

Mise en place du VLAN entre les deux serveurs

Notes : le terme région est à rapproché de "datacenter". Mais un datacenter peut habiter plusieurs codes régions (ex. UK et UK1 pour le datacenter de Londres).

Attacher le VPN aux instances

- Vérifier que vous avez bien préalablement créer un utilisateur OpenStack avec le rôle "Administrator"
- Se rendre sur l'interface Manager OVH > Public Cloud > Management Interfaces > Horizon
 - Utiliser l'identifiant (hash) et le mot de passe générés automatiquement lors de la création du compte Administrateur : Manager OVH > Public Cloud > Project management > Users & roles
- Dans l'interface d'OpenStack Horizon, sélectionner votre projet et région (GRA7 ou UK1 par exemple) dans le bandeau, puis se rendre dans "Compute" > "Instances"
 - Sélectionner une instance ("sinp-<region>-web" ou "sinp-<region>-db") et choisir l'action "Attach interface"
 - The way to specify an interface : by network
 - Network : sinp-<region>-vpn (10.0.0.0/16)(nom du vRack préalablement donné)
 - IP Address : 10.0.1.10 (pour sinp-<region>-db), 10.0.1.20 (pour sinp-<region>-web), 10.0.1.30 (pour sinp-<region>-bcp.
 - Gateway : 10.0.0.1

Fixer l'IP privée sur les instances

Debian 12

- Ressource : [Configurer une adresse IP en alias](#)
- Se connecter ensuite aux instances. Pour vous connecter, si c'est la première fois, utiliser : `ssh debian@<ipv4-instance>`
- Une fois connecté, afficher les interfaces réseau du serveur : `ip a`
 - Normalement, l'interface concernant le réseau public (IPv4 public) est sur "ens3" et l'interface concernant le réseau privé (IPv4 privé) est sur "ens7".
- Désactiver la configuration automatique du réseau, en créant le fichier suivant : `vi /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg`
 - Ajouter le contenu suivant :

```
network: {config: disabled}
```

- **Note :** la création de ce fichier de configuration empêche l'exécution automatique des modifications apportées à la configuration de votre réseau.
- Modifier le fichier suivant : `vi /etc/netplan/50-cloud-init.yaml`
- Ajouter la nouvelle interface réseau correspondant au VPN avec le rang d'adresses à utiliser. Exemple de contenu du fichier :

```
network:
  version: 2
  ethernets:
    ens3:
      dhcp4: true
      match:
        macaddress: fa:.....:a0
      mtu: 1500
      set-name: ens3
    ens7:
      dhcp4: true
      match:
        macaddress: fa:.....:e9
      mtu: 1500
      set-name: ens7
      addresses:
        - 10.0.1.30/16
```

- Tester vos changements avec la commande : `netplan try`
- Appliquer les changements avec la commande : `netplan apply`
- Vérifier l'application des modification avec : `ip a`
- Redémarrer votre instance avec la commande : `systemctl start reboot.target`
 - Cela vous déconnecte de l'instance, c'est normal.
 - Vous pourrez vous y re-connecter avec SSH au bout de quelques dizaines de secondes.

Debian 11 et inférieurs

- Se connecter ensuite aux instances, puis modifier le fichier : `vi /etc/network/interfaces.d/50-cloud-init.cfg` ou `vi /etc/network/interfaces.d/50-cloud-init`
 - Pour vous connecter, si c'est la première fois, utiliser : `ssh debian@<ip>`
 - Afficher les interfaces réseau du serveur : `ip a`
 - Sur Debian 11, les interfaces ont changées de nom "eth0" devient "ens3" et l'interface "eth1" devient "ens7"
 - Dans le fichier `50-cloud-init`, une interface "ens3" devrait être présente avec l'IP publique et une interface "ens7" doit y être ajouté avec le contenu suivant :
 - Pour `sinp-<region>-web`

```
auto ens7
iface ens7 inet static
    address 10.0.1.10
    netmask 255.255.0.0
    mtu 9000
```

- Pour `sinp-<region>-db`

```
auto ens7
iface ens7 inet static
    address 10.0.1.20
    netmask 255.255.0.0
```

```
mtu 9000
```

- Pour `sinp-<region>-bkp`

```
auto ens7
iface ens7 inet static
    address 10.0.1.30
    netmask 255.255.0.0
    mtu 9000
```

- Si d'autres IP en 10.0... sont attachées à votre instance vous pouvez les détacher via l'interface d'Horizon.
- Redémarrer votre instance avec la commande : `reboot`
 - Cela vous déconnecte de l'instance, c'est normal.
 - Vous pourrez vous y re-connecter avec SSH au bout de quelques dizaines de secondes.

