

Master 2 RSH
TP3 ADMI
Routage Dynamique et RIP

Encadré par:
Ahmad FADEL

Réalisé par:
Fanny PRIEUR / Lorenzo MAZZOCCHI

Année 2018/2019

- Rappel Quagga/Zebra :

GNU Zebra est une suite de logiciels de routage. Il prend en charge plusieurs protocoles et permet de transformer une machine Unix en routeur.

Quagga est une suite de logiciels de routage implémentant les protocoles **OSPF** (v2 & v3), **RIP** (v1, v2 & v3), **BGP** (v4) et **IS-IS** pour les plates-formes de type Unix. Tout comme GNU Zebra, il permet de transformer une machine Unix en routeur.

Quagga est un fork du projet **GNU Zebra** (inactif depuis 2005).

Partie 1 : Configuration préliminaire

- Les commandes permettant de désactiver les différentes interfaces sont les suivantes :

```
ifconfig eth0 down  
ifconfig eth1 down  
ifconfig eth2 down  
ifconfig eth3 down  
ifconfig eth4 down
```

- Afin de répondre à la topologie demandée, nous avons effectué les commandes suivantes sur nos interfaces situées sur la machine Linux :

```
ifconfig eth1 8.0.6.1 netmask 255.255.255.0  
ifconfig eth2 9.0.6.2 netmask 255.255.255.0  
ifconfig eth3 10.0.5.2 netmask 255.255.255.0
```

- Afin de répondre à la topologie demandée, nous avons effectué la commande suivante sur l'interface située sur la machine Windows :

L'adresse IP et le masque : **8.0.6.2/24**
Passerelle par défaut : **8.0.6.1/24**

- Nous avons effectué un **ping** entre notre machine Windows 8.0.6.2/24 et la machine Windows voisine 8.0.5.1/24. Nous constatons que le **ping** fonctionne.

Partie 2 : Protocole RIP

Rappels:

- ❑ Pour lancer Quagga : **/etc/init.d/quagga start**
- ❑ Mot de passe des démons **zebra** et **ripd** est "**zebra**".
- ❑ Edition du fichier **/etc/quagga/daemons** en modifiant **zebra=yes** et **ripd=yes**.
- ❑ En lançant Quagga, nous n'avons eu aucun message d'erreur et nous n'avons donc pas fait la liste de commande indiqué dans le tp.

2.1-Mise en place

Question 1 :

La commande permettant d'obtenir le port de zebra est « **netstat -antp | grep zebra** ».

Le port zebra est : **2601**

La commande permettant d'obtenir le port de ripd est « **netstat -antp | grep ripd** ».

Le port ripd est : **2602**

```
root@localhost:/etc/quagga# netstat -antp | grep zebra
tcp        0      0 127.0.0.1:2601      0.0.0.0:*           LISTEN      1720/zebra
root@localhost:/etc/quagga# netstat -antp | grep ripd
tcp        0      0 127.0.0.1:2602      0.0.0.0:*           LISTEN      1724/ripd
```

Question 2 :

Pour se connecter au démon zebra, nous utilisons la commande « **telnet localhost 2601** ».

mdp : zebra

Pour se connecter au démon ripd, nous utilisons la commande « **telnet localhost 2602** ».

mdp : zebra

Question 3 :

Il faut se placer en mode « **configure terminal** ».

Puis entrer les commandes :

« **interface ethX** », X étant l'interface que nous voulons configurer.

« **ip address <ip>/24** », <ip> étant l'adresse IP à utiliser.

« **write file** », qui nous permet de sauvegarder la configuration.

Nous obtenons le fichier de configuration pour zebra :

```
GNU nano 2.2.6 Fichier : zebra.conf
!
! Zebra configuration saved from vty
!   2018/10/17 09:22:44
!
hostname Router
password zebra
enable password zebra
!
interface eth0
  ipv6 nd suppress-ra
!
interface eth1
  ip address 8.0.6.1/24
  ipv6 nd suppress-ra
!
interface eth2
  ip address 9.0.6.2/24
  ipv6 nd suppress-ra
!
interface eth3
  ip address 10.0.5.2/24
  ipv6 nd suppress-ra
!
interface eth4
  ipv6 nd suppress-ra
!
interface lo
!
interface wlan0
  ipv6 nd suppress-ra
!
!
!
line vty
!
```

La commande **echo "1" > /proc/sys/net/ipv4/ip_forward** écrit le chiffre 1 dans le fichier ip_forward. Cette commande permet d'activer l'IP forwarding en mettant la valeur 1. Sinon, l'IP Forwarding est désactivé par défaut avec la valeur 0.

Cette commande sert sous linux à activer les fonctions de routage entre les interfaces réseau du système. Cela n'a de sens que dans le cas d'une machine possédant plusieurs interfaces réseau. La commande autorise le système à rediriger un paquet de données

arrivé par une interface réseau vers une autre interface réseau, conformément à la table de routage du système.

2.2-RIP

Question 1 :

« **configure terminal** » : Passer en mode SuperUtilisateur sur le routeur.

