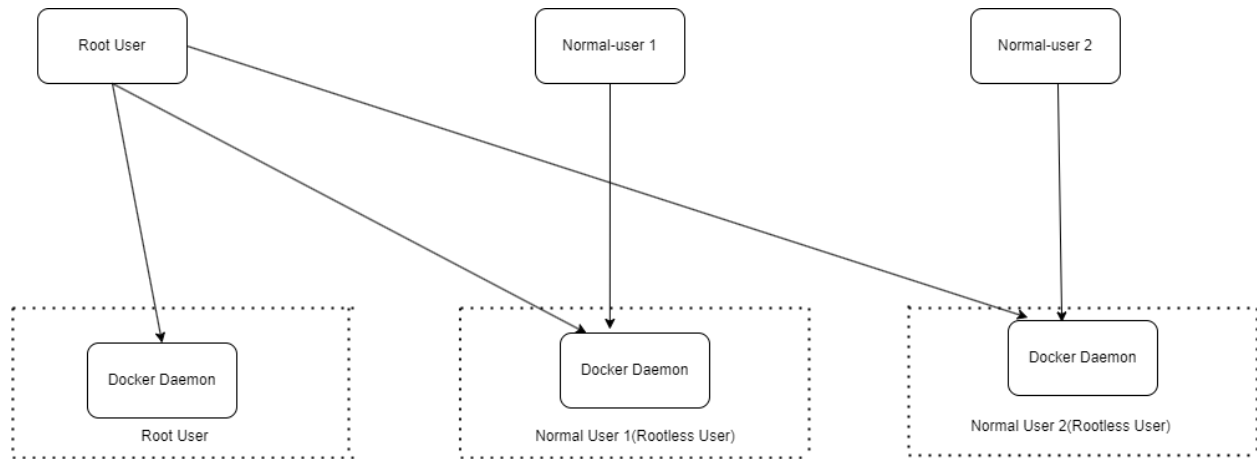


Running Rootless Container's using Docker



Rootless Docker is a security feature that allows you to run Docker containers without root privileges.

1. **Definition:** Rootless Docker enables running the Docker daemon and containers as a non-root user, reducing the potential security risks associated with running containers with root privileges.

2. Key benefits:

- Enhanced security: Limits the impact of potential container breakouts
- Improved isolation: Each user can have their own Docker daemon
- Reduced attack surface: Minimizes the risk of privilege escalation attacks

3. How it works:

- Uses user namespaces to map the root user inside the container to a non-root user on the host
- Utilizes a separate daemon process for each non-root user

4. Use cases:

- Multi-tenant environments
- Development workstations
- CI/CD pipelines where root access is restricted

5. Limitations:

- Some features may not be available or may require additional setup
- Performance overhead due to additional mapping layers
- Networking setup can be more complex

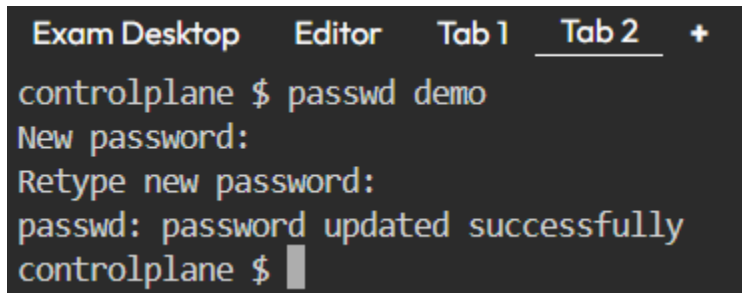
Step 1 – Install this 2 Package's

```
4 Clear
5 apt install dbus-user-session -y
6 apt install fuse-overlayfs
7 sleep
```

Step 2 – Disable Docker Service Socket

```
Exam Desktop  Editor  Tab 1  Tab 2  +
controlplane $ systemctl disable --now docker.service docker.socket
Removed /etc/systemd/system/sockets.target.wants/docker.socket.
Removed /etc/systemd/system/multi-user.target.wants/docker.service.
controlplane $ █
```

Step 3 – Create a new Normal user and set the Password

A terminal window titled 'Exam Desktop' with tabs 'Editor', 'Tab 1', and 'Tab 2'. The prompt is 'controlplane \$'. The user enters 'passwd demo', followed by 'New password:', 'Retype new password:', and 'passwd: password updated successfully'. The prompt returns to 'controlplane \$' with a cursor.

```
Exam Desktop  Editor  Tab 1  Tab 2  +
controlplane $ passwd demo
New password:
Retype new password:
passwd: password updated successfully
controlplane $
```