Migration vers Netplan.io

- Ressources : https://pedroagrodrigues.com/posts/Debian_To_Netplan/
- Les serveurs mis jour vers Debian 12 ne sont pas migrés automatiquement vers Netplan.
- Collecter les informations :
 - Afficher les noms, adresses MAC, IPv4 concernant les 2 interfaces réseaux avec : `ip a`
 - Afficher les IP des serveurs DNS : `cat /etc/resolv.conf`
 - Note: avec netplan c'est une commande : `resolvectl status`
 - Noter les IPv6 du serveur et de sa gateway indiquées sur l'interface OVH Public Cloud de l'instance concernée.
- Installer les paquets suivant : `apt update && apt install netplan.io systemd-resolved`
- Activer les services Systemd :

```
systemctl unmask systemd-networkd.service;
systemctl unmask systemd-resolved.service;
systemctl enable systemd-networkd.service;
systemctl mask networking;
systemctl enable systemd-resolved.service;
```

- Essayer de migrer avec : `ENABLE_TEST_COMMANDS=1 netplan migrate && sudo netplan try`
 - En cas de message d'erreur concernant "mtu", essayer de commenter les lignes en question dans le fichier `/etc/network/interfaces.d/50-cloud-init.cfg` puis essayer à nouveau
 - Renommer le fichier généré : `mv /etc/netplan/10-ifupdown.yaml /etc/netplan/50-cloud-init.yaml`
 - Corriger les droits du fichier : `chmod 600 /etc/netplan/*`
 - Compléter le fichier qui devrait ressembler à ceci:

```
network:
  version: 2
  ethernets:
    eth0:
      addresses:
```

```
- <ipv6-du-serveur>/56
dhcp4: true
match:
  macaddress: <adresse-mac-de-eth0>
mtu: 1500
nameservers:
  addresses:
    - 213.186.33.99
    - 0.0.0.0
  search: []
routes:
-   to: ::/0
  via: <ipv6-gateway-serveur>
set-name: eth0
eth1:
addresses:
- <ipv4-privée-du-serveur>/16
match:
  macaddress: <adresse-mac-de-eth1>
mtu: 9000
nameservers:
  addresses:
    - 213.186.33.99
    - 0.0.0.0
  search: []
set-name: eth1
```

- Attention :
 - les interfaces réseaux peuvent être nommé différemment. Utiliser les noms fournis par ip a.
 - si vous changer les noms des interfaces, s'assurer de modifier les interfaces des zones "public" et "internal" du parefeu.
- Désactiver la configuration automatique du réseau, en créant le fichier suivant : `vi /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg`
 - Ajouter le contenu suivant :

```
network: {config: disabled}
```

- Tester la configuration : `netplan try`
- Appliquer la configuration : `netplan apply`
- Redémarrer : `reboot`
- Nettoyer : `apt purge ifupdown resolvconf -y && rm -rf /etc/network`
- Vérifier la présence du lien suivant : `ll /etc/resolv.conf`
 - Si absent, le créer : `ln -sf /run/systemd/resolve/stub-resolv.conf /etc/resolv.conf`
- Si tout est ok, il doit être possible de joindre un domaine externe : `ping google.fr`

Notes

- **Connexion impossible en SSH** : si pour une raison ou une autre, vous ne pouvez plus vous connecter à une instance via SSH, il est possible de passer par l'interface du Manager d'OVH ou

l'interface d'Horizon. Ces 2 interfaces permettent d'accéder à une Console VNC. Attention, la console est en Qwerty par défaut, ce qui peut compliquer la saisie du mot de passe...

