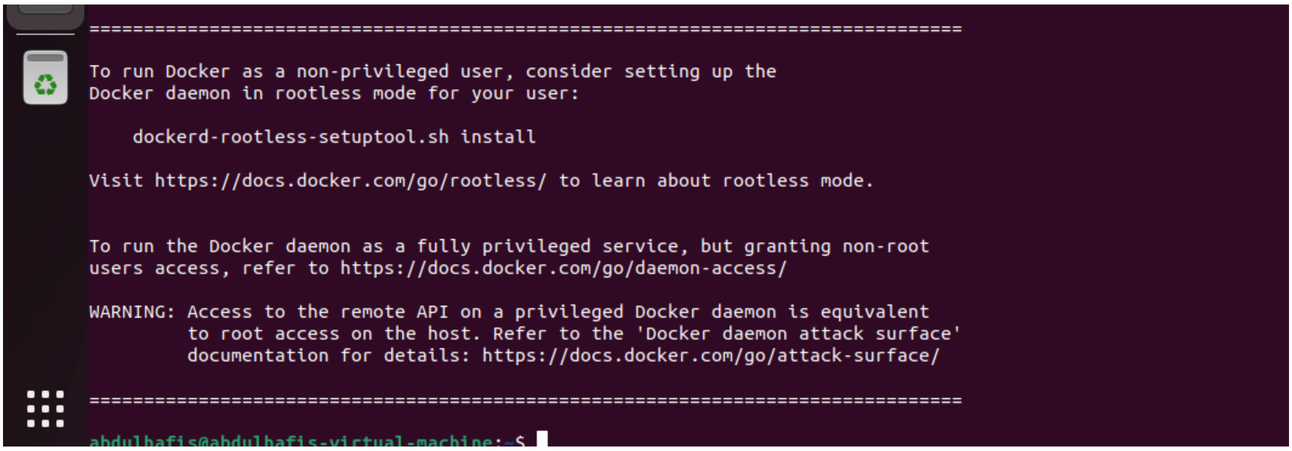
Connecting an External Database (PostgreSQL) to Keycloak

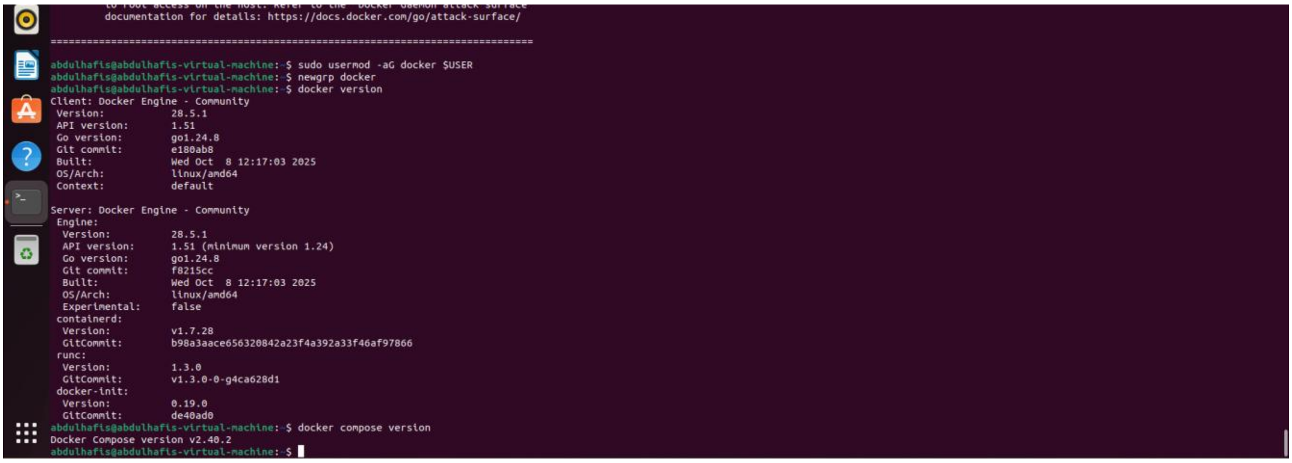
The aim of this exercise is to connect Keycloak to an external database. PostgreSQL was selected as the external database due to its reliability, scalability, and full compatibility with Keycloak's distribution. The setup was deployed using Docker Compose on Ubuntu Linux.