« **router rip** » : Cette commande active le mode RIP (Routing Information Protocol) qui est un protocole de routage. Il permet à chaque routeur de communiquer aux autres routeurs la distance qui les sépare du réseau.

« **version 2** » : Nous définissons RIP en version 2 qui utilise le multicast.

« **network 8.0.6.0/24** » : Adresse réseau que RIP doit gérer.

« **network 9.0.6.0/24** » : Adresse réseau que RIP doit gérer.

« **network 10.0.6.0/24** » : Adresse réseau que RIP doit gérer.

« **redistribute connected** » : Commande permettant de définir RIP pour qu'il propager ces routes statiques.

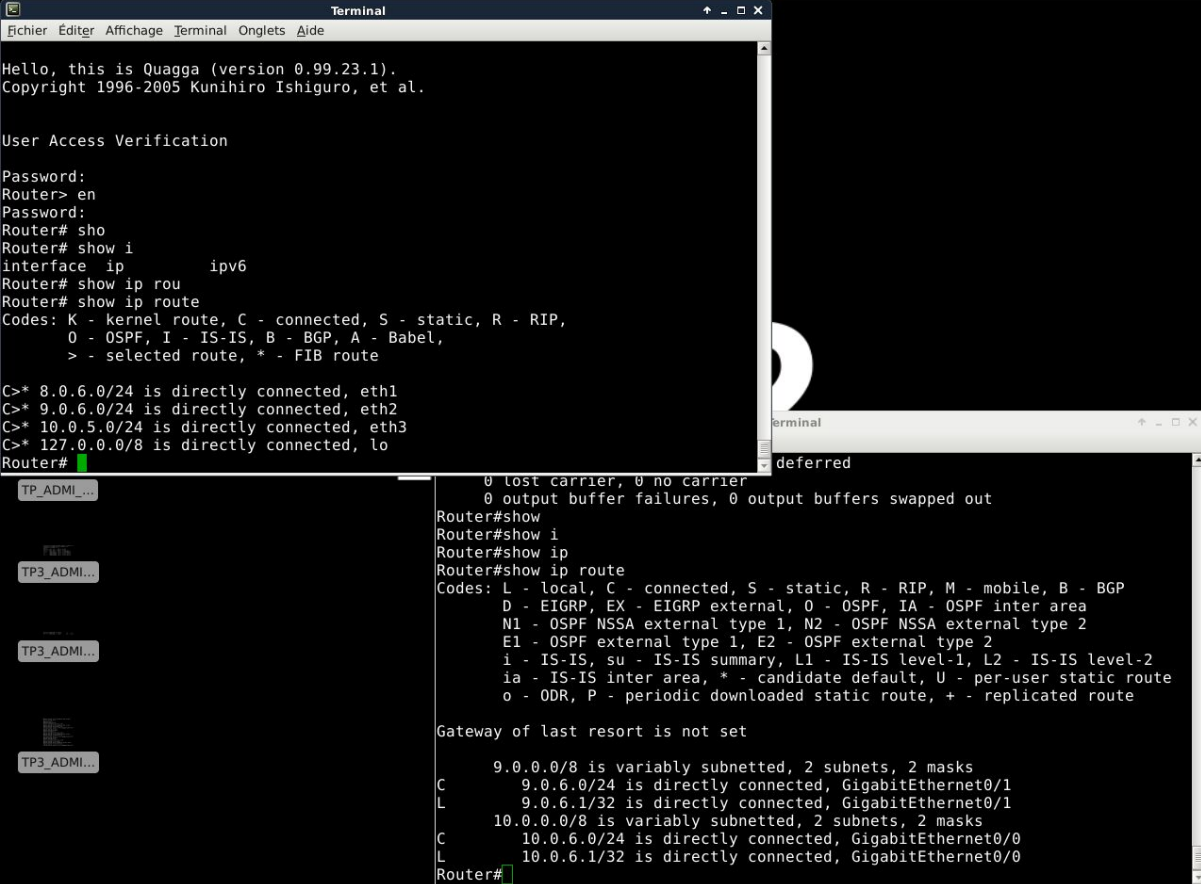
« **write file** » : Commande pour sauvegarder les configurations faites.

Nous obtenons le fichier de configuration pour ripd suivant :

```
GNU nano 2.2.6                                Fichier : ripd.conf
!
! Zebra configuration saved from vty
!   2018/10/17 09:27:27
!
hostname ripd
password zebra
log stdout
!
router rip
  version 2
  redistribute connected
  network 8.0.6.0/24
  network 9.0.6.0/24
  network 10.0.5.0/24
!
line vty
!
```

Question 2 :

