Routage inter-VLAN et protocole PPPoE dans un contexte cloud

Philippe Latu
philippe.latu(at)inetdoc.net

https://www.inetdoc.net

Résumé

L'évolution des réseaux étendus (WAN vers la fibre optique a entrainé un changement radical au niveau de la couche liaison. Le format de trame historique HDLC est remplacé par Ethernet qui devient universel. Le hic, c'est que par définition, Ethernet est un réseau de diffusion. C'est là que le protocole PPPoE intervient. Il permet de passer d'un réseau de diffusion à un fonctionnement point à point caractéristique des réseaux étendus.

Les manipulations présentées dans ces travaux pratiques illustrent l'interconnexion entre réseaux locaux et réseaux étendus dans un contexte de type Cloud IAAS (Infrastructure As A Service).

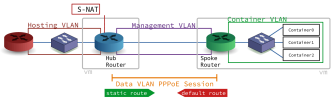


Table des matières

1. Copyright et Licence 1
2. Interface Ethernet & protocole PPP 2
3. Topologies logiques et virtuelles 3
4. Raccordement au commutateur de distribution 4
5. Routeur Hub (bleu) 4
    5.1. Configuration des interfaces du routeur 5
    5.2. Activation de la fonction routage 7
    5.3. Activation de la traduction d'adresses 8
    5.4. Activation du protocole PPPoE côté réseau étendu 9
    5.5. Ajout des routes statiques vers le réseau des conteneurs 13
6. Routeur Spoke (vert) 14
    6.1. Configuration des interfaces du routeur 14
    6.2. Activation de la fonction routage 15
    6.3. Activation du protocole PPP dans le VLAN orange 16
    6.4. Activation du commutateur virtuel asw-host 19
    6.5. Activation de la configuration IPv6 automatique pour le réseau de conteneurs 20
    6.6. Validation des routes par défaut vers le réseau opérateur 21
    6.7. Installation du gestionnaire de conteneurs LXD 23
    6.8. Configuration du gestionnaire de conteneurs LXD 23
7. Trace d'une transaction complète PPPoE 26
    7.1. Routeur Spoke (vert) 26
    7.2. Routeur Hub (bleu) 28

1. Copyright et Licence

Méta-information

Ce document est écrit avec DocBook XML sur un système Debian GNU/Linux. Il est disponible en version imprimable au format PDF : `interco.pppoe-cloud.qa.pdf`.

2. Interface Ethernet & protocole PPP

Avec la généralisation de l'utilisation de la fibre optique dans les réseaux étendus, le format de trame historique HDLC est progressivement abandonné. Il faut dire que ce format de trame date du développement des liaisons séries asynchrones. Aujourd'hui, les liaisons sur fibres optiques sont Full-Duplex et on ne se préoccupe plus de synchronisation au niveau de la couche liaison de données. Le format de trame Ethernet devient ainsi une référence universelle.

Le protocole PPP offre depuis l'origine une configuration indépendante de la technologie du réseau étendu.

L'association entre trame Ethernet et PPP se fait grâce à un autre protocole baptisé PPPoE. Ce dernier permet d'encapsuler des trames PPP dans des trames Ethernet. Il est décrit à la page Point-to-point protocol over Ethernet qui permet de traiter les questions ci-après.

Q1. Quelle est la raison de l'ajout d'un nouveau protocole entre Ethernet et PPP ?

Consulter la page Point-to-point protocol over Ethernet.

Le protocole PPP a été conçu pour fonctionner sur des liaisons point-à-point alors qu'un réseau local Ethernet est par définition un réseau de diffusion.

Sur un réseau de diffusion, le canal de transmission est partagé entre tous les hôtes qui accèdent au canal. Il a donc été nécessaire d'introduire un mécanisme de découverte des deux extrémités en communication avant de lancer les opérations du protocole PPP.

Q2. Donner la liste des messages de découverte et de session PPPoE en précisant qui est l'initiative de cette découverte.

Consulter la page Point-to-point protocol over Ethernet.

- Client to server: Initiation (PADI)
- Server to client: Offer (PADO)
- Client to server: request (PADR)
- Server to client: session-confirmation (PADS)
- Either end to other end: termination (PADT)

Q3. Quels sont les autres mécanismes de découverte de voisins connus dans un réseau local Ethernet ?

Voici la liste des «grands classiques».

- Address Resolution Protocol (ARP).

  Quelle est l'adresse MAC d'un hôte dont on connaît l'adresse IPv4 ?

- Neighbor Discovery Protocol (NDP).

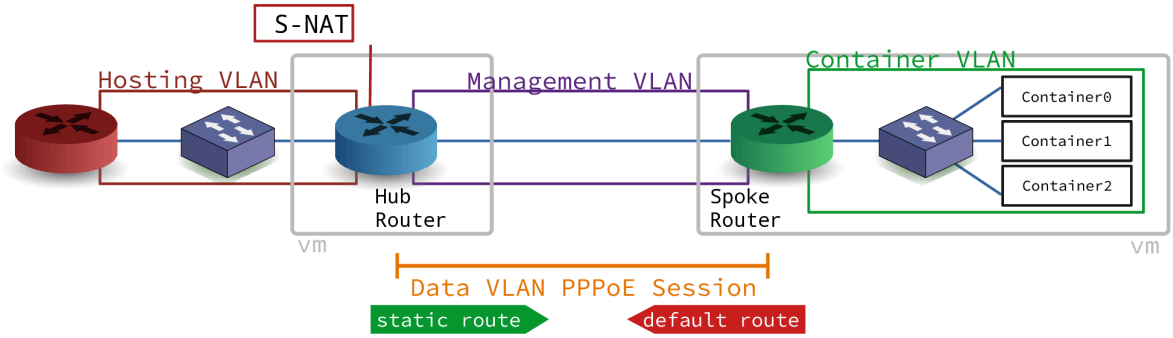
Ce protocole est associé à IPv6. Il définit 5 messages ICMPv6 qui couvrent les mêmes opérations que celles réalisées par le protocole ARP sans avoir recours à la diffusion et qui ajoutent de nouvelles fonctions.

- Multicast DNS (mDNS) ou Bonjour.

  Ce protocole entre dans la famille zeroconf qui a pour but d'annoncer et de fournir des éléments de configuration aux hôtes du réseau sans faire appel à une infrastructure de services de la couche application tels que DNS et DHCP.

3. Topologies logiques et virtuelles

La représentation de la topologie logique ci-dessous montre que le routeur de couleur bleue assure l'interconnexion entre un réseau d'infrastructure opérateur appelé Hosting VLAN et un réseau étendu qui dessert un site distant. Ce site distant est représenté par le routeur de couleur verte. Les services hébergés sur le site distant appartiennent au réseau appelé Container VLAN. Sur le réseau étendu on distingue deux autres VLANs : le VLAN violet appelé Management VLAN est utilisé pour la supervision et le VLAN orange Data VLAN est utilisé pour l'acheminement des données du site distant. Ce dernier réseau à la particularité d'utiliser une session PPPoE entre les routeurs bleu et vert. Les deux rectangles en gris “matérialisent“ les machines virtuelles qui sont utilisées pour les manipulations.

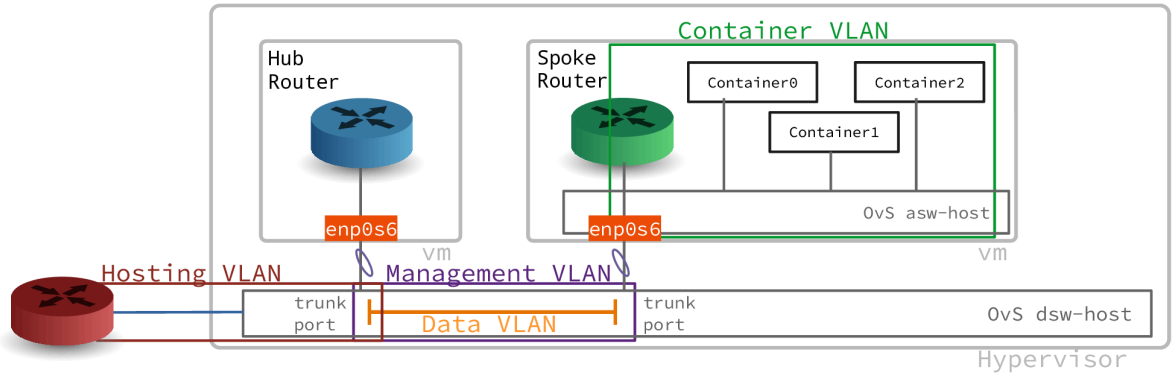


Topologie logique

La représentation de la topologie vue sous l'angle de l'hébergement sur un système hôte hyperviseur montre que le VLAN de couleur verte appelé Container VLAN n'est visible qu'à l'intérieur de la machine virtuelle qui représente le site distant. Ce VLAN est isolé et ses préfixes réseau IPv4 et IPv6 doivent être ***routés***. C'est la raison pour laquelle la machine virtuelle du site distant dispose de son propre commutateur : `asw-host`.

