NetReach Setup for Virtual/On-Premises Environments

This quick guide provides instructions on how to run NetReach docker image on a virtual environment, like a virtual machine (VMs), or as a standalone on a physical computer (on-premises).

Minimum Requirements:

- **Operating System (OS):** Linux OS – e.g. Ubuntu 24.04 or latest, Debian 13
- **Processor:** 2 GHz dual-core processor.
- **RAM:** 4 GB (for a basic server installation, 8 GB recommended for desktop).
- **Storage:** 40 GB of hard drive space (or a virtual disk).
- **Virtualization Support:** VMWare ESXi, VMWare Workstation Pro, Proxmox VE, Microsoft Hyper-V, Virtual Box and etc.

NOTE: CPU and RAM utilization increase with more demanding workloads (storage, network utilization and the number of devices).

Step 1: Install Docker

Run the following inside the Ubuntu VM:

Bash:

sudo apt update

sudo apt install -y docker.io

sudo systemctl enable docker

sudo systemctl start docker

Optional: Allow running Docker as your user (not just sudo):

bash:

sudo usermod -aG docker $USER

Then log out and back in.

IMPORTANT: **sudo** works on most Linux distributions, but its behavior and availability can vary depending on the system's configuration.

Step 2: Pull NetReach Docker Image

- Authenticate to ACR
  - Login into WTI NetReach ACR (Contact WTI to obtain ACR credentials)

  bash:

  docker login netreachacr.azurecr.io

- Pull the NetReach Image

  bash:

  docker pull netreachacr.azurecr.io/netreach-image:<tag>

  - <tag>: The specific version or tag of the image (e.g., latest, v1.00). If no tag is specified, Docker defaults to latest.

Step 3: Run a Docker Container

Run NetReach

bash:

docker run -v /home/NetReachCloud:/home/NetReachCloud -d -p 3000:3000 netreachacr.azurecr.io/netreach-image:<tag>

- **<tag>**: The specific version or tag of the image (e.g., latest, v1.00). If no tag is specified, Docker defaults to latest.
- **-v <HOST_PATH>: <CONTAINER_PATH>**: Mounts a volume from the host into the container.
- **-d**: Runs the container in the background (detached mode).
- **-p <HOST_PORT>:<CONTAINER_PORT>**: Maps a port on the host machine to a port inside the container.

Check that it's working:

bash:

docker ps

(optional)

Run container after VM reboot:

bash:

systemctl enable docker.service

sudo docker update --restart unless-stopped <your_container_id_or_name>

From your browser (host or LAN), visit:

visit:

curl http://localhost:3000

You should see some output (Found or even HTML). If not, the backend isn't running or listening on that port/interface.

Step 4: Install Nginx on the Host VM

bash:

sudo apt update

sudo apt install nginx -y

Step 5: Configure Nginx as a Reverse Proxy

To configure Nginx as a reverse proxy over HTTPS using an IP address in a private LAN with Nginx

- Generate a Self-Signed Certificate for IP address

Use OpenSSL:

IP=192.168.1.100

sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /path/to/netreach-ip.key -out /path/to/netreach-ip.crt -subj "/CN=<IP>" -addext "subjectAltName=IP:<IP>"

This generates:

- /path/to/netreach-ip.key – private key
- /path/to/netreach-ip.crt – self-signed cert

- Nginx Configuration

nginx config (/etc/nginx/sites-available/netreachapp);

bash:

cd /etc/nginx/sites-available/

sudo vi netreachapp

e.g. – copy and paste into netreachapp:

```
server {
    listen 443 ssl;
    server_name <ip address>;     e.g. 192.168.1.100
    ssl_certificate /path/to/netreach-ip.crt;
    ssl_certificate_key /path/to/netreach-ip.key;
   ssl_protocols TLSv1.2 TLSv1.3;
    ssl_ciphers HIGH:!aNULL:!MD5;
    location / {
       proxy_pass http://localhost:3000;
       proxy_set_header Host $host;
       proxy_set_header X-Real-IP $remote_addr;
       proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
       proxy_set_header X-Forwarded-Proto $scheme;
    }
```

```
}
server {
  listen 80;
  server_name <ip address>;    e.g. 192.168.1.100
  return 301 https://$host$request_uri;
}
```

Enable the site and reload Nginx:

bash:

```
sudo ln -s /etc/nginx/sites-available/netreachapp /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl reload nginx
```

Access NetReach from Browser, using the admin default user account ***(username: netreach, password: netreach)***

Go to:

https://<your-netreach-ip-address>/login

Expect a **security warning** like:

“Your connection is not private”

That’s normal with self-signed certs.