1. Installing Docker



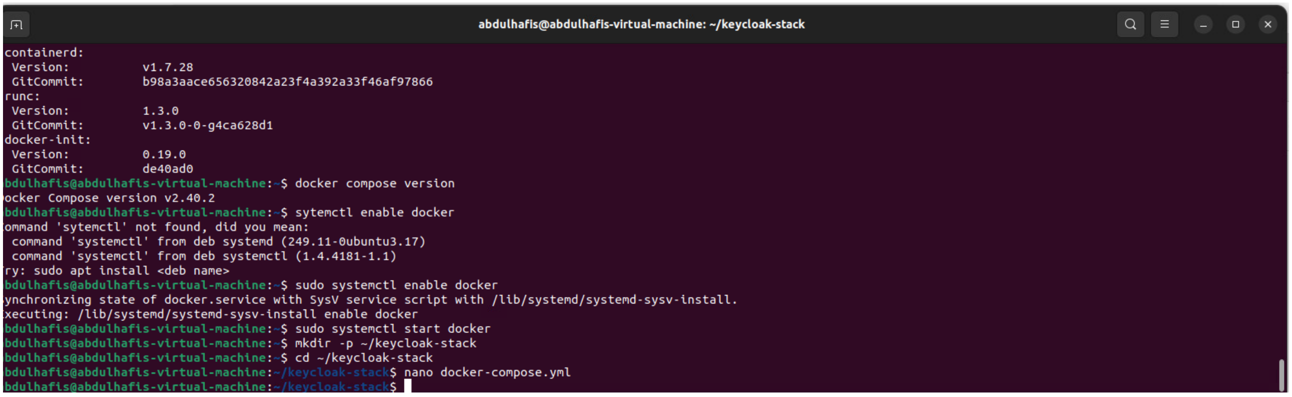
```
================================================================================

To run Docker as a non-privileged user, consider setting up the
Docker daemon in rootless mode for your user:

    dockerd-rootless-setuptool.sh install

Visit https://docs.docker.com/go/rootless/ to learn about rootless mode.


To run the Docker daemon as a fully privileged service, but granting non-root
users access, refer to https://docs.docker.com/go/daemon-access/

WARNING: Access to the remote API on a privileged Docker daemon is equivalent
         to root access on the host. Refer to the 'Docker daemon attack surface'
         documentation for details: https://docs.docker.com/go/attack-surface/

================================================================================

abdulhafis@abdulhafis-virtual-machine:~$
```

2. Confirming the installation of Docker



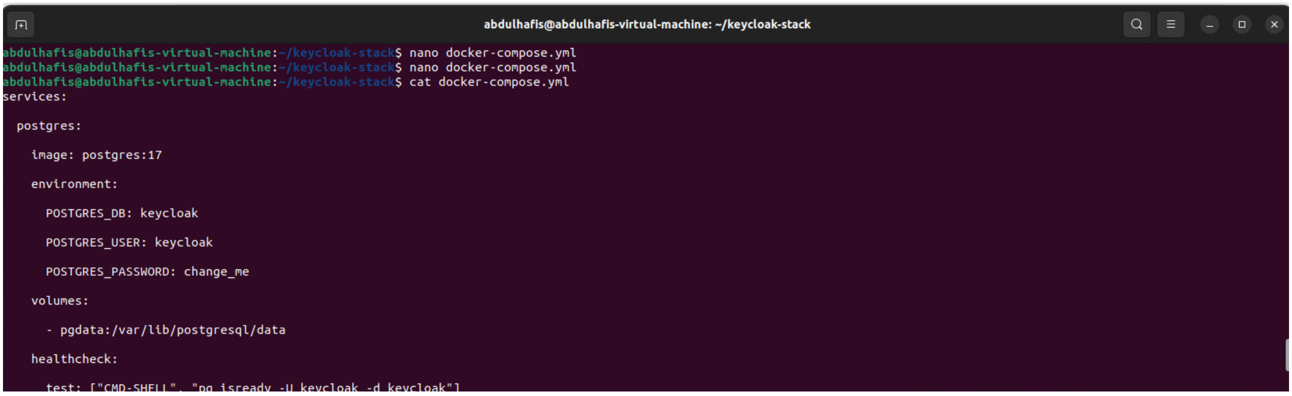
```
         to root access on the host. Refer to the 'Docker daemon attack surface'
         documentation for details: https://docs.docker.com/go/attack-surface/

================================================================================

abdulhafis@abdulhafis-virtual-machine:~$ sudo usermod -aG docker $USER
abdulhafis@abdulhafis-virtual-machine:~$ newgrp docker
abdulhafis@abdulhafis-virtual-machine:~$ docker version
Client: Docker Engine - Community
 Version:           28.5.1
 API version:       1.51
 Go version:        go1.24.8
 Git commit:        e180ab8
 Built:             Wed Oct  8 12:17:03 2025
 OS/Arch:           linux/amd64
 Context:           default

Server: Docker Engine - Community
 Engine:
  Version:          28.5.1
  API version:      1.51 (minimum version 1.24)
  Go version:       go1.24.8
  Git commit:       f8215cc
  Built:            Wed Oct  8 12:17:03 2025
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          v1.7.28
  GitCommit:        b98a3aace656320842a23f4a392a33f46af97866
 runc:
  Version:          1.3.0
  GitCommit:        v1.3.0-0-g4ca628d1
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
abdulhafis@abdulhafis-virtual-machine:~$ docker compose version
Docker Compose version v2.40.2
abdulhafis@abdulhafis-virtual-machine:~$
```