Tous les autres VLANs sont présents sur le commutateur virtuel de couche distribution appelé `dsw-host`. Ce commutateur appartient au système hôte. Il assure le raccordement entre les réseaux physiques et virtualisés. Chacun des routeurs bleu et vert est raccordé avec un lien unique (port en mode trunk) sur lequel le trafic des VLANs doit transiter.

Côté conteneurs, le raccordement au commutateur `asw-host` sera assuré automatiquement par le gestionnaire LXD.



Topologie hébergée

Voici le plan d'adressage utilisé pour la maquette qui sert à la rédaction de ce support de travaux pratiques.

Tableau 1. Affectation des numéros de VLANs, des adresses de passerelle et des authentifiants

Planète	VLAN	Numéro	Type	Adresse
Maquette	Rouge	360	Passerelle	`192.168.104.130/29` `2001:678:3fc:168::1/64`
	Violet	440	Adresse	`fe80:1b8::1`
				`fe80:1b8::2`
	Orange	441	Point à point	`10.4.41.1:10.4.41.2`
			Authentifiants	`green / 5p0k3`
	Vert	40	Passerelle	`203.0.113.1/24` `fda0:7a62:28::1/64`

4. Raccordement au commutateur de distribution

Dans cette section, on étudie le raccordement des deux machines virtuelles au commutateur de distribution sur le système hôte.

Q4. Comment contrôler la configuration des ports du commutateur de distribution sur le système hôte ?

Le commutateur virtuel implanté sur le système hôte est géré par Open vSwitch. On fait donc appel à la commande ovs-vsctl pour afficher la configuration des ports. Le mot clé dans le cas de cette question est `vlan_mode`.

- Pour le port de raccordement du routeur bleu, on obtient :

```
sudo ovs-vsctl list port tap200 | grep vlan_mode
vlan_mode           : trunk
```

- Pour le port de raccordement du routeur vert, on obtient :

```
sudo ovs-vsctl list port tap201 | grep vlan_mode
vlan_mode           : trunk
```

Q5. Comment s'assurer que le port du commutateur est bien configuré à chaque nouveau lancement de machine virtuelle ?

On place les commandes de configuration dans une section dédiée du script de lancement. Voici deux exemples de script de lancement :

```
#!/bin/bash
RAM=1024

echo "Lancement routeur Bleu ou Vert"
SWITCH_PORT=200

sudo ovs-vsctl set port tap$SWITCH_PORT vlan_mode=trunk

$HOME/vm/scripts/ovs-startup.sh router.qcow2 $RAM $SWITCH_PORT
```

Les numéros de port et de VLAN donnés dans les exemples ci-dessus sont à changer suivant le contexte.

5. Routeur Hub (bleu)

Dans cette section, on étudie la machine virtuelle qui joue le rôle de routeur entre le réseau opérateur et le réseau étendu qui dessert le site distant.

5.1. Configuration des interfaces du routeur

Une fois la machine virtuelle routeur lancée, les premières étapes consistent à lui attribuer un nouveau nom et à configurer les interfaces réseau pour joindre les hôtes voisins.

Q6. Comment changer le nom de la machine virtuelle ?

Il faut éditer les deux fichiers `/etc/hosts` et `/etc/hostname` en remplaçant le nom de l'image maître vm0 par le nom voulu. Il est ensuite nécessaire de redémarrer pour que le nouveau nom soit pris en compte par tous les outils du système.

```
etu@vm0:~$ sudo sed -i 's/vm0/bleu/g' /etc/hosts /etc/hostname
etu@vm0:~$ sudo reboot
```

Q7. Comment appliquer les configurations réseau IPv4 et IPv6 à partir de l'unique interface du routeur ?

Consulter les pages de manuels du fichier de configuration système à l'aide de la commande man interfaces.

Il existe plusieurs possibilités pour configurer une interface réseau. Dans le contexte de ces manipulations, on utilise le fichier de configuration fourni par la distribution Debian GNU/Linux : `/etc/network/interfaces`.

La configuration de base fournie avec l'image maître suppose que l'interface obtienne un bail DHCP pour la partie IPv4 et une configuration automatique via SLAAC pour la partie IPv6. Cette configuration par défaut doit être éditée et remplacée. Il faut configurer trois interfaces.

Une interface doit être créée pour chacun des différents réseaux avec le numéro de VLAN désigné dans le plan d'adressage.

- L'interface principale doit être placée en mode manuel (manual). Elle doit être activée/désactivée au niveau de la couche liaison.
- Une interface doit être créée pour le VLAN rouge. Cette interface doit désigner les passerelles IPv4 et IPv6 de façon à joindre l'Internet.
- Une interface doit être créée pour le VLAN violet avec une adresse de lien local IPv6 pour la supervision.
- Une interface doit être créée pour le VLAN orange avec les adresses IPv4 et IPv6 de passerelle pour le réseau étendu.

Voici une copie du fichier `/etc/network/interfaces` de la maquette.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s1
iface enp0s1 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

# ---------- VLAN ROUGE ----------
auto enp0s1.360
iface enp0s1.360 inet static
 address 192.168.104.130
 gateway 192.168.104.129
 dns-nameserver 172.16.0.2

iface enp0s1.360 inet6 static
 address 2001:678:3fc:168::2/64
 gateway fe80:168::1

# ---------- VLAN VIOLET ----------
auto enp0s1.440
iface enp0s1.440 inet6 static
 address fe80:1b8::1/64

# ---------- VLAN ORANGE ----------
auto enp0s1.441
iface enp0s1.441 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down
```

Une fois le fichier de configuration en place, il est préférable de redémarrer la machine virtuelle de façon à vérifier que la configuration des interfaces est bien appliquée après chaque réinitialisation.

Q8. Quels sont les tests de connectivité réalisables après application de la nouvelle configuration des interfaces réseau ?

Relever l'état des trois interfaces et procédez aux tests en respectant les couches de la modélisation.

La commande ip addr ls permet de relever l'état de la configuration pour chaque interface.

```
ip addr ls | grep state
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
2: enp0s1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
3: enp0s1.360@enp0s1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group def
4: enp0s1.440@enp0s1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group def
6: enp0s1.441@enp0s1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group def
```

Sans la confirmation que la configuration du routeur vert est prête, c'est du côté hébergement et accès Internet qu'il faut orienter les tests. Classiquement, on cherche à joindre la passerelle en premier puis l'Internet ensuite via des requêtes ICMP. Enfin, on effectue un test de couche application avec une requête DNS.

```
ping -q -c2 192.168.104.129
PING 192.168.104.129 (192.168.104.129) 56(84) bytes of data.

--- 192.168.104.129 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 1.190/1.712/2.235/0.522 ms
```

```
ping -q -c2 9.9.9.9
PING 9.9.9.9 (9.9.9.9) 56(84) bytes of data.

--- 9.9.9.9 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 20.820/22.682/24.545/1.862 ms
```

```
ping -q -c2 fe80:168::1%enp0s1.360
PING fe80:168::1%enp0s1.360(fe80:168::1%enp0s1.360) 56 data bytes

--- fe80:168::1%enp0s1.360 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.631/1.911/2.192/0.280 ms
```

```
ping -q -c2 2620:fe::fe
PING 2620:fe::fe(2620:fe::fe) 56 data bytes

--- 2620:fe::fe ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 42.231/45.095/47.959/2.864 ms
```

```
host quad9.net
quad9.net has address 216.21.3.77
quad9.net has IPv6 address 2620:0:871:9000::77
quad9.net mail is handled by 5 mx1.quad9.net.
quad9.net mail is handled by 20 mx2.quad9.net.
quad9.net mail is handled by 100 keriomail.pch.net.
```

5.2. Activation de la fonction routage

Sans modification de la configuration par défaut, un système GNU/Linux n'assure pas la fonction de routage du trafic d'une interface réseau à une autre.

L'activation du routage correspond à un réglage de paramètres du sous-système réseau du noyau Linux. L'outil qui permet de consulter et modifier les réglages de paramètre sur le noyau est appelé sysctl. Son fichier de configuration principal est `/etc/sysctl.conf`.

Q9. Comment activer le routage dans le sous-système réseau du noyau Linux ?

Utiliser la commande sysctl pour effectuer des recherches et identifier les paramètres utiles. Par exemple : `sudo sysctl -a -r ".*forward.*"`.

Le fichier `/etc/sysctl.conf` contient des commentaires qui guident facilement vers les bons paramètres.

Attention ! Il ne faut pas oublier d'appliquer les nouvelles valeurs des paramètres de configuration.

Voici un extrait du fichier `/etc/sysctl.conf` du routeur de la maquette après édition.

