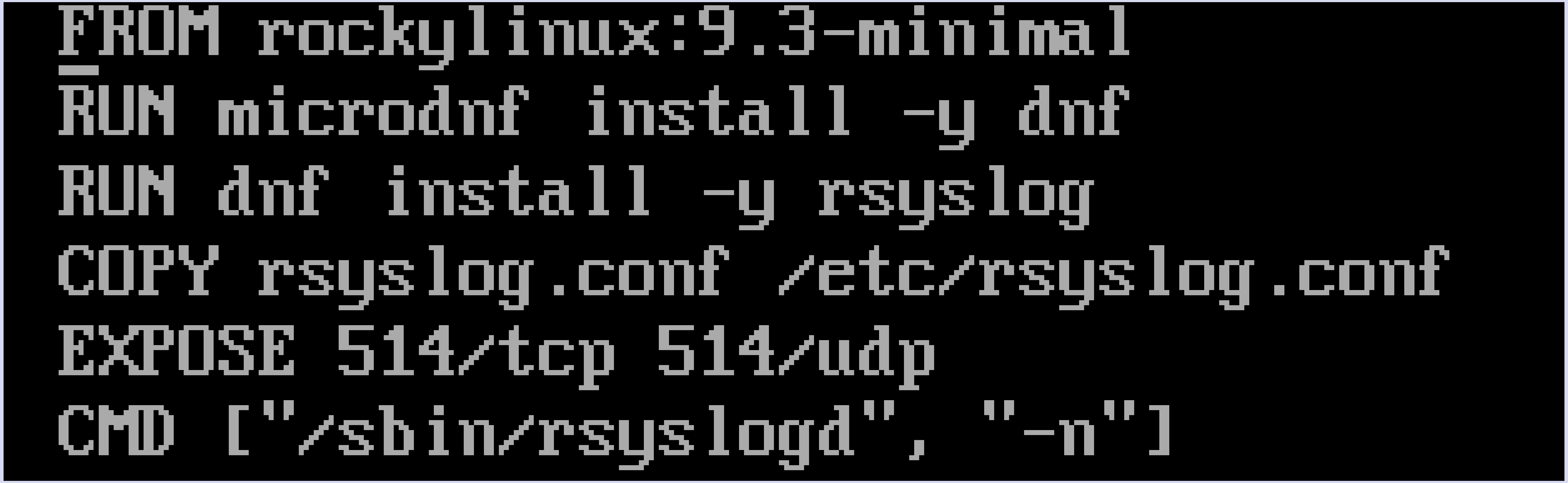
✓ Creating full clone

✓ Done

5 Création du conteneur Rsyslog

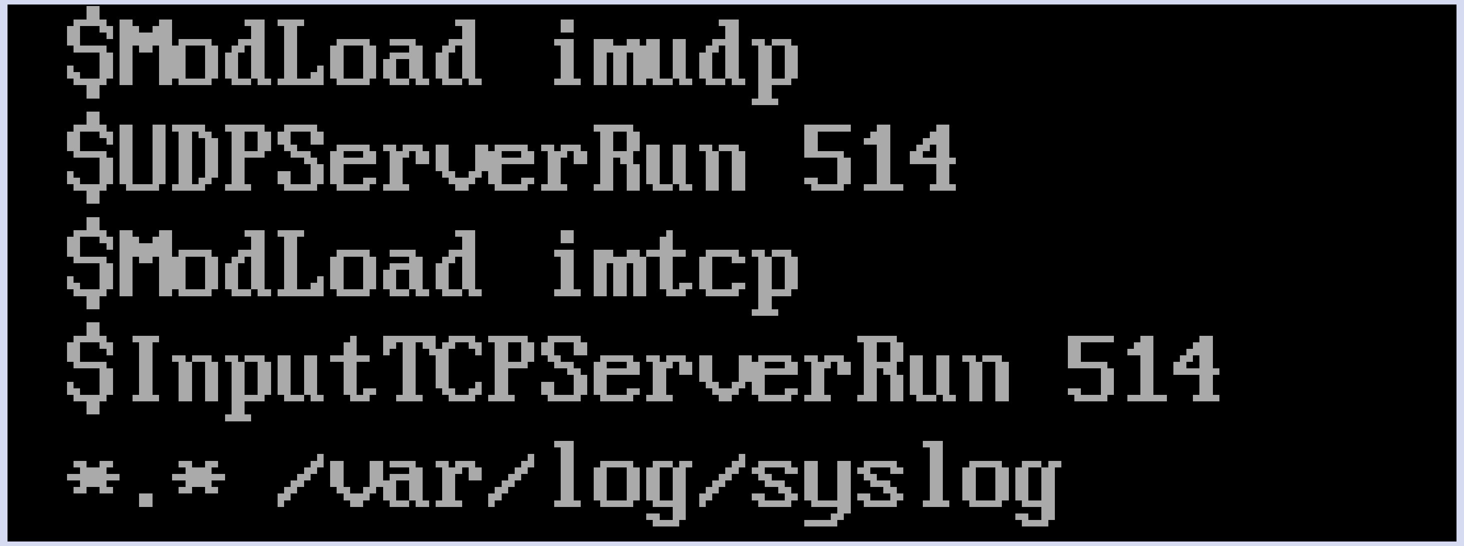
--->Créer un fichier Dockerfile pour définir notre conteneur rsyslog

Le **Dockerfile** contient les instructions nécessaires pour construire l'image Docker.



```
FROM rockylinux:9.3-minimal
RUN microdnf install -y dnf
RUN dnf install -y rsyslog