Step 4 – Switch to Normal User and move to Home Directory of the User

and Download the following package and perform this step's-

wget <https://get.docker.com/rootless>

sh rootless

exit

```
controlplane $ su demo
```

```
\h $ pwd
```

```
/root
```

```
\h $ cd ~
```

```
\h $ pwd
```

```
/home/demo
```

```
\h $ wget https://get.docker.com/rootless
```

```
--2024-07-31 15:49:44-- https://get.docker.com/rootless
```

```
Resolving get.docker.com (get.docker.com)... 108.158.61.39, 108.158.61.95, 108.158.61.42, ...
```

```
Connecting to get.docker.com (get.docker.com)|108.158.61.39|:443... connected.
```

```
HTTP request sent, awaiting response... 200 OK
```

```
Length: 8263 (8.1K) [text/plain]
```

```
Saving to: 'rootless'
```

```
rootless          100%[=====>] 8.07K --.-KB/s in 0s
```

```
2024-07-31 15:49:44 (153 MB/s) - 'rootless' saved [8263/8263]
```

```
\h $ sh rootless
```

```
# Installing stable version 27.1.1
```

```
# Executing docker rootless install script, commit: 0d6f72e
```

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed

100 70.4M	100 70.4M	0 0	83.2M 0	--:--:--	--:--:--	--:--:--	83.1M
-----------	-----------	-----	---------	----------	----------	----------	-------

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed

100 19.7M	100 19.7M	0 0	59.3M 0	--:--:--	--:--:--	--:--:--	59.3M
-----------	-----------	-----	---------	----------	----------	----------	-------

```
+ PATH=/home/demo/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin /home/demo/bin/dockerd-rootless-setupool.sh install
```

```
[INFO] Creating /home/demo/.config/systemd/user/docker.service
```

```
[INFO] starting systemd service docker.service
```

```
+ systemctl --user start docker.service
```

```
+ sleep 3
```

```
+ systemctl --user --no-pager --full status docker.service
```

```
● docker.service - Docker Application Container Engine (Rootless)
```

```
Loaded: loaded (/home/demo/.config/systemd/user/docker.service; disabled; vendor preset: enabled)
```

```
Active: active (running) since Wed 2024-07-31 15:50:08 UTC; 3s ago
```

Step 5 – Reload the Daemon And Switch to Normal User again.

Docker has become rootless and now we just have to run this command using Normal User

```
systemctl --user restart docker
```

```
systemctl --user enable docker
```

Now you can RUN, STOP , DELETE container's using Normal User.

```
Exam Desktop  Editor  Tab 1  Tab 2  +
controlplane $ systemctl daemon-reload
controlplane $ su demo
\h $ systemctl --user restart docker
\h $ systemctl --user enable docker
\h $ docker run -d httpd
Unable to find image 'httpd:latest' locally
latest: Pulling from library/httpd
efc2b5ad9eec: Pull complete
fce1785eb819: Pull complete
4f4fb700ef54: Pull complete
f214daa0692f: Pull complete
05383fd8b2b3: Pull complete
88ad12232aa1: Pull complete
Digest: sha256:932ac36fabe1d2103ed3edbe66224ed2afe0041b317bcd6f5d9be63594f0030
Status: Downloaded newer image for httpd:latest
c06dd161c7ab03d4329a2ca14c0822ab509a44b815d5fb8be27b35789b6e1d85
\h $ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
c06dd161c7ab   httpd     "httpd-foreground"      4 seconds ago Up 3 seconds  80/tcp       distracted_dijkstra
\h $
```

Step 6 – This is Additional Step and can be used to check the Rootless behavior of Docker and can also be used to Isolate two user.

Come back to Super User/Root User and Enable docker Service Socket

```
Exam Desktop  Editor  Tab 1  Tab 2  +
controlplane $ systemctl enable --now docker.service docker.socket
```

Run the Following Command using Root User

```
docker run -d nginx
```

```
docker ps -a
```

Now you will notice, Root user is only able to see the Container's which are started by him and he cannot see the container of Normal User we need to Switch into Normal User to see his Container's.

Hence we can also Isolate two user's in Docker using the Concept of Rootless Docker

```
Exam Desktop  Editor  Tab 1  Tab 2  +
controlplane $ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
e281a82dc1ed   nginx    "/docker-entrypoint...."  5 seconds ago Up 4 seconds  80/tcp       happy_ritchie
controlplane $ su demo
\h $ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
c06dd161c7ab   httpd     "httpd-foreground"      About a minute ago Exited (0)    About a minute ago        distracted_dijkstra
\h $
```

-----Thank You-----