```
egrep -v '(^#|^$)' /etc/sysctl.conf
net.ipv4.conf.default.rp_filter=1
net.ipv4.conf.all.rp_filter=1
net.ipv4.ip_forward=1
net.ipv6.conf.all.forwarding=1
net.ipv4.conf.all.log_martians = 1
```

Voici une copie d'écran de l'application des nouveaux paramètres.

```
sudo sysctl --system
* Applying /usr/lib/sysctl.d/50-pid-max.conf ...
kernel.pid_max = 4194404
* Applying /etc/sysctl.d/99-sysctl.conf ...
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.ip_forward = 1
net.ipv6.conf.all.forwarding = 1
net.ipv4.conf.all.log_martians = 1
* Applying /usr/lib/sysctl.d/protect-links.conf ...
fs.protected_fifos = 1
fs.protected_hardlinks = 1
fs.protected_regular = 2
fs.protected_symlinks = 1
* Applying /etc/sysctl.conf ...
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.ip_forward = 1
net.ipv6.conf.all.forwarding = 1
net.ipv4.conf.all.log_martians = 1
```

Q10. Quelles sont les conditions à réunir pour tester le fonctionnement du routage ?

Rechercher comment utiliser l'analyseur réseau tshark pour caractériser l'acheminement du trafic d'un réseau à l'autre.

Le plan d'adressage prévoit d'utiliser des préfixes ayant une portée locale pour les réseaux de conteneurs. Il n'est donc pas possible de passer par une requête ICMP pour caractériser l'accès aux réseaux distants. En effet, l'adresse source n'est pas reconnue par l'hôte distant et les routeurs de l'Internet ne disposent d'aucune solution pour joindre le réseau des conteneurs.

Voici un extrait de capture qui montre que le serveur de conteneur cherche à joindre un hôte sur l'Internet sans succès. Cette capture étant réalisée sur l'interface réseau côté hébergement, elle montre que le trafic est bien acheminé d'un réseau à l'autre.

```
tshark -i enp0s1.360
Capturing on 'enp0s1.360'
    1 0.000000000    192.0.2.2 → 9.9.9.9      DNS 81 Standard query 0xbdab A 1.debian.pool.ntp.or
    2 0.000056361    192.0.2.2 → 9.9.9.9      DNS 81 Standard query 0xab92 AAAA 1.debian.pool.ntp
```

5.3. Activation de la traduction d'adresses

Le résultat de la question ci-dessus montre que les hôtes situés dans le réseau des conteneurs ne peuvent pas joindre l'Internet puisque les préfixes réseau utilisés ont une portée limitée.

Q11. Quels sont les paquets qui fournissent les outils de gestion de la traduction d'adresses ?

Rechercher les paquets relatifs au filtrage et à la gestion des règles de pare-feux.

Sur les systèmes GNU/Linux, le système de pare-feux comprend une partie "espace utilisateur" appelée iptables et une partie "noyau" appelée netfilter.

C'est la partie "espace utilisateur" qui nous intéresse ici.

```
apt search ^iptables
Sorting... Done
Full Text Search... Done
iptables/testing 1.8.8-1 amd64
  administration tools for packet filtering and NAT

iptables-netflow-dkms/testing 2.6-3 amd64
  iptables target which generates netflows

iptables-persistent/testing 1.0.16 all
  boot-time loader for netfilter rules, iptables plugin

rtpengine-iptables/testing 10.5.2.4-1+b1 amd64
  IPtables extension module for the kernel-space NGCP media proxy

rtpengine-kernel-dkms/testing 10.5.2.4-1 all
  IPtables kernel module for the NGCP media proxy - DKMS
```

On lance l'installation des deux paquets propres à notre contexte.

```
sudo apt -y install iptables iptables-persistent
```

Q12. Quelles sont les règles à appliquer pour assurer une traduction des adresses sources en sortie sur le réseau hébergement ?

Rechercher dans les pages de manuel de la commande iptables.

C'est la cible `MASQUERADE` qui nous intéresse. Voici un exemple de règles de traduction des adresses sources pour la maquette.

```
sudo iptables -t nat -A POSTROUTING -o enp0s1.360 -j MASQUERADE
 sudo sh -c "iptables-save >/etc/iptables/rules.v4"
```

```
sudo ip6tables -t nat -A POSTROUTING -o enp0s1.360 -j MASQUERADE
 sudo sh -c "ip6tables-save >/etc/iptables/rules.v6"
```

Q13. Comment caractériser le fonctionnement de la traduction d'adresses sources ?

Rechercher dans les pages de manuel de la commande iptables les options d'affichage du décompte du trafic traité.

Voici un exemple d'affichage pour le trafic IPv4 uniquement.

```
sudo iptables -vnL -t nat
Chain PREROUTING (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain POSTROUTING (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    8   598 MASQUERADE  all  --  *      enp0s1.360  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
```

5.4. Activation du protocole PPPoE côté réseau étendu

Pour acheminer le trafic depuis et vers le site distant, il est nécessaire de passer par une authentification. Cette fonction est assurée à l'aide du protocole PPPoE.

Le protocole PPP n'a pas été conçu suivant le modèle Client/Serveur. Il suppose que deux processus pairs échangent des informations. Dans les questions qui suivent, le routeur bleu doit exiger que le routeur vert s'authentifie auprès de lui avant de délivrer les adresses de couche réseau.

Q14. Quel paquet spécifique à la gestion du dialogue PPPoE à installer sur le routeur Hub ?

Rechercher dans le catalogue des paquets, la référence pppoe.

```
apt search ^pppoe
Sorting... Done
Full Text Search... Done
pppoe/testing 3.15-1+b1 amd64
  PPP over Ethernet driver
```

Le résultat de la commande apt show pppoe montre que c'est bien ce paquet qui répond au besoin.

```
sudo apt -y install pppoe
```

Q15. Quel est le rôle de l'outil contenu dans le paquet demandé à la question précédente relativement au démon pppd fourni avec le paquet ppp ?

Rechercher dans les pages de manuels de l'outil demandé à la question précédente.

L'outil pppoe-server gère directement l'encapsulation des trames PPP dans les trames Ethernet. Il communique ensuite les paramètres utiles au démon pppd qui fonctionne de façon totalement transparente vis-à-vis de la technologie du réseau sous-jacent.

Q16. Quels sont les noms des deux sous-couches du protocole PPP qui apparaissent dans les journaux systèmes ?

Quels sont les rôles respectifs de ces deux sous-couches ?

Consulter la page Point-to-Point Protocol.

La consultation des journaux système lors du dialogue PPP fait apparaître des informations du type suivant.

La Section 7, « Trace d'une transaction complète PPPoE » montre en détails les différentes phases de l'établissement de la session PPP.

Q17. Quels sont les en-têtes du dialogue qui identifient les requêtes (émises|reçues), les rejets et les acquittements ?

Consulter les journaux système contenant les traces d'une connexion PPP.

La copie d'écran donnée ci-dessus fait apparaître les directives `Conf*` pour chaque paramètre négocié.

- `ConfReq` indique une requête.
- `ConfAck` indique un acquittement.
- `ConfNak` indique un rejet.

Q18. Dans quel fichier sont stockés les paramètres d'identité et d'authentification utilisés par le protocole CHAP ?

Consulter les pages de manuels du démon pppd à la section AUTHENTICATION.

C'est le fichier `/etc/ppp/chap-secrets` qui contient les couples login/password utilisés lors de l'authentification.

Voici un exemple du contenu de ce fichier.

```
# Secrets for authentication using CHAP
# client server secret   IP addresses
"green"  *  "5p0k3"   *
```

Q19. Dans quel fichier sont stockés les paramètres passés au démon pppd lors du lancement du serveur PPPoE ?

Consulter les pages de manuels de l'outil pppoe-server.

C'est le fichier `/etc/ppp/pppoe-server-options` qui contient la liste des paramètres utilisés lors du dialogue PPP.

Q20. Quelles sont les options du protocole PPP qui doivent être implantées dans le fichier demandé à la question précédente ?

Consulter les pages de manuels du démon pppd et rechercher les paramètres correspondant à la liste suivante.

- Afficher en détail toutes les étapes d'établissement de session dans les journaux système.
- Référencer l'identifiant du compte utilisateur à utiliser lors de l'authentification du routeur vert. Cette option implique que le compte utilisateur existe sur le système et qu'il soit présent dans le fichier `/etc/ppp/chap-secrets`.
- Imposer au routeur vert une authentification via le protocole CHAP (Challenge Handshake Authentication Protocol).
- Préserver la route par défaut, et donc l'accès Internet, du routeur bleu.
- Publier l'adresse IP du serveur DNS à utiliser pour la résolution des noms de domaines.
- Activer l'utilisation des protocoles IPv6CP et IPv6.

Voici une copie de la commande de création du fichier `/etc/ppp/pppoe-server-options` qui contient la liste des paramètres demandés.