COPY rsyslog.conf /etc/rsyslog.conf
EXPOSE 514/tcp 514/udp
CMD ["/sbin/rsyslogd", "-n"]
```

--->Ajouter de la configuration de rsyslog



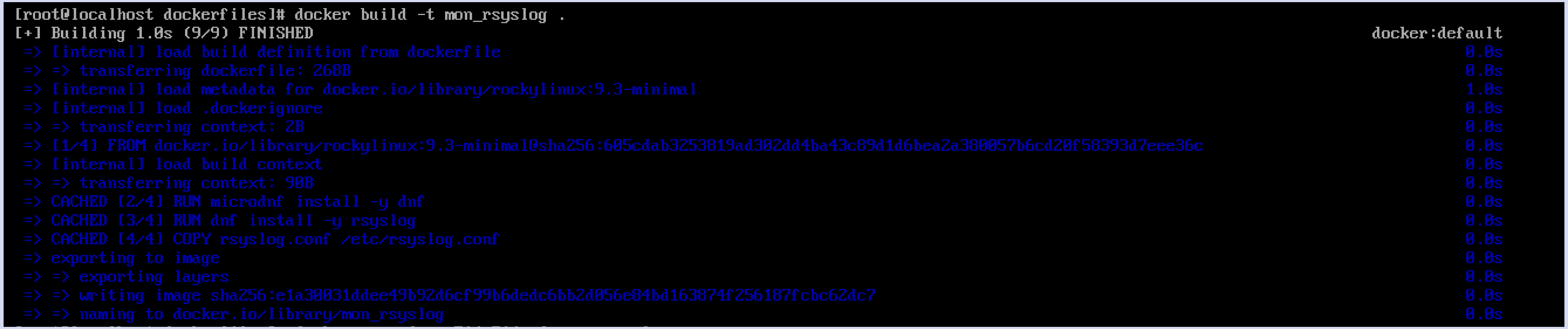
```
$ModLoad imudp
$UDPServerRun 514
$ModLoad imtcp
$InputTCPServerRun 514
*.* /var/log/syslog
```

Ce fichier de configuration configure rsyslog pour écouter les messages sur les ports UDP et TCP 514 et les rediriger vers **/var/log/syslog**

