Installation front One

Virtualisation, One, front, install

Le front Open Nebula permet de gérer la virtualisation.

Prérequis

Nous utilisons une installation minimale de Debian 11 sur une machine dédiée.

Configuration du dépôt APT

Nous installons les paquets dont nous avons besoin

```
apt install gpg wget apt-transport-https
```

Nous récupérons la clé publique du dépôt

```
wget -q -O- https://downloads.opennebula.io/repo/repo2.key | gpg --dearmor -
o /usr/share/keyrings/one-archive-keyring.gpg
```

Nous créons le fichier de dépôt

```
cat > /etc/apt/sources.list.d/opennebula.list <<EOF
##
# APT OpenNebula repository
##

deb [signed-by=/usr/share/keyrings/one-archive-keyring.gpg]
https://downloads.opennebula.io/repo/6.6/Debian/11 stable opennebula
EOF
```

Nous mettons à jour la liste des paquets

```
apt update
```

Installation base de données

```
apt install mariadb-server
```

Nous nous connectons au client MariaDB

```
mysql
```

```
CREATE USER 'oneadmin' IDENTIFIED BY '<thepassword>';
GRANT ALL PRIVILEGES ON opennebula.* TO 'oneadmin';
```

Nous sortons du client

```
QUIT
```

Nous configurons l'isolation des transactions

```
SET GLOBAL TRANSACTION ISOLATION LEVEL READ COMMITTED;
```

Installation OpenNebula

Nous installons les paquets OpenNebula à proprement parler

```
apt install opennebula opennebula-sunstone opennebula-fireedge opennebula-
gate opennebula-flow opennebula-provision
```

Configuration OpenNebula

Nous ajoutons la configuration de la base de données dans le fichier /etc/one/oned.conf

```
vi /etc/one/oned.conf
```

Nous remplaçons la configuration avec sqlite par

```
DB = [ BACKEND = "mysql",
       SERVER  = "localhost",
       PORT    = 0,
       USER    = "oneadmin",
       PASSWD  = "<thepassword>",
       DB_NAME = "opennebula",
       CONNECTIONS = 25,
       COMPARE_BINARY = "no" ]
```

Nous configurons FireEdge

```
vi /etc/one/sunstone-server.conf
```

```
:public_fireedge_endpoint: http://one.example.com:443
```

Nous configurons onegate server

```
vi /etc/one/onegate-server.conf
```

```
:host: 0.0.0.0
```

Nous configurons onegate endpoint

```
vi /etc/one/oned.conf
```

```
ONEGATE_ENDPOINT = "http://one.example.com:5030"
```

Démarrage et activation des services

Nous démarrons les services

```
systemctl start opennebula opennebula-sunstone opennebula-fireedge
opennebula-gate opennebula-flow
```

Nous activons les services

```
systemctl enable opennebula opennebula-sunstone opennebula-fireedge
opennebula-gate opennebula-flow
```

Proxy Nginx

Nous installons Nginx

```
apt install nginx
```

Nous configurons le vHost

```
vi /etc/nginx/site-available/one.example.com.conf
```

```
##
# Nginx vHost
# Application: OpenNebula Sunstone
# Sources:
# https://github.com/storpool/addon-vnctoken/blob/master/vnctoken.conf.nginx
#
https://forum.opennebula.io/t/fireedge-public-endpoint-is-not-working/9611/5
##

# No squealing.
server_tokens off;

# OpenNebula Sunstone upstream
upstream sunstone {
  server 127.0.0.1:9869;
}
```

```
# OpenNebula fireedge upstream
upstream fireedge {
  server 127.0.0.1:2616;
}

# OpenNebula websocketproxy upstream
upstream websocketproxy {
  server 127.0.0.1:29876;
}

# HTTP virtual host, redirect to HTTPS
server {
    listen 80;
    server_name one.example.com;
    return 301 https://one.example.com;
}

#
# Example Sunstone configuration (/etc/one/sunstone-server.conf)
#
#:vnc_proxy_port: 127.0.0.1:29876
#:vnc_proxy_support_wss: only
#:vnc_proxy_cert: /etc/letsencrypt/live/frontend/fullchain.pem
#:vnc_proxy_key: /etc/letsencrypt/live/frontend/privkey.pem
#:vnc_proxy_ipv6: false
#:vnc_request_password: false
#:vnc_client_port: 443

# HTTPS virtual host, proxy to Sunstone
server {
    listen 443 ssl;
    server_name one.example.com;
    ssl_certificate     /etc/ssl/certs/one.example.com.crt;
    ssl_certificate_key /etc/ssl/private/one.example.com.key;

    location / {
        proxy_pass http://sunstone;
        proxy_redirect     off;
        log_not_found      off;
        proxy_set_header   X-Real-IP $remote_addr;
        proxy_set_header   Host $http_host;
        proxy_set_header   X-Forwarded-FOR $proxy_add_x_forwarded_for;
    }
    location /websockify {
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header x-forwarded-proto  $scheme;
        proxy_set_header Host $host;
```

```
        proxy_buffering off;
        proxy_http_version 1.1;
        proxy_read_timeout 86400;
        proxy_pass https://websocketproxy;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "upgrade";
    }
}

# HTTPS virtual host, proxy to FireEdge
server {
    listen 443 ssl;
    server_name fireedge.example.com;
    ssl_certificate     /etc/ssl/certs/fireedge.example.com.crt;
    ssl_certificate_key /etc/ssl/private/fireedge.example.com.key;

    location / {
        proxy_pass http://fireedge;
        proxy_redirect     off;
        log_not_found      off;
        proxy_set_header   X-Real-IP $remote_addr;
        proxy_set_header   Host $http_host;
        proxy_set_header   X-Forwarded-FOR $proxy_add_x_forwarded_for;
    }
}
```

```
cd /etc/nginx/site-enabled
ln -s /etc/nginx/site-available/one.example.com.conf
```

Nous testons la configuration et redémarrons

```
nginx -t
systemctl reload nginx.service
```

Liens

- documentation Open Nebula
- configuration MariaDB
- single frontend installation