```
cat << EOF | sudo tee /etc/ppp/pppoe-server-options
login
require-chap
nodefaultroute
ms-dns 172.16.0.2
+ipv6
EOF
```

Q21. Comment créer le compte utilisateur local sur le routeur bleu sachant qu'il n'est autorisé ni à se connecter ni à avoir un répertoire personnel ?

Consulter les options de la commande adduser.

Voici un exemple de commande adduser.

```
sudo adduser --gecos 'GREEN Router' --disabled-login --no-create-home green
```

Q22. Quels sont les paramètres à donner au lancement de l'outil pppoe-server pour qu'il délivre les adresses au routeur vert après authentification de celui-ci ?

Consulter les options de la commande pppoe-server.

Voici un exemple de commande pppoe-server.

```
sudo pppoe-server -I enp0s1.441 -C BRAS -L 10.4.41.1 -R 10.4.41.2 -N 1
```

Q23. Quels sont les résultats obtenus une fois que la session PPP est établie et que les adresses de couche réseau ont été délivrées ?

Consulter les journaux système, la liste des processus, l'état des interfaces réseau et de la table de routage.

Attention ! Les résultats ne sont pertinents que si le dialogue avec le routeur vert est effectif.

- Consultation des journaux système.

  Voir la Section 7, « Trace d'une transaction complète PPPoE » pour le détail des phases de l'établissement de la session PPP.

- Liste des processus.

```
pppoe-server -I enp0s1.441 -C BRAS -L 10.4.41.1 -R 10.4.41.2 -N 1
 \_ pppd pty /usr/sbin/pppoe -n -I enp0s1.441 -e 1:b0:ad:ca:fe:00:65 -S '' \
    file /etc/ppp/pppoe-server-options 10.4.41.1:10.4.41.2 nodetach noaccomp \
    nopcomp default-asyncmap mru 1492 mtu 1492
  \_ sh -c /usr/sbin/pppoe -n -I enp0s1.441 -e 1:b0:ad:ca:fe:00:65 -S ''
   \_ /usr/sbin/pppoe -n -I enp0s1.441 -e 1:b0:ad:ca:fe:00:65 -S
```

- État des interfaces.

```
$ ip addr ls dev enp0s1.441
6: enp0s1.441@enp0s1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group d
    link/ether b0:ad:ca:fe:00:64 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::b2ad:caff:fefe:64/64 scope link
       valid_lft forever preferred_lft forever

 ip addr ls dev ppp0
26: ppp0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1492 qdisc pfifo_fast state UNKNOWN gro
    link/ppp
    inet 10.4.41.1 peer 10.4.41.2/32 scope global ppp0
       valid_lft forever preferred_lft forever
    inet6 fe80::6c07:edbf:a0b4:f114/10 scope link
       valid_lft forever preferred_lft forever
```

- Table de routage.

```
ip route ls dev ppp0
10.4.41.2 proto kernel scope link src 10.4.41.1
```

Q24. Quelles sont les modifications à apporter au fichier de configuration système des interfaces réseau pour que l'ouverture de session PPP soit disponible après chaque réinitialisation ?

Consulter les pages de manuel du fichier `/etc/network/interfaces` : `man interfaces`.

Voici une copie du fichier dans le contexte de la maquette.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s1
iface enp0s1 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

# ---------- VLAN ROUGE ----------
auto enp0s1.360
iface enp0s1.360 inet static
 address 192.168.104.130/29
 gateway 192.168.104.129
 dns-nameserver 172.16.0.2

iface enp0s1.360 inet6 static
 address 2001:678:3fc:168::2/64
 gateway 2001:678:3fc:168::1

# ---------- VLAN VIOLET ----------
auto enp0s1.440
iface enp0s1.440 inet6 static
 address fe80:1b8::1/64

# ---------- VLAN ORANGE ----------
auto enp0s1.441
iface enp0s1.441 inet manual
 up ip link set dev $IFACE up
 up pppoe-server -I $IFACE -C BRAS -L 10.4.41.1 -R 10.4.41.2 -N 1
 down killall pppoe-server
 down ip link set dev $IFACE down
```

5.5. Ajout des routes statiques vers le réseau des conteneurs

Pour joindre le réseau des conteneurs situé au delà du routeur vert, il est nécessaire d'ajouter une route statique pour chaque protocole de la couche réseau IPv4 et IPv6. Le choix du routage statique est justifié par le fait que l'on adresse un site distant d'extrémité via un lien unique.

Q25. Comment ajouter manuellement les routes IPv4 et IPv6 vers le réseau desservi par le routeur vert ?

Consulter les pages de manuel sur le routage avec la commande : `man ip-route`.

Sachant que le site distant est raccordé via une liaison point à point unique, on choisit de désigner la destination par l'interface de la liaison.

```
sudo ip route add 203.0.113.0/24 dev ppp0
sudo ip -6 route add fda0:7a62:28::/64 dev ppp0
```

Q26. Quels sont les tests de connectivité qui permettent valider la communication à destination des conteneurs du réseau distant ?

Collecter les adresses IPv4 et IPv6 des conteneurs avant de lancer des requêtes ICMP.

Une fois que l'on est assuré que la question Q : Q53 a été traitée, on peut relever les adresses des conteneurs et lancer les tests ICMP.

- Séquence de tests IPv4 :

```
for addr in {10..12}
do
 ping -q -c2 203.0.113.$addr
done
```

- Séquence de tests IPv6 :

```
for addr in {10..12}
do
 ping -q -c2 fda0:7a62:28::$(printf "%x" $addr)
done
```

Q27. Comment appliquer ces routes statiques dans la configuration système pour qu'elles soient activées à chaque établissement de session PPP ?

Il faut parcourir l'arborescence du répertoire `/etc/ppp/` pour repérer les scripts exécutés lors de l'ouverture de session. Créer un script pour chaque protocole de couche réseau qui ajoute la route statique voulue.

- Pour IPv4, le répertoire est `/etc/ppp/ip-up.d/`. Voici comment créer le script exécutable `staticroute`.

```
cat << 'EOF' | sudo tee /etc/ppp/ip-up.d/staticroute
#!/bin/sh

if [ -z "${CONNECT_TIME}" ]; then
    ip route add 203.0.113.0/24 dev ${PPP_IFACE}
fi
EOF
```

```
sudo chmod +x /etc/ppp/ip-up.d/staticroute
```

- Pour IPv6, le répertoire est `/etc/ppp/ipv6-up.d/`. Voici comment créer le script exécutable `staticroute`.

```
cat << 'EOF' | sudo tee /etc/ppp/ipv6-up.d/staticroute
#!/bin/sh

if [ -z "${CONNECT_TIME}" ]; then
    ip -6 route add fda0:7a62:28::/64 dev ${PPP_IFACE}
fi
EOF
```

```
sudo chmod +x /etc/ppp/ipv6-up.d/staticroute
```

6. Routeur Spoke (vert)

6.1. Configuration des interfaces du routeur

Une fois la machine virtuelle serveur de conteneurs lancée, les premières étapes consistent à lui attribuer un nouveau nom et à configurer les interfaces réseau pour joindre le routeur voisin et l'Internet.

Q28. Comment changer le nom de la machine virtuelle ?

Il faut éditer les deux fichiers `/etc/hosts` et `/etc/hostname` en remplaçant le nom de l'image maître vm0 par le nom voulu. Il est ensuite nécessaire de redémarrer pour que le nouveau nom soit pris en compte par tous les outils du système.

```
etu@vm0:~$ sudo sed -i 's/vm0/vert/g' /etc/hosts /etc/hostname
etu@vm0:~$ sudo reboot
```

Q29. Comment appliquer la configuration réseau IPv4 et IPv6 de l'interface du serveur ?

Consulter les pages de manuels du fichier de configuration système à l'aide de la commande man interfaces.

Il existe plusieurs possibilités pour configurer une interface réseau. Dans le contexte de ces manipulations, on utilise le fichier de configuration fourni par la distribution Debian GNU/Linux : `/etc/network/interfaces`.

La configuration de base fournie avec l'image maître suppose que l'interface obtienne un bail DHCP pour la partie IPv4 et une configuration automatique via SLAAC pour la partie IPv6.

La configuration par défaut doit être éditée et remplacée par une configuration manuelle pour l'interface `enp0s1` et pour le VLAN orange. Pour la supervision dans le VLAN violet, l'adresse IPv6 de lien locale est fournie dans le tableau du plan d'adressage.

En attendant que la configuration du routeur bleu soit prête, on ajoute temporairement une interface `enp0s1.60` avec une configuration automatique. Ainsi, il est possible d'installer et de configurer des services en parallèle. Cette interface doit être désactivée dès que tous les outils sont en place.