6 Construction et exécution du conteneur

--->Construire une image Docker à partir du Dockerfile

#docker build -t mon_rsyslog



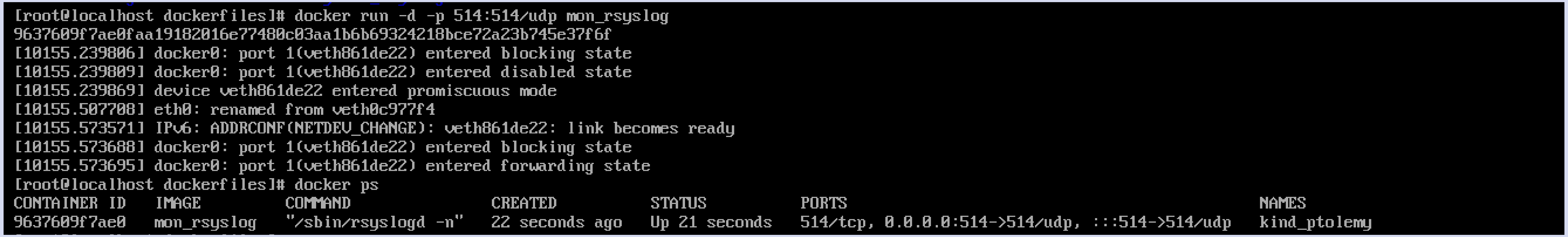
```
[root@localhost dockerfiles]# docker build -t mon_rsyslog .
[+] Building 1.0s (9/9) FINISHED                                                                                                                   docker:default
 => [internal] load build definition from dockerfile                                                                                                         0.0s
 => => transferring dockerfile: 268B                                                                                                                         0.0s
 => [internal] load metadata for docker.io/library/rockylinux:9.3-minimal                                                                                    1.0s
 => [internal] load .dockerignore                                                                                                                            0.0s
 => => transferring context: 2B                                                                                                                              0.0s
 => [1/4] FROM docker.io/library/rockylinux:9.3-minimal@sha256:605cdab3253819ad302dd4ba43c89d1d6bea2a380057b6cd20f58393d7eee36c                              0.0s
 => [internal] load build context                                                                                                                            0.0s
 => => transferring context: 90B                                                                                                                             0.0s
 => CACHED [2/4] RUN microdnf install -y dnf                                                                                                                 0.0s
 => CACHED [3/4] RUN dnf install -y rsyslog                                                                                                                  0.0s
 => CACHED [4/4] COPY rsyslog.conf /etc/rsyslog.conf                                                                                                         0.0s
 => exporting to image                                                                                                                                       0.0s
 => => exporting layers                                                                                                                                      0.0s
 => => writing image sha256:e1a30031ddee49b92d6cf99b6dedc6bb2d056e84bd163874f256187fcbc62dc7                                                                 0.0s
 => => naming to docker.io/library/mon_rsyslog                                                                                                               0.0s
```

--->Exécuter le conteneur rsyslog

#docker run -d -p 514:514/udp mon_rsyslog

--->Puis, vérifier les conteneurs en cours d’éxecution

#docker ps



```
[root@localhost dockerfiles]# docker run -d -p 514:514/udp mon_rsyslog
9637609f7ae0faa19182016e77480c03aa1b6b69324218bce72a23b745e37f6f
[10155.239806] docker0: port 1(veth861de22) entered blocking state
[10155.239809] docker0: port 1(veth861de22) entered disabled state
[10155.239869] device veth861de22 entered promiscuous mode
[10155.507708] eth0: renamed from veth0c977f4
[10155.573571] IPv6: ADDRCONF(NETDEV_CHANGE): veth861de22: link becomes ready
[10155.573688] docker0: port 1(veth861de22) entered blocking state
[10155.573695] docker0: port 1(veth861de22) entered forwarding state
[root@localhost dockerfiles]# docker ps
CONTAINER ID   IMAGE         COMMAND                CREATED          STATUS          PORTS                                            NAMES
9637609f7ae0   mon_rsyslog   "/sbin/rsyslogd -n"   22 seconds ago   Up 21 seconds   514/tcp, 0.0.0.0:514->514/udp, :::514->514/udp   kind_ptolemy
```