3. Creating project folder and file using a docker-compose.yml file that defined two services — postgres and keycloak



abdulhafis@abdulhafis-virtual-machine: ~/keycloak-stack

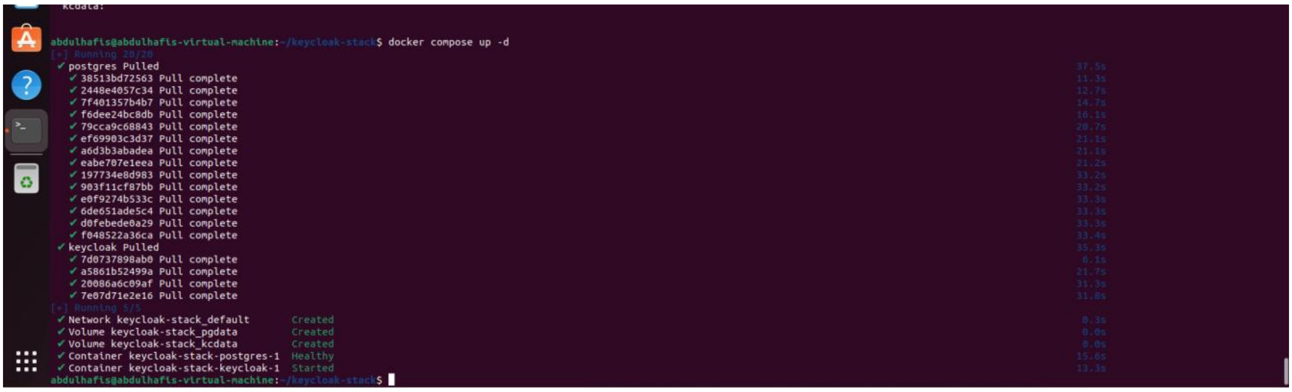
```
containerd:
 Version:          v1.7.28
 GitCommit:        b98a3aace656320842a23f4a392a33f46af97866
runc:
 Version:          1.3.0
 GitCommit:        v1.3.0-0-g4ca628d1
docker-init:
 Version:          0.19.0
 GitCommit:        de40ad0
bdulhafis@abdulhafis-virtual-machine:~$ docker compose version
ocker Compose version v2.40.2
bdulhafis@abdulhafis-virtual-machine:~$ sytemctl enable docker
ommand 'sytemctl' not found, did you mean:
 command 'systemctl' from deb systemd (249.11-0ubuntu3.17)
 command 'systemctl' from deb systemctl (1.4.4181-1.1)
ry: sudo apt install <deb name>
bdulhafis@abdulhafis-virtual-machine:~$ sudo systemctl enable docker
ynchronizing state of docker.service with SysV service script with /lib/systemd/systemd-sysv-install.
xecuting: /lib/systemd/systemd-sysv-install enable docker
bdulhafis@abdulhafis-virtual-machine:~$ sudo systemctl start docker
bdulhafis@abdulhafis-virtual-machine:~$ mkdir -p ~/keycloak-stack
bdulhafis@abdulhafis-virtual-machine:~$ cd ~/keycloak-stack
bdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ nano docker-compose.yml
bdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$
```