Voici une copie du fichier `/etc/network/interfaces` de la maquette.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s1
iface enp0s1 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

# ---------- VLAN VIOLET ----------
auto enp0s1.440
iface enp0s1.440 inet6 static
 address fe80:1b8::2/64

# ---------- VLAN ORANGE ----------
auto enp0s1.441
iface enp0s1.441 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

# ---------- TEMPORAIRE -----------
auto enp0s1.60
iface enp0s1.60 inet dhcp
```

6.2. Activation de la fonction routage

Sans modification de la configuration par défaut, un système GNU/Linux n'assure pas la fonction de routage du trafic d'une interface réseau à une autre.

L'activation du routage correspond à un réglage de paramètres du sous-système réseau du noyau Linux. L'outil qui permet de consulter et modifier les réglages de paramètre sur le noyau est appelé sysctl. Son fichier de configuration principal est `/etc/sysctl.conf`.

Q30. Comment activer le routage dans le sous-système réseau du noyau Linux ?

Utiliser la commande sysctl pour effectuer des recherches et identifier les paramètres utiles. Par exemple : `sudo sysctl -a -r ".*forward.*"`.

Le fichier `/etc/sysctl.conf` contient des commentaires qui guident facilement vers les bons paramètres.

Attention ! Il ne faut pas oublier d'appliquer les nouvelles valeurs des paramètres de configuration.

Voici un extrait du fichier `/etc/sysctl.conf` du routeur de la maquette après édition.

```
egrep -v '(^#|^$)' /etc/sysctl.conf
net.ipv4.conf.default.rp_filter=1
net.ipv4.conf.all.rp_filter=1
net.ipv4.ip_forward=1
net.ipv6.conf.all.forwarding=1
net.ipv4.conf.all.log_martians = 1
```

Voici une copie d'écran de l'application des nouveaux paramètres.

```
sudo sysctl --system
* Applying /usr/lib/sysctl.d/50-pid-max.conf ...
kernel.pid_max = 4194404
* Applying /etc/sysctl.d/99-sysctl.conf ...
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.ip_forward = 1
net.ipv6.conf.all.forwarding = 1
net.ipv4.conf.all.log_martians = 1
* Applying /usr/lib/sysctl.d/protect-links.conf ...
fs.protected_fifos = 1
fs.protected_hardlinks = 1
fs.protected_regular = 2
fs.protected_symlinks = 1
* Applying /etc/sysctl.conf ...
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.ip_forward = 1
net.ipv6.conf.all.forwarding = 1
net.ipv4.conf.all.log_martians = 1
```

6.3. Activation du protocole PPP dans le VLAN orange

Le routeur vert utilise un démon pppd sur le VLAN `Data` pour établir une session PPP avec le routeur bleu. À la différence de ce dernier, il n'est pas à l'initiative du dialogue PPPoE mais il doit être capable de gérer l'encapsulation des trames PPP sur un réseau local Ethernet.

Q31. Quel paquet fournit le démon de gestion des sessions du protocole PPP sur le routeur vert ?

Rechercher dans le catalogue des paquets, la référence ppp.

```
apt search ^ppp
Sorting... Done
Full Text Search... Done
ppp/testing 2.4.9-1+1.1+b1 amd64
  Point-to-Point Protocol (PPP) - daemon

ppp-dev/testing 2.4.9-1+1.1 all
  Point-to-Point Protocol (PPP) - development files

ppp-gatekeeper/testing 0.1.0-201406111015-1.1 all
  PPP manager for handling balanced, redundant and failover links

pppconfig/testing 2.3.26 all
  Text menu based utility for configuring ppp

pppoe/testing 3.15-1+b1 amd64
  PPP over Ethernet driver

wmppp.app/testing 1.3.2-2 amd64
  PPP dial control and network load monitor w/ NeXTStep look
```

Le résultat de la commande apt show ppp montre que c'est bien ce paquet qui répond au besoin.

```
sudo apt -y install ppp
```

Q32. Comment utiliser l'encapsulation des trames PPP dans Ethernet à partir du démon pppd fourni avec le paquet ppp ?

Rechercher dans le répertoire de documentation du paquet ppp.

Dans le répertoire `/usr/share/doc/ppp/`, on trouve le fichier `README.pppoe` qui indique que l'appel au module `rp-pppoe.so` permet d'encapsuler des trames PPP sur un réseau local Ethernet.

Toujours à partir du même répertoire, on trouve dans la liste des fichiers d'exemples de configuration un modèle adapté à notre contexte : `peers-pppoe`.

Q33. Dans quel fichier sont stockés les paramètres d'identité et d'authentification utilisés par le protocole CHAP ?

Consulter les pages de manuels du démon pppd à la section AUTHENTICATION.

C'est le fichier `/etc/ppp/chap-secrets` qui contient les couples login/password utilisés lors de l'authentification.

Voici un exemple du contenu de ce fichier. Le nom du client ainsi que son mot de passe secret doivent être identiques à chaque extrémité de la session PPP.

```
# Secrets for authentication using CHAP
# client server secret   IP addresses
"green"  *  "5p0k3"   *
```

Q34. Quelles sont les options de configuration du démon pppd à placer dans le fichier `/etc/ppp/peers/pppoe-provider` pour assurer l'établissement de la session PPP entre les routeurs ?

Utiliser le fichier exemple PPPoE fourni avec la documentation du paquet ppp.

Voici comment créer un fichier `/etc/ppp/peers/pppoe-provider` avec les options correspondant au contexte de la maquette du routeur vert.

```
cat << 'EOF' | sudo tee /etc/ppp/peers/pppoe-provider
# There should be a matching entry with the password in /etc/ppp/chap-secrets.
user "green"

# Load the PPPoE plugin.
plugin rp-pppoe.so

# Ethernet interface to which the modem is connected.
enp0s1.441

# Assumes that your IP address is allocated dynamically by the ISP.
noipdefault
# Try to get the name server addresses from the ISP.
usepeerdns
# Use this connection as the default route.
defaultroute

# Makes pppd "dial again" when the connection is lost.
persist

# Do not ask the remote to authenticate.
noauth

debug
+ipv6
EOF
```

Q35. Comment lancer le démon pppd pour qu'il prenne en compte les paramètres définis dans le fichier complété à la question précédente ?

Consulter les pages de manuels du démon pppd.

C'est l'option `file` qui permet de désigner le fichier de configuration à utiliser. Voici une copie d'écran du lancement de pppd.

```
sudo pppd file /etc/ppp/peers/pppoe-provider
```

Q36. Quels sont les noms des deux sous-couches du protocole PPP qui apparaissent dans les journaux systèmes ? Quels sont les rôles respectifs de ces deux sous-couches ?

Consulter la page Point-to-Point Protocol.

La consultation des journaux système lors du dialogue PPP fait apparaître tous les détails. Voir la Section 7, « Trace d'une transaction complète PPPoE ».

Q37. Quels sont les en-têtes du dialogue qui identifient les requêtes (émises|reçues), les rejets et les acquittements ?

Consulter les journaux système contenant les traces d'une connexion PPP.

La copie d'écran donnée ci-dessus fait apparaître les directives `Conf*` pour chaque paramètre négocié.

- `ConfReq` indique une requête.
- `ConfAck` indique un acquittement.
- `ConfNak` indique un rejet.

Q38. Quelles sont les modifications à apporter au fichier système de configuration des interfaces réseau pour ouvrir la session PPP à chaque réinitialisation système ?

Consulter les pages de manuel du fichier `/etc/network/interfaces` : `man interfaces`.

Voici une copie du fichier modifié dans le contexte de la maquette.

Attention ! L'interface temporaire doit être impérativement désactivée au moment de l'activation de la session PPP.

Voir la Section 7, « Trace d'une transaction complète PPPoE » pour le détail des phases de l'établissement de la session PPP.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s1
iface enp0s1 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

# ---------- VLAN VIOLET ----------
auto enp0s1.440
iface enp0s1.440 inet6 static
 address fe80:1b8::2/64

# ---------- VLAN ORANGE ----------
auto enp0s1.441
iface enp0s1.441 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

# ---------- PPPoE ---------------
auto pppoe-provider
iface pppoe-provider inet ppp
        pre-up ifup enp0s1.441
        provider pppoe-provider

# ---------- TEMPORAIRE -----------
#auto enp0s1.60
#iface enp0s1.60 inet dhcp
```

6.4. Activation du commutateur virtuel asw-host

Dans le scénario étudié, les services sont hébergés dans un réseau de conteneurs propre au routeur vert. La mise en œuvre de cette configuration passe par l'installation d'un commutateur virtuel appelé `asw-host`. On utilise Open vSwitch pour configurer ce commutateur.

Q39. Quel est le paquet à installer pour pouvoir ajouter un commutateur virtuel au routeur vert ?

Rechercher le mot clé openvswitch dans la liste des paquets.