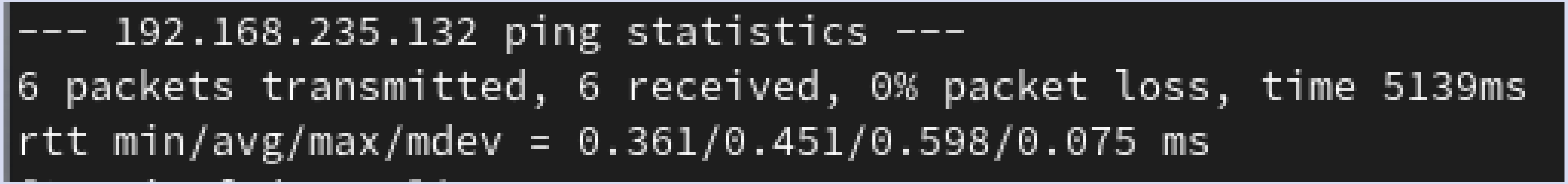
7

Phase de test avec FIDORA

--->Envoyer des journaux depuis un autre appareil (machine virtuelle Fidora)

--->S’assurer que la machine virtuelle Fedora peut atteindre la machine hôte où le conteneur Docker rsyslog est en cours d'exécution

#ping 192.168.235.132

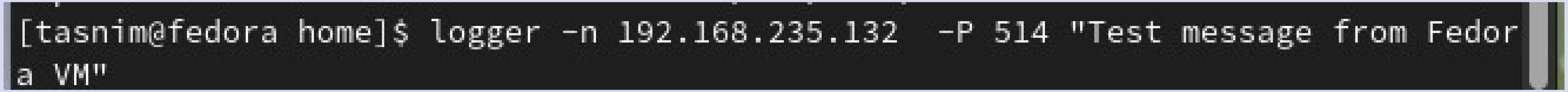


```
--- 192.168.235.132 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5139ms
rtt min/avg/max/mdev = 0.361/0.451/0.598/0.075 ms
```

--->Envoyer un message de test à votre conteneur Docker rsyslog depuis la machine virtuelle Fedora

--->Envoyer un message de test au rsyslog

>logger -n 192.168.235.132 -p 514 “Test message from Fedora à VM”



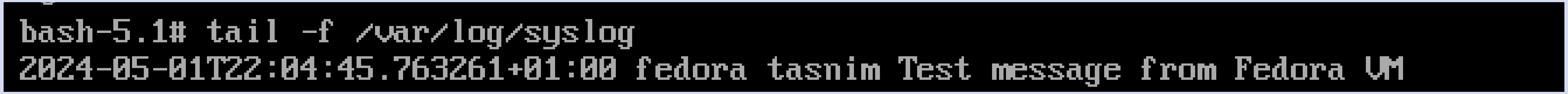
```
[tasnim@fedora home]$ logger -n 192.168.235.132  -P 514 "Test message from Fedor
a VM"
```