Le protocole RIP (Routing Information Protocol) demande aux autres interfaces qui possède quoi comme route, d'où les routes affichées. Il n'y a pas de différences entre les deux tables. Il n'y a pas de route par défaut car nous effectuons une demande de route par défaut.



```
Terminal
Fichier Éditer Affichage Terminal Onglets Aide

Hello, this is Quagga (version 0.99.23.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
Router> en
Password:
Router# sho
Router# show i
interface ip          ipv6
Router# show ip rou
Router# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

C>* 8.0.6.0/24 is directly connected, eth1
C>* 9.0.6.0/24 is directly connected, eth2
C>* 10.0.5.0/24 is directly connected, eth3
C>* 127.0.0.0/8 is directly connected, lo
Router#
```

```
TP_ADMI...
TP3_ADMI...
TP3_ADMI...
TP3_ADMI...

0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
Router#show
Router#show i
Router#show ip
Router#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is not set

 9.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C    9.0.6.0/24 is directly connected, GigabitEthernet0/1
L    9.0.6.1/32 is directly connected, GigabitEthernet0/1
 10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C    10.0.6.0/24 is directly connected, GigabitEthernet0/0
L    10.0.6.1/32 is directly connected, GigabitEthernet0/0
Router#
```

Question 3 :

Les paquets RIP qui circulent sont d'une part, des paquets *Request* qui servent à ses interfaces pour savoir à qui elles sont connectés et d'autre part, des *Response* qui renvoient les adresses des réseaux auxquelles les interfaces sont connectées.

2.3-Configuration routeur Cisco

Rappel:

- ❑ Adresses IP des interfaces GigabitEthernet0/0 **10.0.6.1/24** (routeur/serveur) et GigabitEthernet0/1 **9.0.6.1/24** (routeur/serveur voisin).

Résultat de la commande **show ip route** :

```
Terminal
Fichier Edit Affichage Terminal Onglets Aide

Hello, this is Quagga (version 0.99.23.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
Router> en
Password:
Router# sho
Router# show i
interface ip          ipv6
Router# show ip rou
Router# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

C>* 8.0.6.0/24 is directly connected, eth1
C>* 9.0.6.0/24 is directly connected, eth2
C>* 10.0.5.0/24 is directly connected, eth3
C>* 127.0.0.0/8 is directly connected, lo
Router#
```

Question 1 :

Nous voyons passer des paquets RIPv2, ICMPv3 et ICMP ainsi que des paquets *Request* et des paquets *Response*.

Routing Information Protocol Command: Response (2) Version: RIPv2 (2)	Routing Information Protocol Command: Request (1) Version: RIPv2 (2)
---	--

Time	Source	Destination	Protocol	Length	Info
1 0.000000000	8.0.4.1	224.0.0.9	RIPv2	66	Request
2 0.000034000	8.0.2.1	224.0.0.9	RIPv2	66	Request
3 0.000062000	9.0.2.2	224.0.0.9	RIPv2	66	Request
4 0.007682000	8.0.4.1	224.0.0.22	IGMPv3	54	Membership Report / J
5 0.755647000	8.0.4.1	224.0.0.22	IGMPv3	54	Membership Report / J
6 5.074827000	8.0.4.1	224.0.0.9	RIPv2	146	Response
7 5.074904000	8.0.2.1	224.0.0.9	RIPv2	146	Response
8 5.074922000	9.0.2.2	224.0.0.9	RIPv2	146	Response
9 36.087346000	8.0.4.1	224.0.0.9	RIPv2	146	Response
10 36.087389000	8.0.2.1	224.0.0.9	RIPv2	146	Response
11 36.087417000	9.0.2.2	224.0.0.9	RIPv2	146	Response
12 37.627943000	8.0.4.1	8.0.4.2	ICMP	98	Echo (ping) request
13 37.628376000	8.0.4.2	8.0.4.1	ICMP	98	Echo (ping) reply
14 38.627766000	8.0.4.1	8.0.4.2	ICMP	98	Echo (ping) request
15 38.628159000	8.0.4.2	8.0.4.1	ICMP	98	Echo (ping) reply
16 39.627752000	8.0.4.1	8.0.4.2	ICMP	98	Echo (ping) request
17 39.628124000	8.0.4.2	8.0.4.1	ICMP	98	Echo (ping) reply
18 40.627728000	8.0.4.1	8.0.4.2	ICMP	98	Echo (ping) request
19 40.628106000	8.0.4.2	8.0.4.1	ICMP	98	Echo (ping) reply
20 41.627731000	8.0.4.1	8.0.4.2	ICMP	98	Echo (ping) request

Ici, nous voyons bien les paquets RIPv2, IGMPv3 et ICMP.

Question 2 :

Nous observons sur la table de routage de notre routeur les adresses dynamiques que nous avons configurés.

Partie 3 : Convergence du protocole RIP lors de la panne d'un lien

- A partir des traces affichées sur le **ping**, compter le nombre de paquets perdus. Par la suite, déduire le temps mis à RIP pour trouver un chemin alternatif suite à la perte d'un lien (note : généralement la commande ping envoie approximativement un *ICMP echo_request* toutes les secondes).

Par défaut, le protocole RIP met à jour sa table de routage toutes les 30 secondes. Cette valeur peut varier de 10 à 60 secondes.

Ici, nous avons perdus un total de 30 paquets sur les 100 envoyés. Cela est dû au délai de mise à jour de la table de routage par la protocole RIP.