abdulhafis@abdulhafis-virtual-machine: ~/keycloak-stack

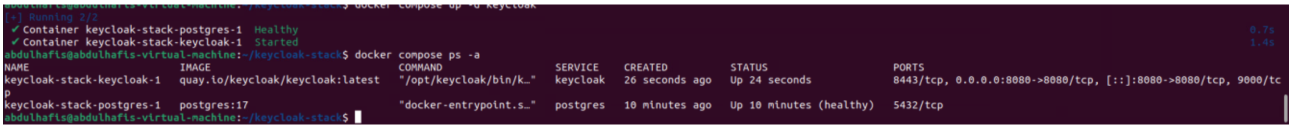
```
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ nano docker-compose.yml
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ nano docker-compose.yml
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ cat docker-compose.yml
services:

  postgres:

    image: postgres:17

    environment:

      POSTGRES_DB: keycloak

      POSTGRES_USER: keycloak

      POSTGRES_PASSWORD: change_me

    volumes:

      - pgdata:/var/lib/postgresql/data

    healthcheck:

      test: ["CMD-SHELL", "pg_isready -U keycloak -d keycloak"]
```

4. Docker downloading and extracting PostgreSQL and keycloak



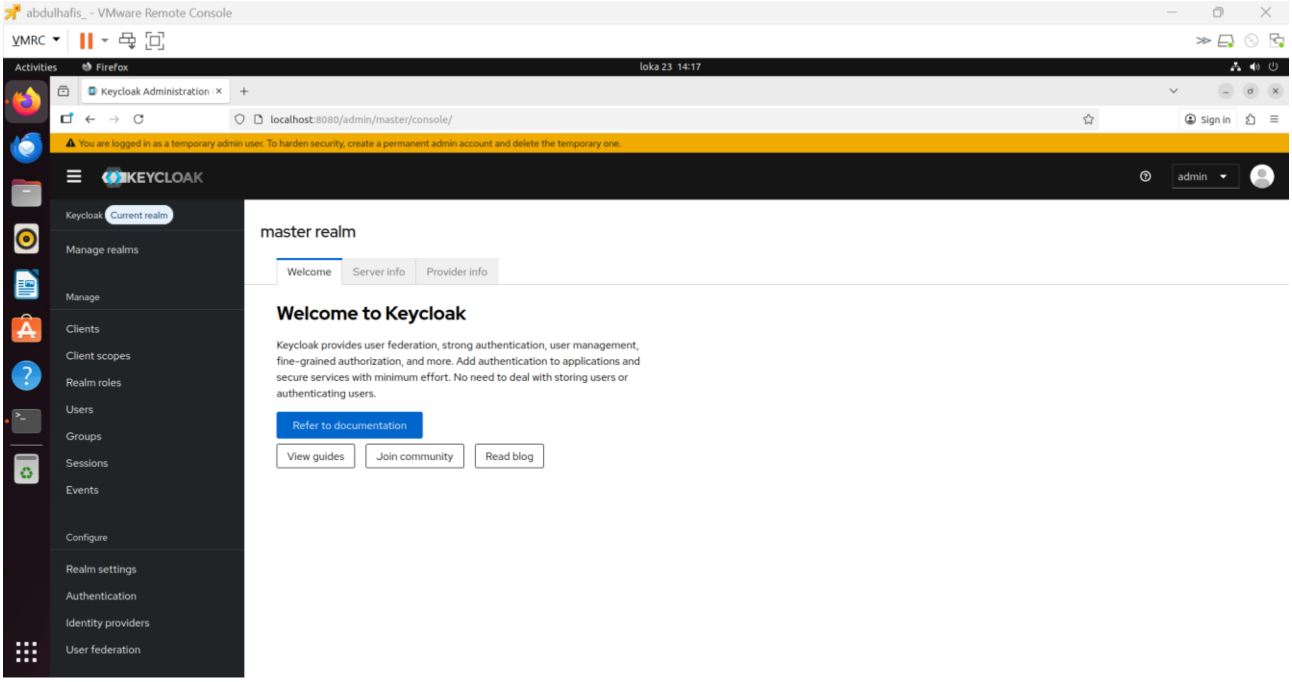
```
  kcdata:

abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ docker compose up -d
[+] Running 20/20
 ✔ postgres Pulled                                                                  37.5s
   ✔ 38513bd72563 Pull complete                                                     11.3s
   ✔ 2448e4057c34 Pull complete                                                     12.7s
   ✔ 7f401357b4b7 Pull complete                                                     14.7s
   ✔ f6dee24bc8db Pull complete                                                     16.1s
   ✔ 79cca9c68843 Pull complete                                                     20.7s
   ✔ ef69903c3d37 Pull complete                                                     21.1s
   ✔ a6d3b3abadea Pull complete                                                     21.1s
   ✔ eabe707e1eea Pull complete                                                     21.2s
   ✔ 197734e8d983 Pull complete                                                     33.2s
   ✔ 903f11cf87bb Pull complete                                                     33.2s
   ✔ e0f9274b533c Pull complete                                                     33.3s
   ✔ 6de651ade5c4 Pull complete                                                     33.3s
   ✔ d0febede0a29 Pull complete                                                     33.3s
   ✔ f048522a36ca Pull complete                                                     33.4s
 ✔ keycloak Pulled                                                                  35.3s
   ✔ 7d0737898ab0 Pull complete                                                      6.1s
   ✔ a5861b52499a Pull complete                                                     21.7s
   ✔ 20086a6c09af Pull complete                                                     31.3s
   ✔ 7e07d71e2e16 Pull complete                                                     31.8s
[+] Running 5/5
 ✔ Network keycloak-stack_default       Created                                      0.3s
 ✔ Volume keycloak-stack_pgdata         Created                                      0.0s
 ✔ Volume keycloak-stack_kcdata         Created                                      0.0s
 ✔ Container keycloak-stack-postgres-1  Healthy                                     15.6s
 ✔ Container keycloak-stack-keycloak-1  Started                                     13.3s
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$
```