- **Log *DHCPDISCOVER* avec Debian 10** : Pour éviter les logs de *DHCPDISCOVER* dans le fichier de log `/var/log/syslog`, commenter les lignes suivantes dans le fichier : `vi /etc/network/interfaces`

`/etc/network/interfaces`

```
allow-hotplug eth1
iface eth1 inet dhcp

allow-hotplug eth2
iface eth2 inet dhcp
```

Ajouter une nouvelle région à un VLAN existant

Pour réaliser cette opération, il est nécessaire de passer par l'API OVH v6. Cette API peut s'utiliser via l'interface web disponible :

- Se rendre sur l'interface web de l'API : <https://api.ovh.com/console/>
- Cliquer en haut à droite sur "login" pour se connecter avec son utilisateur OVH permettant d'accéder au projet Public Cloud concernant dans le Manager OVH.
- Ouvrir le web service : GET `/cloud/project/{serviceName}/network/private`
 - Indiquer l'identifiant du projet Public Cloud dans le champ "*serviceName*" (à récupérer sous le nom du projet Public Cloud en haut à gauche dans l'interface du Manager d'OVH).
 - Cliquer sur le bouton "Execute"
 - Récupérer l'identifiant (propriété "id") du VLAN. Format : `pn-10xxxxx_0`
 - La propriété "*regions*" devrait à ce stade n'indiquer que "GRA7"
- Ouvrir maintenant le web service : POST `/cloud/project/{serviceName}/network/private/{networkId}/region`
 - Remplir le champ "*serviceName*" comme précédemment
 - Remplir le champ "*networkId*" avec l'identifiant du VLAN récupéré précédemment.
 - Remplir le champ "*ProjectNetworkPrivateRegionCreation*" > "*region*" avec le code du datacentre où l'on veut pouvoir accéder à ce VLAN. Dans notre cas : UK1
 - Cliquer sur le bouton "Execute"
- Ré-ouvrir le web service : GET `/cloud/project/{serviceName}/network/private`
 - Cliquer à nouveau sur le bouton "Execute"
 - Vérifier que la nouvelle région, dans notre cas UK1, apparaît bien dans la liste.
- Pour rendre visible ce réseau sur le nouveau Datacentre dans l'interface du Manager d'OVH, il peut être nécessaire de se déconnecter et se connecter à nouveau.
- **Notes** : la création de l'instance en l'associant directement à ce réseau n'a pas fonctionné. Erreur obtenue : `Network ... : requires a subnet in order to boot instances on..` L'instance a été créé sans réseau lié. Utilisation d'OpenStack Horizon pour associer le réseau à l'instance.

Ajouter un sous-réseau à un VLAN étendu à une nouvelle région

Lorsque le VLAN existant est étendu au nouveau datacentre, ce dernier ne possède pas de sous-réseau. Il faut donc lui associer le même sous-réseau que celui présent dans le datacentre principal. Pour réaliser cela nous passons par la ligne de commande :

- Cela implique de mettre en place un environnement OpenStack comme indiqué dans [la documentation d'installation du serveur BKP](#).
- Récupérer depuis l'interface d'Horizon sur le **datacentre principal**, dans notre cas GRA7 : le nom du réseau (Ex. `sinp-aura-vpn`), le nom du sous-réseau (Ex. : `(d977881c-84cc)`) et le rang d'IPs du sous-réseau (Ex. `10.0.0.0/16`).
- En local, se connecter sur le datacentre secondaire ou se trouve "bcp-srv" : `cd sinp-aura-UK1 ; source openrc.sh`
- Créer le sous-réseau : `openstack subnet create --network "<nom-reseau>" --subnet-range "<rang-ip-sous-reseau>" "<nom-sous-reseau>"`
 - Ex. : `openstack subnet create --network "sinp-aura-vpn" --subnet-range "10.0.0.0/16" "(d977881c-84cc)"`
- Il semble aussi nécessaire de modifier le sous-réseau des 2 datacentres pour utiliser la même IP dans le champ *Gateway*. Par exemple : `10.0.0.1`
 - Les 2 sous-réseaux devraient aussi avoir : le même nom et le même ensemble d'IP.
- Attacher ensuite le réseau sur une instance (voir ci-dessus) puis tester la connectivité entre les différentes instances : `ping 10.0.1.20` et `ping 10.0.1.10`

From:
<http://sinp-wiki.cbn-alpin.fr/> - **CBNA SINP**

Permanent link:
<http://sinp-wiki.cbn-alpin.fr/serveurs/installation/vlan?rev=1707837632>

Last update: **2024/02/13 15:20**