Voici un exemple de recherche.

```
apt search ^openvswitch
Sorting... Done
Full Text Search... Done
openvswitch-common/testing 2.17.2-5+b1 amd64
  Open vSwitch common components

openvswitch-doc/testing 2.17.2-5 all
  Open vSwitch documentation

openvswitch-ipsec/testing 2.17.2-5+b1 amd64
  Open vSwitch IPsec tunneling support

openvswitch-pki/testing 2.17.2-5 all
  Open vSwitch public key infrastructure dependency package

openvswitch-source/testing 2.17.2-5 all
  Open vSwitch source code

openvswitch-switch/testing 2.17.2-5+b1 amd64
  Open vSwitch switch implementations

openvswitch-switch-dpdk/testing 2.17.2-5+b1 amd64
  DPDK enabled Open vSwitch switch implementation

openvswitch-test/testing 2.17.2-5 all
  Open vSwitch test package

openvswitch-testcontroller/testing 2.17.2-5+b1 amd64
  Simple controller for testing OpenFlow setups

openvswitch-vtep/testing 2.17.2-5+b1 amd64
  Open vSwitch VTEP utilities
```

C'est le paquet `openvswitch-switch` qui nous intéresse.

```
sudo apt -y install openvswitch-switch
```

Q40. Quel est le fichier de documentation qui fournit les directives de configuration d'un commutateur intégré au fichier système `/etc/network/interfaces` ?

Rechercher dans la liste des fichiers des paquets installés à la question précédente.

Voici un exemple de recherche.

```
dpkg -L openvswitch-switch | grep README
/usr/share/doc/openvswitch-switch/README.Debian.gz
```

Q41. Quelles sont les modiifications à apporter au fichier `/etc/network/interfaces` pour configurer le commutateur `asw-host` ?

Voici une copie du fichier de configuration réseau système dans le contexte de la maquette.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s1
iface enp0s1 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

# ---------- VLAN VIOLET ----------
auto enp0s1.440
iface enp0s1.440 inet6 static
 address fe80:1b8::2/64

# ---------- VLAN ORANGE ----------
auto enp0s1.441
iface enp0s1.441 inet manual
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

# ---------- PPPoE ----------------
auto pppoe-provider
iface pppoe-provider inet ppp
        pre-up ifup enp0s1.441
        provider pppoe-provider

# ---------- VLAN VERT ------------
auto asw-host
iface asw-host inet manual
 ovs_type OVSBridge
 ovs_ports sw-vlan40
 up ip link set dev $IFACE up
 down ip link set dev $IFACE down

allow-asw-host sw-vlan40
iface sw-vlan40 inet static
 ovs_type OVSBridge
 ovs_bridge asw-host
 ovs_options asw-host 40
 address 203.0.113.1/24

iface sw-vlan40 inet6 static
 ovs_type OVSBridge
 ovs_bridge asw-host
 ovs_options asw-host 40
 address fda0:7a62:28::1/64
```

6.5. Activation de la configuration IPv6 automatique pour le réseau de conteneurs

Pour que les hôtes du réseau de conteneurs obtiennent automatiquement une configuration IPv6, il faut que le routeur assure les annonces auprès de ces voisins. Un moyen simple pour assurer la configuration SLAAC des hôtes voisins du routeur consiste à utiliser le paquet radvd.

On débute par l'installation de ce paquet.

```
sudo apt install radvd
```

On voit que le lancement du service a échoué.

```
systemctl status radvd
# radvd.service - Router advertisement daemon for IPv6
     Loaded: loaded (/lib/systemd/system/radvd.service; enabled; preset: enabled)
     Active: inactive (dead)
  Condition: start condition failed at Thu 2022-10-20 10:38:12 CEST; 38s ago
             └─ ConditionPathExists=/etc/radvd.conf was not met
       Docs: man:radvd(8)
```

Q42. Comment configurer le service radvd pour publier les annonces côté conteneurs ?

Rechercher les options utiles dans les pages de manuel du service : `man radvd.conf`.

Voici comment créer le fichier de configuration `/etc/radvd.conf` de la maquette.

```
cat << 'EOF' | sudo tee /etc/radvd.conf
interface sw-vlan40
{
        AdvSendAdvert on;

        prefix fda0:7a62:28::/64
        {
                AdvOnLink on;
                AdvAutonomous on;
                AdvRouterAddr on;
        };

        RDNSS 2620:fe::fe
        {
        };
};
EOF
```

Attention ! Une fois le fichier créé, il ne faut pas oublier de redémarrer le service et de contrôler l'état de son fonctionnement.

```
sudo systemctl restart radvd
```

```
systemctl status radvd
• radvd.service - Router advertisement daemon for IPv6
     Loaded: loaded (/lib/systemd/system/radvd.service; enabled; preset: enabled)
     Active: active (running) since Thu 2022-10-20 10:42:10 CEST; 13s ago
       Docs: man:radvd(8)
    Process: 1693 ExecStartPre=/usr/sbin/radvd --logmethod stderr_clean --configtest (code=exited
    Process: 1694 ExecStart=/usr/sbin/radvd --logmethod stderr_clean (code=exited, status=0/SUCCE
   Main PID: 1695 (radvd)
      Tasks: 2 (limit: 1114)
     Memory: 476.0K
        CPU: 30ms
     CGroup: /system.slice/radvd.service
             ├─1695 /usr/sbin/radvd --logmethod stderr_clean
             └─1696 /usr/sbin/radvd --logmethod stderr_clean
```

6.6. Validation des routes par défaut vers le réseau opérateur

Pour joindre l'Internet situé au delà du routeur bleu, il est nécessaire de vérifier la route par défaut pour chaque protocole de la couche réseau : IPv4 et IPv6.

Attention ! Les tests de connectivité vers l'Internet supposent que le routeur bleu soit fonctionnel.

Q43. Comment tester la présence des routes par défaut IPv4 et IPv6 vers le routeur bleu ?

Consulter les pages de manuel sur le routage avec la commande : `man ip-route`.

Sachant que le site distant est raccordé via une liaison point à point unique, on doit voir apparaître le nom de l'interface de la session PPP.

- Table de routage IPv4 : la route par défaut est bien présente.

```
ip route ls
default dev ppp0 scope link
10.4.41.1 dev ppp0 proto kernel scope link src 10.4.41.2
203.0.113.0/24 dev sw-vlan40 proto kernel scope link src 203.0.113.1
```

- Table de routage IPv6 : la route par défaut est absente.

```
ip -6 route ls
::1 dev lo proto kernel metric 256 pref medium
fda0:7a62:28::/64 dev sw-vlan40 proto kernel metric 256 pref medium
fe80::5d72:1dd3:781a:ff02 dev ppp0 proto kernel metric 256 pref medium
fe80::dd00:24bc:4b7d:f4df dev ppp0 proto kernel metric 256 pref medium
fe80::/64 dev enp0s1 proto kernel metric 256 pref medium
fe80::/64 dev enp0s1.440 proto kernel metric 256 pref medium
fe80::/64 dev enp0s1.441 proto kernel metric 256 pref medium
fe80::/64 dev sw-vlan40 proto kernel metric 256 pref medium
fe80::/64 dev asw-host proto kernel metric 256 pref medium
fe80:1b8::/64 dev enp0s1.440 proto kernel metric 256 pref medium
```

Q44. Comment ajouter des routes statiques dans la configuration système pour qu'elles soient activées à chaque établissement de session PPP ?

Il faut parcourir l'arborescence du répertoire `/etc/ppp/` pour repérer les scripts exécutés lors de l'ouverture de session. Créer un script pour chaque protocole de couche réseau qui ajoute la route statique voulue.

- Pour IPv4, le répertoire est `/etc/ppp/ip-up.d/`. Voici comment créer un script exécutable appelé `staticroute`.

```
cat << 'EOF' | sudo tee /etc/ppp/ip-up.d/staticroute
#!/bin/sh

if [ -z "${CONNECT_TIME}" ]; then
 ip route add default dev ${PPP_IFACE}
fi
EOF
```

```
sudo chmod +x /etc/ppp/ip-up.d/staticroute
```

- Pour IPv6, le répertoire est `/etc/ppp/ipv6-up.d/`. Voici comment créer un script exécutable aussi appelé`staticroute`.

```
cat << 'EOF' | sudo tee /etc/ppp/ipv6-up.d/staticroute
#!/bin/sh

  if [ -z "${CONNECT_TIME}" ]; then
      ip -6 route add default dev ${PPP_IFACE}
  fi
EOF
```

```
sudo chmod +x /etc/ppp/ipv6-up.d/staticroute
```

Q45. Quels sont les tests de connectivité qui permettent valider la communication vers l'Internet en passant par le routeur bleu ?

Au niveau de la couche réseau, on lance les requêtes ICMP classques.

Les tests suivants doivent nécessairement se faire depuis le routeur ***Spoke*** après avoir réinitialisé la session PPP de façon à bien utiliser les routes par défaut imposées.

```
sudo ifdown pppoe-provider
sudo ifup pppoe-provider
```

Voici un exemple de test pour IPv4.