7 Phase de test avec FIDORA

--->Se connecter au conteneur Docker rsyslog

#docker exec -it 0af40285 /bin/bash

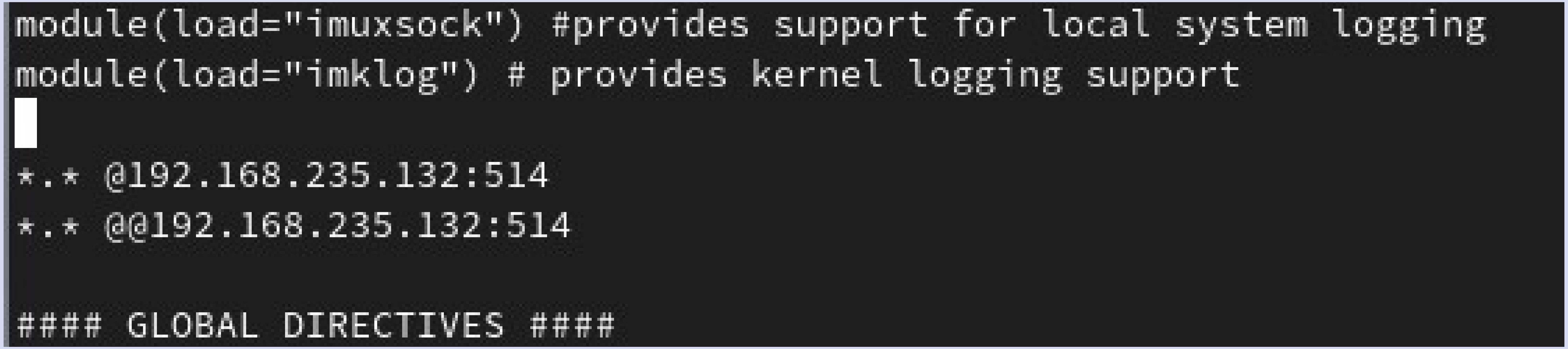
--->Vérifier les journaux actuels en utilisant la commande tail
#tail -f /var/log/syslog



```
bash-5.1# tail -f /var/log/syslog
2024-05-01T22:04:45.763261+01:00 fedora tasnim Test message from Fedora VM
```

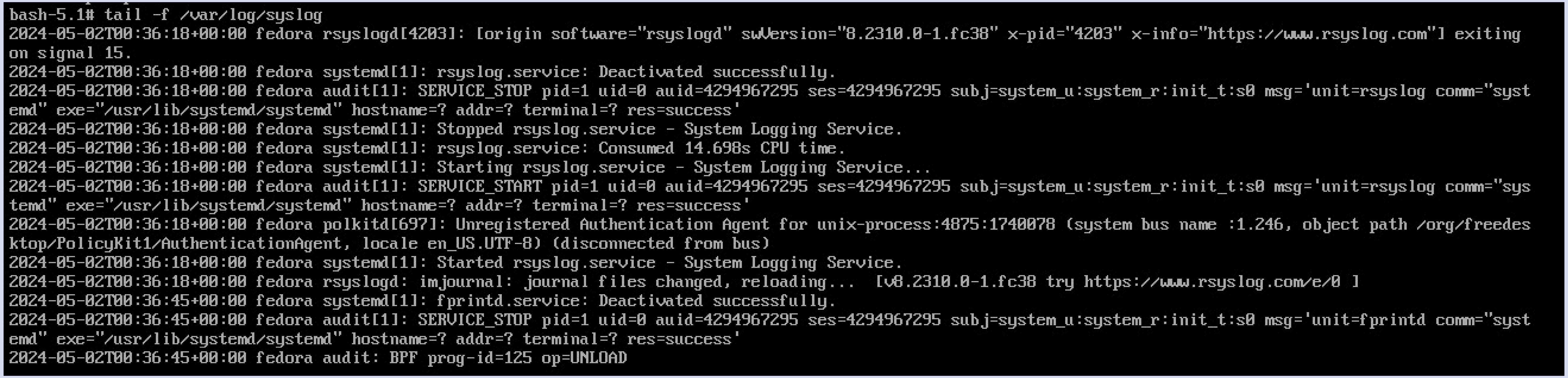
7 Phase de test avec FIDORA

--->Rediriger Log de fedora vers notre conteneur Rsyslog:
-->Mettre ces configurations dans la fichier **rsyslog.conf** sur Fedora



```
module(load="imuxsock") #provides support for local system logging
module(load="imklog") # provides kernel logging support

*.* @192.168.235.132:514
*.* @@192.168.235.132:514

#### GLOBAL DIRECTIVES ####
```