5. Keycloak and PostgreSQL up and running



```
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ docker compose up -d keycloak
[+] Running 2/2
 ✔ Container keycloak-stack-postgres-1  Healthy                                                                              0.7s
 ✔ Container keycloak-stack-keycloak-1  Started                                                                              1.4s
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ docker compose ps -a
NAME                        IMAGE                            COMMAND                  SERVICE    CREATED          STATUS                    PORTS
keycloak-stack-keycloak-1   quay.io/keycloak/keycloak:latest   "/opt/keycloak/bin/k…"   keycloak   26 seconds ago   Up 24 seconds             8443/tcp, 0.0.0.0:8080->8080/tcp, [::]:8080->8080/tcp, 9000/tc
p
keycloak-stack-postgres-1   postgres:17                      "docker-entrypoint.s…"   postgres   10 minutes ago   Up 10 minutes (healthy)   5432/tcp
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$
```

6. **Accessed http://localhost:8080, logged in using the bootstrap admin credentials (admin/admin123), and confirmed successful connection to PostgreSQL by creating new realms, users, and clients**



abdulhafis_ - VMware Remote Console

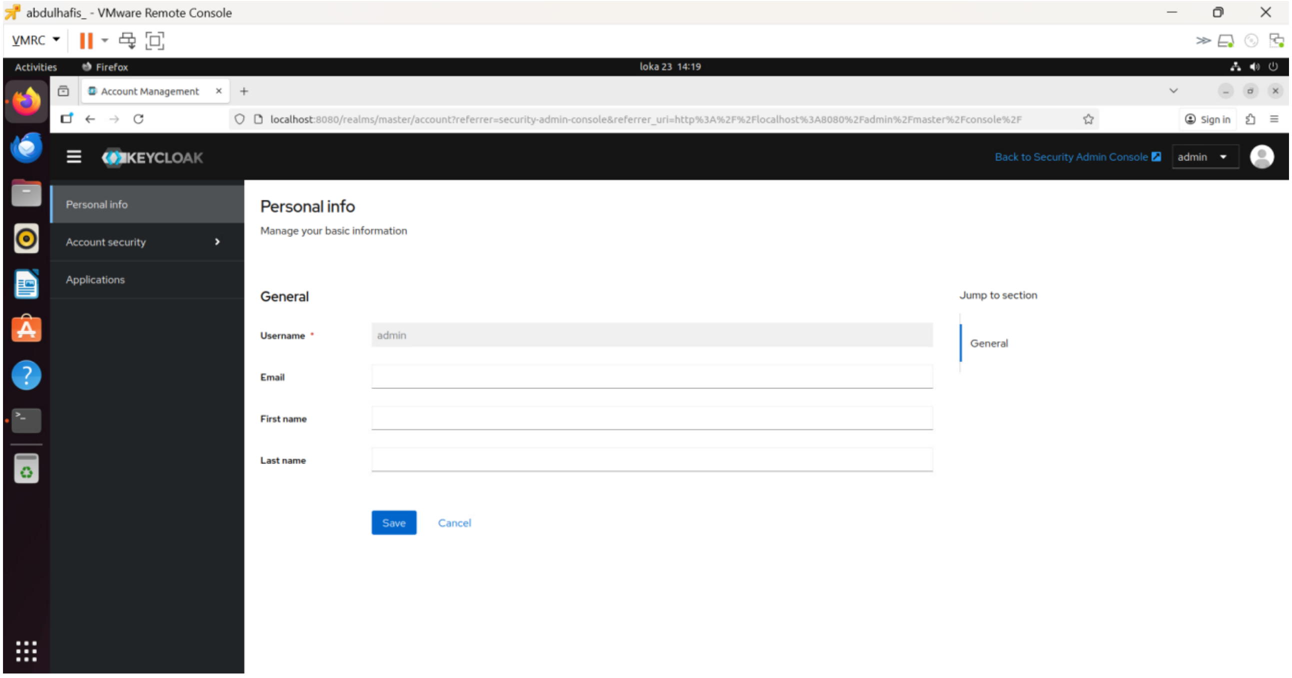
localhost:8080/admin/master/console/

You are logged in as a temporary admin user. To harden security, create a permanent admin account and delete the temporary one.

master realm

Welcome to Keycloak

Keycloak provides user federation, strong authentication, user management, fine-grained authorization, and more. Add authentication to applications and secure services with minimum effort. No need to deal with storing users or authenticating users.



abdulhafis_ - VMware Remote Console

localhost:8080/realms/master/account?referrer=security-admin-console&referrer_uri=http%3A%2F%2Flocalhost%3A8080%2Fadmin%2Fmaster%2Fconsole%2F