```
ip route get 9.9.9.9
9.9.9.9 dev ppp0 src 10.4.41.2 uid 1000
    cache
```

```
ping -q -c2 9.9.9.9
PING 9.9.9.9 (9.9.9.9) 56(84) bytes of data.

--- 9.9.9.9 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 112.766/127.443/142.120/14.677 ms
```

On change l'adresse de destination IPv6.

```
ip -6 route get 2620:fe::fe
2620:fe::fe from :: dev ppp0 src fda0:7a62:28::1 metric 1024 pref medium
```

```
ping -q -c2 2620:fe::fe
PING 2620:fe::fe(2620:fe::fe) 56 data bytes

--- 2620:fe::fe ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 44.403/44.627/44.852/0.224 ms
```

6.7. Installation du gestionnaire de conteneurs LXD

Sur le routeur vert, la gestion des conteneurs est confiée à LXD. Pour des raisons de rapidité de mise en œuvre, on choisit de passer par le gestionnaire de paquets snapd pour l'installation des outils.

Q46. Comment installer le gestionnaire de paquets snap sur une distribution Debian GNU/Linux ?

Effectuer une recherche dans les paquets fournis via APT.

Il existe tout simplement un paquet appelé snapd.

```
sudo apt -y install snapd
```

Q47. Comment installer le gestionnaire de conteneurs LXD ?

Rechercher dans la liste des snaps.

Le snap s'appelle tout simplement lxd.

```
sudo snap install lxd
2022-10-20T11:14:38+02:00 INFO Waiting for automatic snapd restart...
lxd 5.6-794016a from Canonical√ installed
```

On peut lister les snaps installés.

```
snap list
Name    Version      Rev    Tracking       Publisher   Notes
core20  20220826     1623   latest/stable  canonical√  base
lxd     5.6-794016a  23680  latest/stable  canonical√  -
snapd   2.57.4       17336  latest/stable  canonical√  snapd
```

Q48. Comment faire pour que l'utilisateur normal `etu` ait la capacité à gérer les conteneurs ?

Rechercher le nom du groupe système correspondant à l'utilisation des outils LXD.

Il faut que l'utilisteur normal appartienne au groupe système `lxd` pour qu'il est tous les droits sur la gestion des conteneurs.

```
sudo adduser etu lxd
```

Attention ! Il faut se déconnecter/reconnecter pour bénéficier de la nouvelle attribution de groupe. On peut utiliser la commande groups pour vérifier le résultats.

```
groups
etu adm cdrom floppy sudo audio dip video plugdev staff netdev lxd
```

6.8. Configuration du gestionnaire de conteneurs LXD

Q49. Quelle est l'instruction de configuration initiale du gestionnaire LXD ?

Utiliser l'aide de la commande lxd.

C'est l'instruction `lxd init` qui nous intéresse.

Voici une copie d'écran de son exécution.

```
lxd init
Would you like to use LXD clustering? (yes/no) [default=no]:
Do you want to configure a new storage pool? (yes/no) [default=yes]:
Name of the new storage pool [default=default]:
Name of the storage backend to use (lvm, btrfs, ceph, cephobject, dir) [default=btrfs]:
Create a new BTRFS pool? (yes/no) [default=yes]:
Would you like to use an existing empty block device (e.g. a disk or partition)? (yes/no) [defaul
Size in GiB of the new loop device (1GiB minimum) [default=23GiB]:
Would you like to connect to a MAAS server? (yes/no) [default=no]:
Would you like to create a new local network bridge? (yes/no) [default=yes]: no
Would you like to configure LXD to use an existing bridge or host interface? (yes/no) [default=no
Name of the existing bridge or host interface: sw-vlan40
Would you like the LXD server to be available over the network? (yes/no) [default=no]:
Would you like stale cached images to be updated automatically? (yes/no) [default=yes]:
Would you like a YAML "lxd init" preseed to be printed? (yes/no) [default=no]:
```

Q50. Comment changer le type de raccordement défini par le paramètre `nictype` de `macvlan` à `bridged` ?

Rechercher dans les options d'édition des paramètres du profil avec la commande lxc.

Il faut suivre les champs du fichier yaml de description du profil.

```
lxc profile device set default eth0 nictype bridged
lxc profile device get default eth0 nictype
bridged
```

Q51. Quelle est l'instruction qui permet d'afficher le profil par défaut des conteneur ?

Rechercher dans les options de la commande lxc.

Voici un exemple d'exécution.

```
lxc profile show default
config: {}
description: Default LXD profile
devices:
  eth0:
    name: eth0
    nictype: bridged
    parent: sw-vlan40
    type: nic
  root:
    path: /
    pool: default
    type: disk
name: default
used_by: []
```

Q52. Quelle est l'instruction de lancement d'un conteneur ?

Rechercher dans les options de la commande lxc.

Tester son exécution avec un conteneur de type `debian/bullseye`.

Voici un exemple de création de 3 conteneurs.

```
for i in {0..2}
do
 lxc launch images:debian/12 c$i
done
```

```
lxc ls
+------+---------+------+-----------------------------------------+-----------+-----------+
| NAME |  STATE  | IPV4 |                   IPV6                  |   TYPE    | SNAPSHOTS |
+------+---------+------+-----------------------------------------+-----------+-----------+
| c0   | RUNNING |      | fda0:7a62:28:0:216:3eff:fe0d:3814 (eth0) | CONTAINER | 0         |
+------+---------+------+-----------------------------------------+-----------+-----------+
| c1   | RUNNING |      | fda0:7a62:28:0:216:3eff:fe8e:dfbb (eth0) | CONTAINER | 0         |
+------+---------+------+-----------------------------------------+-----------+-----------+
| c2   | RUNNING |      | fda0:7a62:28:0:216:3eff:fe48:e06f (eth0) | CONTAINER | 0         |
+------+---------+------+-----------------------------------------+-----------+-----------+
```

Q53. Comment appliquer une configuration IPv4 statique à chaque conteneur ?

Identifier le fichier de configuration système et modifier ce fichier pour chaque conteneur

Comme les conteneurs utilisent `systemd-networkd` comme gestionnaire de la configuration réseau, le fichier à identifier est : `/etc/systemd/network/eth0.network`.

```
for i in {0..2}
do
        echo ">>>>>>>>>>>>>>>>> c$i"
config=$(cat << EOF
[Match]
Name=eth0

[Network]
DHCP=false
Address=203.0.113.$((i + 10))/24
Address=fda0:7a62:28::$(printf "%x" $((i + 10)))/64
Gateway=203.0.113.1
DNS=172.16.0.2
EOF
)
        lxc exec c$i -- bash -c "echo \"${config}\" | tee /etc/systemd/network/eth0.network"
        lxc exec c$i -- systemctl restart systemd-networkd
done
```

On relève le résultat avec la commande `lxc ls`.

```
lxc ls
+------+---------+--------------------+-----------------------------------------+-----------+--
| NAME |  STATE  |        IPV4        |                   IPV6                  |   TYPE    | S
+------+---------+--------------------+-----------------------------------------+-----------+--
| c0   | RUNNING | 203.0.113.10 (eth0) | fda0:7a62:28::a (eth0)                  | CONTAINER | 0
|      |         |                    | fda0:7a62:28:0:216:3eff:fe0d:3814 (eth0) |           |
+------+---------+--------------------+-----------------------------------------+-----------+--
| c1   | RUNNING | 203.0.113.11 (eth0) | fda0:7a62:28::b (eth0)                  | CONTAINER | 0
|      |         |                    | fda0:7a62:28:0:216:3eff:fe8e:dfbb (eth0) |           |
+------+---------+--------------------+-----------------------------------------+-----------+--
| c2   | RUNNING | 203.0.113.12 (eth0) | fda0:7a62:28::c (eth0)                  | CONTAINER | 0
|      |         |                    | fda0:7a62:28:0:216:3eff:fe48:e06f (eth0) |           |
+------+---------+--------------------+-----------------------------------------+-----------+--
```

Q54. Comment vérifier la connectivité réseau depuis les conteneurs ?

La question précédente montre que la configuration réseau des conteneurs est complète. On doit donc lancer des tests IPv4 et IPv6.

Voici deux exemples de tests ICMP.

```
for i in {0..2}
do
 echo ">>>>>>>>>>>>>>>>> c$i"
 lxc exec c$i -- ping -c2 2620:fe::fe
done
```

```
for i in {0..2}
do
 echo ">>>>>>>>>>>>>>>>> c$i"
 lxc exec c$i -- ping -c2 9.9.9.9
done
```

On peut ensuite passer à la gestion de paquets pour valider les transactions de la couche application.