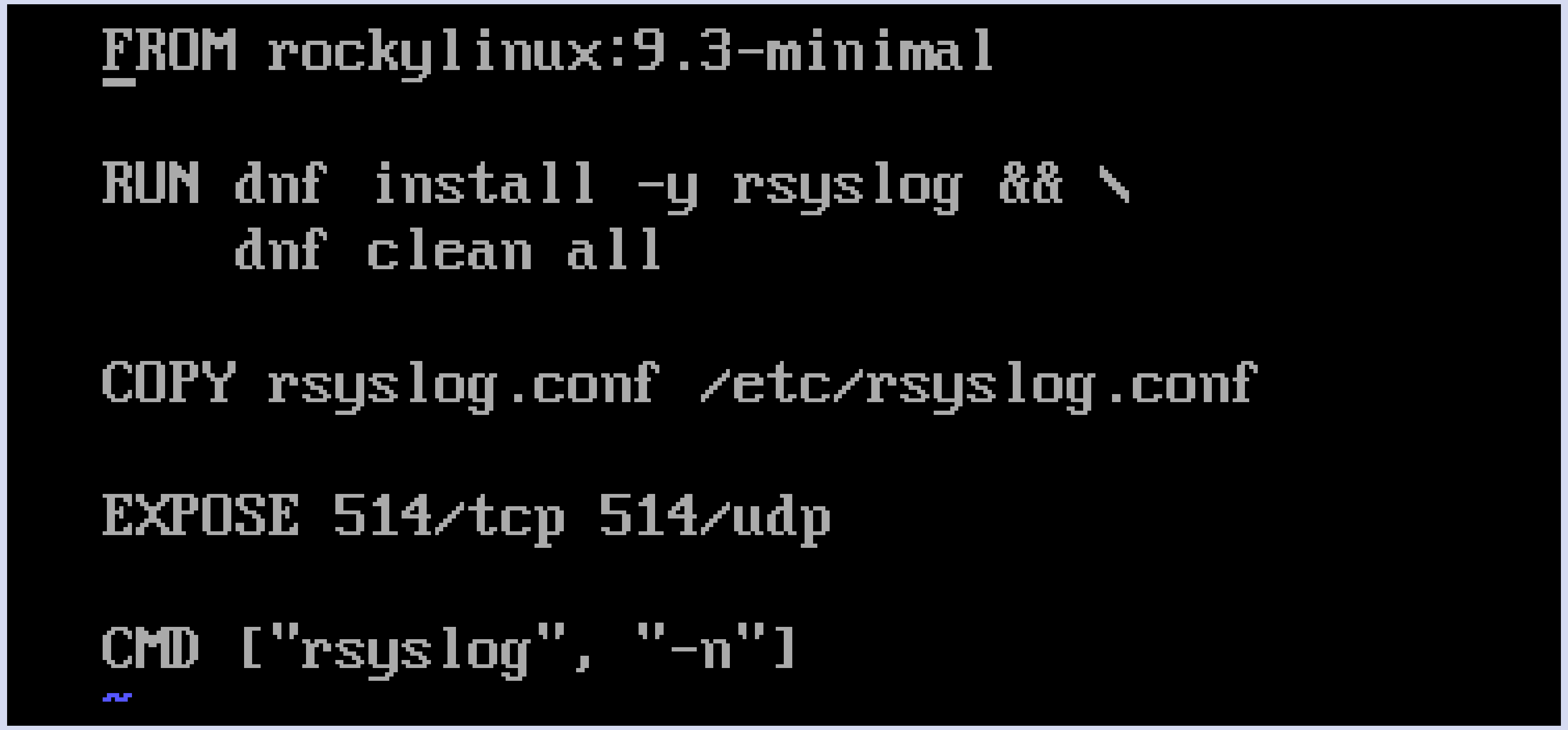
-->pour vérifier les journaux actuels en utilisant la commande: **#tail -f /var/log/syslog**