Personal info

Manage your basic information

General

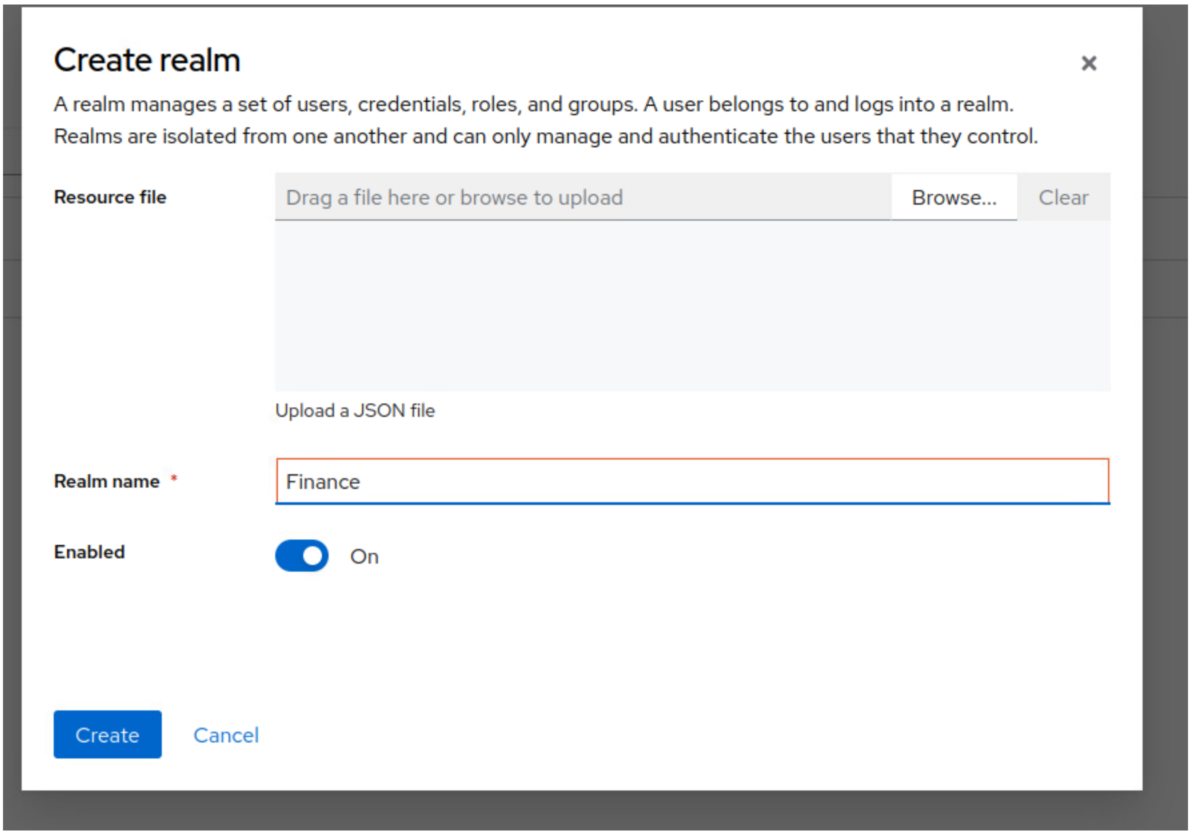
Username * admin

Email

First name

Last name

7. Creating a new realm – Finance



Create realm

A realm manages a set of users, credentials, roles, and groups. A user belongs to and logs into a realm. Realms are isolated from one another and can only manage and authenticate the users that they control.

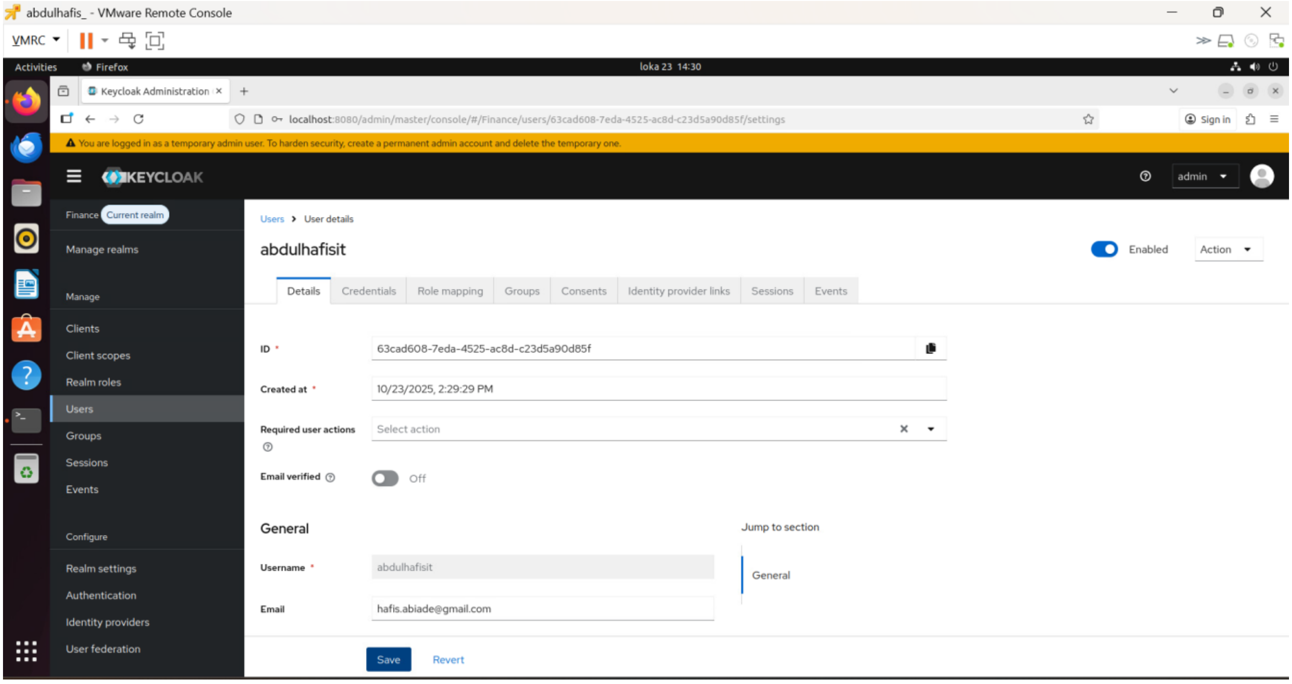
Resource file: Drag a file here or browse to upload

Upload a JSON file

Realm name *: Finance

Enabled: On

8. Creating a user in the Finance Realm



abdulhafis_ - VMware Remote Console

loka 23 14:30

Keycloak Administration

localhost:8080/admin/master/console/#/Finance/users/63cad608-7eda-4525-ac8d-c23d5a90d85f/settings

⚠ You are logged in as a temporary admin user. To harden security, create a permanent admin account and delete the temporary one.