```
for i in {0..2}
do
 echo ">>>>>>>>>>>>>>>>> c$i"
 lxc exec c$i -- apt -y update
 lxc exec c$i -- apt -y full-upgrade
 lxc exec c$i -- apt clean
done
```

7. Trace d'une transaction complète PPPoE

Pour obtenir la trace des transactions entre les deux routeurs de la maquette, on peut utiliser l'une des deux commandes suivantes :

```
grep -i ppp /var/log/syslog
```

```
journalctl /usr/sbin/pppd
```

7.1. Routeur Spoke (vert)

Côté routeur ***Spoke***, on obtient des résultats très détaillés.

```
vert pppd[672]: pppd 2.4.9 started by root, uid 0
vert kernel: [    6.610615] NET: Registered PF_PPPOX protocol family
vert pppd[672]: Send PPPOE Discovery V1T1 PADI session 0x0 length 12 ❶
vert pppd[672]:  dst ff:ff:ff:ff:ff:ff  src b8:ad:ca:fe:00:c9
vert pppd[672]:  [service-name] [host-uniq  a0 02 00 00]
vert pppd[672]: Recv PPPOE Discovery V1T1 PADO session 0x0 length 44
vert pppd[672]:  dst b8:ad:ca:fe:00:c9  src b8:ad:ca:fe:00:c8
vert pppd[672]:  [AC-name BRAS] [service-name] [AC-cookie  3c 46 65 e0 22 81 9c b1 a9 1b fd 9a 56 fe 6f
vert pppd[672]: Send PPPOE Discovery V1T1 PADR session 0x0 length 36
vert pppd[672]:  dst b8:ad:ca:fe:00:c8  src b8:ad:ca:fe:00:c9
vert pppd[672]:  [service-name] [host-uniq  a0 02 00 00] [AC-cookie  3c 46 65 e0 22 81 9c b1 a9 1b fd 9
vert pppd[672]: Recv PPPOE Discovery V1T1 PADS session 0x1 length 12
vert pppd[672]:  dst b8:ad:ca:fe:00:c9  src b8:ad:ca:fe:00:c8
vert pppd[672]:  [service-name] [host-uniq  a0 02 00 00]
vert pppd[672]: PADS: Service-Name: ''
vert pppd[672]: PPP session is 1
vert pppd[672]: Connected to b8:ad:ca:fe:00:c8 via interface enp0s1.441
vert pppd[672]: using channel 1
vert pppd[672]: Using interface ppp0
vert pppd[672]: Connect: ppp0 <--> enp0s1.441
vert pppd[672]: sent [LCP ConfReq id=0x1 <mru 1492> <magic 0x2d6c5498>] ❷
vert pppd[672]: rcvd [LCP ConfReq id=0x1 <mru 1492> <auth chap MD5> <magic 0xac34879f>]
vert pppd[672]: sent [LCP ConfAck id=0x1 <mru 1492> <auth chap MD5> <magic 0xac34879f>]
vert pppd[672]: sent [LCP ConfReq id=0x1 <mru 1492> <magic 0x2d6c5498>]
vert pppd[672]: rcvd [LCP ConfAck id=0x1 <mru 1492> <magic 0x2d6c5498>]
vert pppd[672]: sent [LCP EchoReq id=0x0 magic=0x2d6c5498]
vert pppd[672]: rcvd [LCP EchoReq id=0x0 magic=0xac34879f]
vert pppd[672]: sent [LCP EchoRep id=0x0 magic=0x2d6c5498]
vert pppd[672]: rcvd [CHAP Challenge id=0xb2 <23999618eb56316376ef1f75f76e89e33da0697a7df4>, name = "bl
vert pppd[672]: sent [CHAP Response id=0xb2 <1551c472fd270cc4853896725f1b1757>, name = "green"]
vert pppd[672]: rcvd [LCP EchoRep id=0x0 magic=0xac34879f]
vert pppd[672]: rcvd [CHAP Success id=0xb2 "Access granted"]
vert pppd[672]: CHAP authentication succeeded: Access granted
vert pppd[672]: CHAP authentication succeeded
vert pppd[672]: peer from calling number B8:AD:CA:FE:00:C8 authorized
vert pppd[672]: sent [IPCP ConfReq id=0x1 <addr 0.0.0.0> <ms-dns1 0.0.0.0> <ms-dns2 0.0.0.0>] ❸
vert pppd[672]: sent [IPV6CP ConfReq id=0x1 <addr fe80::dd00:24bc:4b7d:f4df>]
vert pppd[672]: rcvd [CCP ConfReq id=0x1 <deflate 15> <deflate(old#) 15> <bsd v1 15>]
vert pppd[672]: sent [CCP ConfReq id=0x1]
vert pppd[672]: sent [CCP ConfRej id=0x1 <deflate 15> <deflate(old#) 15> <bsd v1 15>]
vert pppd[672]: rcvd [IPCP ConfReq id=0x1 <compress VJ 0f 01> <addr 10.4.41.1>]
vert pppd[672]: sent [IPCP ConfRej id=0x1 <compress VJ 0f 01>]
vert pppd[672]: rcvd [IPV6CP ConfReq id=0x1 <addr fe80::5d72:1dd3:781a:ff02>]
vert pppd[672]: sent [IPV6CP ConfAck id=0x1 <addr fe80::5d72:1dd3:781a:ff02>]
vert pppd[672]: rcvd [IPCP ConfNak id=0x1 <addr 10.4.41.2> <ms-dns1 172.16.0.2> <ms-dns2 172.16.0.2>]
vert pppd[672]: sent [IPCP ConfReq id=0x2 <addr 10.4.41.2> <ms-dns1 172.16.0.2> <ms-dns2 172.16.0.2>]
vert pppd[672]: rcvd [IPV6CP ConfAck id=0x1 <addr fe80::dd00:24bc:4b7d:f4df>]
vert pppd[672]: local  LL address fe80::dd00:24bc:4b7d:f4df
vert pppd[672]: remote LL address fe80::5d72:1dd3:781a:ff02
vert pppd[672]: Script /etc/ppp/ipv6-up started (pid 700)
vert pppd[672]: rcvd [CCP ConfAck id=0x1]
vert pppd[672]: rcvd [CCP ConfReq id=0x2]
vert pppd[672]: sent [CCP ConfAck id=0x2]
vert pppd[672]: rcvd [IPCP ConfReq id=0x2 <addr 10.4.41.1>]
vert pppd[672]: sent [IPCP ConfAck id=0x2 <addr 10.4.41.1>]
vert pppd[672]: rcvd [IPCP ConfAck id=0x2 <addr 10.4.41.2> <ms-dns1 172.16.0.2> <ms-dns2 172.16.0.2>]
vert pppd[672]: Script /etc/ppp/ip-pre-up started (pid 701)
vert pppd[672]: Script /etc/ppp/ip-pre-up finished (pid 701), status = 0x0
vert pppd[672]: local  IP address 10.4.41.2
vert pppd[672]: remote IP address 10.4.41.1
vert pppd[672]: primary   DNS address 172.16.0.2
vert pppd[672]: secondary DNS address 172.16.0.2
vert pppd[672]: Script /etc/ppp/ip-up started (pid 705)
vert pppd[672]: Script /etc/ppp/ipv6-up finished (pid 700), status = 0x0
vert pppd[672]: Script /etc/ppp/ip-up finished (pid 705), status = 0x0
```

❶ Sur un réseau de diffusion il est nécessaire d'identifier les deux hôtes qui doivent établir une session PPP. Cette toute première phase d'identification utilise des trames spécifiques avec les messages décrits dans la Section 2, « Interface Ethernet & protocole PPP ».

❷ La sous-couche Link Control Protocol (LCP) assure la configuration automatique des interfaces à chaque extrémité. Les paramètres négociés entre les deux hôtes en communication sont

multiples : l'adaptation de la taille de datagramme, les caractères d'échappement, les numéros magiques et la sélection des options d'authentification.

❸ La sous-couche Network Control Protocol (NCP) assure l'encapsulation de multiples protocoles de la couche réseau. Dans l'exemple donné, c'est le protocole IPv4 qui est utilisé ; d'où l'acronyme IPCP.

7.2. Routeur Hub (bleu)

Côté routeur ***Spoke***, on obtient des résultats très simplifiés.

```
bleu pppoe-server[3388]: Session 1 created for client b8:ad:ca:fe:00:c9 (10.4.41.2) on enp0s1.441 using
bleu pppd[3388]: pppd 2.4.9 started by etu, uid 0
bleu pppd[3388]: Using interface ppp0
bleu pppd[3388]: Connect: ppp0 <--> /dev/pts/0
bleu pppd[3388]: pam_unix(ppp:session): session opened for user green(uid=1001) by (uid=0)
bleu pppd[3388]: user green logged in on tty  intf ppp0
bleu pppd[3388]: local  LL address fe80::5d72:1dd3:781a:ff02
bleu pppd[3388]: remote LL address fe80::dd00:24bc:4b7d:f4df
bleu pppd[3388]: local  IP address 10.4.41.1
bleu pppd[3388]: remote IP address 10.4.41.2
```