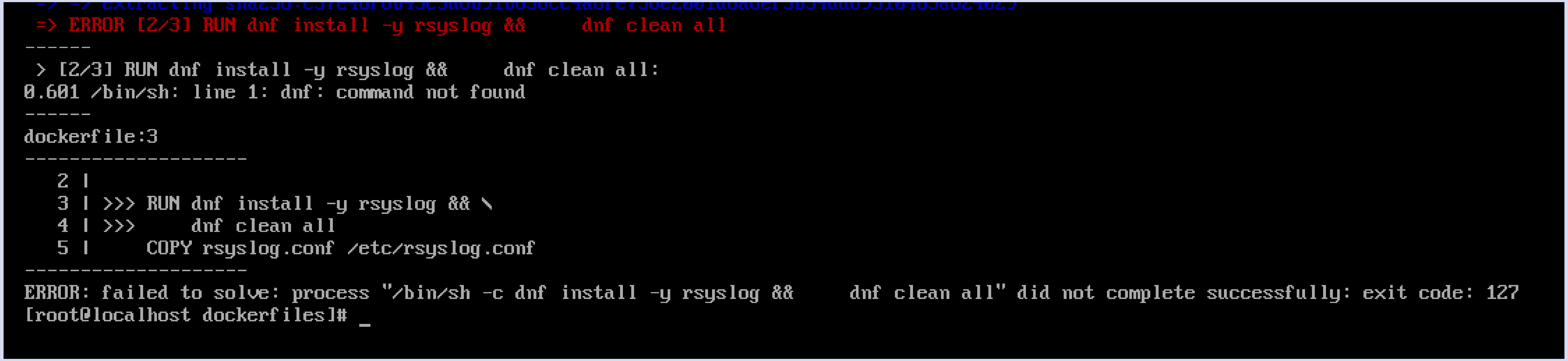
```
bash-5.1# tail -f /var/log/syslog
2024-05-02T00:36:18+00:00 fedora rsyslogd[4203]: [origin software="rsyslogd" swVersion="8.2310.0-1.fc38" x-pid="4203" x-info="https://www.rsyslog.com"] exiting
on signal 15.
2024-05-02T00:36:18+00:00 fedora systemd[1]: rsyslog.service: Deactivated successfully.
2024-05-02T00:36:18+00:00 fedora audit[1]: SERVICE_STOP pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=rsyslog comm="syst
emd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
2024-05-02T00:36:18+00:00 fedora systemd[1]: Stopped rsyslog.service - System Logging Service.
2024-05-02T00:36:18+00:00 fedora systemd[1]: rsyslog.service: Consumed 14.698s CPU time.
2024-05-02T00:36:18+00:00 fedora systemd[1]: Starting rsyslog.service - System Logging Service...
2024-05-02T00:36:18+00:00 fedora audit[1]: SERVICE_START pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=rsyslog comm="sys
temd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
2024-05-02T00:36:18+00:00 fedora polkitd[697]: Unregistered Authentication Agent for unix-process:4875:1740078 (system bus name :1.246, object path /org/freedes
ktop/PolicyKit1/AuthenticationAgent, locale en_US.UTF-8) (disconnected from bus)
2024-05-02T00:36:18+00:00 fedora systemd[1]: Started rsyslog.service - System Logging Service.
2024-05-02T00:36:18+00:00 fedora rsyslogd: imjournal: journal files changed, reloading...  [v8.2310.0-1.fc38 try https://www.rsyslog.com/e/0 ]
2024-05-02T00:36:45+00:00 fedora systemd[1]: fprintd.service: Deactivated successfully.
2024-05-02T00:36:45+00:00 fedora audit[1]: SERVICE_STOP pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=fprintd comm="syst
emd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
2024-05-02T00:36:45+00:00 fedora audit: BPF prog-id=125 op=UNLOAD
```

Défis

1er Défi: Problème avec “RUN dnf”