Finance Current realm

Users › User details

abdulhafisit

Enabled

ID *: 63cad608-7eda-4525-ac8d-c23d5a90d85f

Created at *: 10/23/2025, 2:29:29 PM

Required user actions: Select action

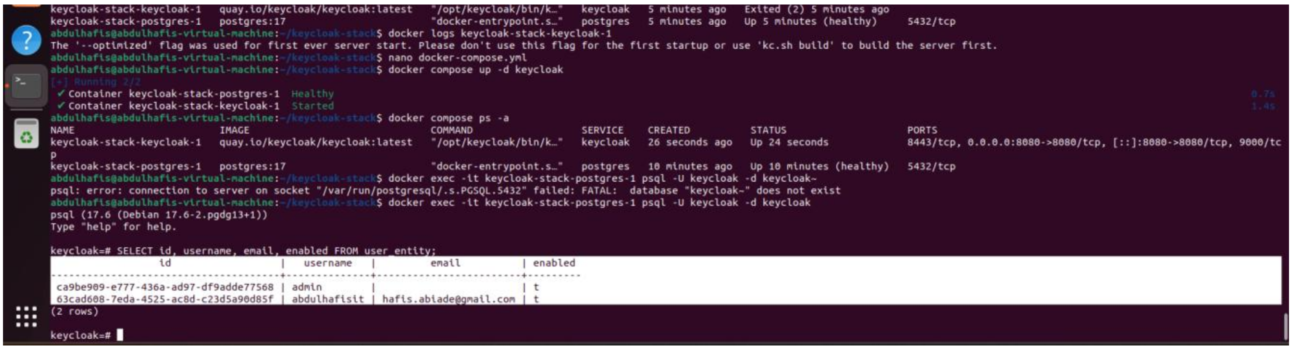
Email verified: Off

General

Username *: abdulhafisit

Email: hafis.abiade@gmail.com

9. **After creating realms and users in the Keycloak Admin Console, the PostgreSQL database confirms that all user and realm data are present.**



```
keycloak-stack-keycloak-1   quay.io/keycloak/keycloak:latest   "/opt/keycloak/bin/k…"   keycloak   5 minutes ago   Exited (2) 5 minutes ago
keycloak-stack-postgres-1   postgres:17                        "docker-entrypoint.s…"   postgres   5 minutes ago   Up 5 minutes (healthy)     5432/tcp
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ docker logs keycloak-stack-keycloak-1
The '--optimized' flag was used for first ever server start. Please don't use this flag for the first startup or use 'kc.sh build' to build the server first.
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ nano docker-compose.yml
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ docker compose up -d keycloak
[+] Running 2/2
 ✓ Container keycloak-stack-postgres-1  Healthy                                                                                          0.7s
 ✓ Container keycloak-stack-keycloak-1  Started                                                                                          1.4s
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ docker compose ps -a
NAME                        IMAGE                              COMMAND                  SERVICE    CREATED          STATUS                    PORTS
keycloak-stack-keycloak-1   quay.io/keycloak/keycloak:latest   "/opt/keycloak/bin/k…"   keycloak   26 seconds ago   Up 24 seconds             8443/tcp, 0.0.0.0:8080->8080/tcp, [::]:8080->8080/tcp, 9000/tc
p
keycloak-stack-postgres-1   postgres:17                        "docker-entrypoint.s…"   postgres   10 minutes ago   Up 10 minutes (healthy)   5432/tcp
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ docker exec -it keycloak-stack-postgres-1 psql -U keycloak -d keycloak~
psql: error: connection to server on socket "/var/run/postgresql/.s.PGSQL.5432" failed: FATAL:  database "keycloak~" does not exist
abdulhafis@abdulhafis-virtual-machine:~/keycloak-stack$ docker exec -it keycloak-stack-postgres-1 psql -U keycloak -d keycloak
psql (17.6 (Debian 17.6-2.pgdg13+1))
Type "help" for help.

keycloak=# SELECT id, username, email, enabled FROM user_entity;
                  id                  |   username   |         email          | enabled
--------------------------------------+--------------+------------------------+---------
 ca9be909-e777-436a-ad97-df9adde77568 | admin        |                        | t
 63cad608-7eda-4525-ac8d-c23d5a90d85f | abdulhafisit | hafis.abiade@gmail.com | t
(2 rows)

keycloak=# 
```