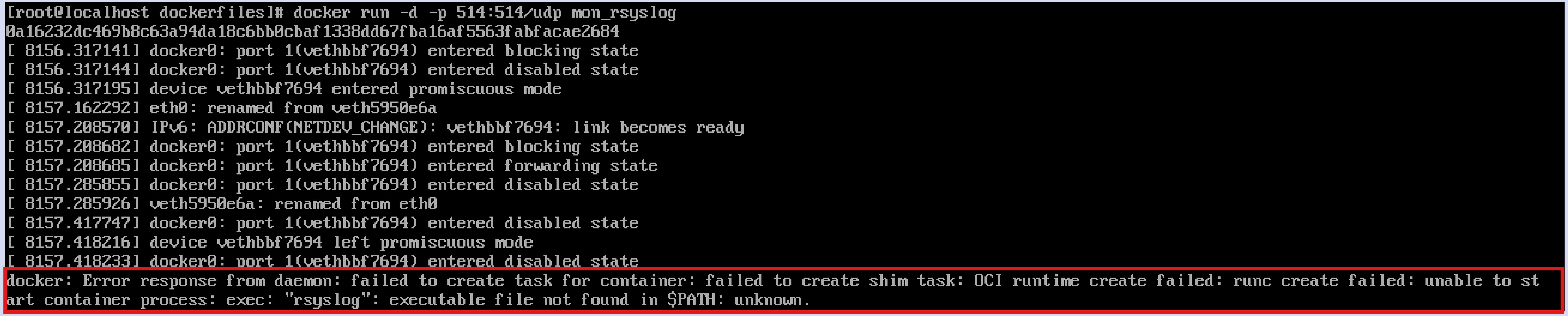
```
FROM rockylinux:9.3-minimal

RUN dnf install -y rsyslog && \
    dnf clean all

COPY rsyslog.conf /etc/rsyslog.conf

EXPOSE 514/tcp 514/udp

CMD ["rsyslog", "-n"]
~
```



```
 => => extracting sha256:[illegible]
 => ERROR [2/3] RUN dnf install -y rsyslog &&     dnf clean all
------
 > [2/3] RUN dnf install -y rsyslog &&     dnf clean all:
0.601 /bin/sh: line 1: dnf: command not found
------
dockerfile:3
--------------------
   2 |
   3 | >>> RUN dnf install -y rsyslog && \
   4 | >>>     dnf clean all
   5 |     COPY rsyslog.conf /etc/rsyslog.conf
--------------------
ERROR: failed to solve: process "/bin/sh -c dnf install -y rsyslog &&     dnf clean all" did not complete successfully: exit code: 127
[root@localhost dockerfiles]# _
```

✓ Ajouter microdnf install pour installer dnf

2ème Défi: Exécutable non trouvé



```
[root@localhost dockerfiles]# docker run -d -p 514:514/udp mon_rsyslog
0a16232dc469b8c63a94da18c6bb0cbaf1338dd67fba16af5563fabfacae2684
[ 8156.317141] docker0: port 1(vethbbf7694) entered blocking state
[ 8156.317144] docker0: port 1(vethbbf7694) entered disabled state
[ 8156.317195] device vethbbf7694 entered promiscuous mode
[ 8157.162292] eth0: renamed from veth5950e6a
[ 8157.208570] IPv6: ADDRCONF(NETDEV_CHANGE): vethbbf7694: link becomes ready
[ 8157.208682] docker0: port 1(vethbbf7694) entered blocking state
[ 8157.208685] docker0: port 1(vethbbf7694) entered forwarding state
[ 8157.285855] docker0: port 1(vethbbf7694) entered disabled state
[ 8157.285926] veth5950e6a: renamed from eth0
[ 8157.417747] docker0: port 1(vethbbf7694) entered disabled state
[ 8157.418216] device vethbbf7694 left promiscuous mode
[ 8157.418233] docker0: port 1(vethbbf7694) entered disabled state
docker: Error response from daemon: failed to create task for container: failed to create shim task: OCI runtime create failed: runc create failed: unable to st
art container process: exec: "rsyslog": executable file not found in $PATH: unknown.
```

✓ Spécifier **/sbin/rsyslogd** comme emplacement de l'exécutable rsyslog dans notre Dockerfile

❗ Dans de nombreuses distributions Linux, l'**exécutable de base** pour **rsyslog** est **rsyslogd**.

Merci pour Votre Attention

Réalisé & Présenté par:

Jihen BOUKHADHRA & Tasnim MAAMOURI

ENICARTHAGE | 2023-2024