

# Rapport Travaux Dirigés

## Routage inter-domaine avec BGP

Chaabouni Nadia  
Douézan-Grard Guillaume

*Enseirb-Matmeca*

13 décembre 2015

### Résumé

L'objectif de ce TP est de se familiariser avec le routage inter-domaine ou plus précisément le protocole BGP qui permet d'échanger des informations de routage entre différents systèmes autonomes, que ce soit dans des réseaux à chemins uniques ou multiples. Par ailleurs, il est possible de le faire interagir avec des protocoles de routage intra-domaine pour exporter des préfixes à l'extérieur de ces réseaux. Enfin, on peut choisir de privilégier certains liens, tout en conservant de la robustesse avec le routage politique.

## 1 Réseau avec chemins uniques

Le protocole de communication utilisé par Cisco est le *High-Level Data Link Control*. Deux types de trames sont échangés à l'établissement de la connexion :

- *Serial Line Address Resolution Protocol* : le routeur va essayer à la manière du DHCP d'obtenir une adresse IP sur une interface série et envoie des paquets *keepalive* pour déterminer l'état du lien.
- *Cisco Discovery Protocol* : qui permet aux routeurs de partager leur configuration. On retrouve notamment au début un échange d'information concernant les adresses IP des routeurs, les interfaces concernées, etc. On peut voir ces informations avec `show cdp neighbors` et correspondent aux *Network Layer Reachability Information*.

Lorsque l'on configure les routeurs pour utiliser BGP et que l'on exporte les routes locales sur  $R_1$  et  $R_4$ , on constate l'établissement d'une communication TCP avec le protocole BGP de la façon suivante :

- message OPEN de  $R_1$  à  $R_2$
- message OPEN de  $R_2$  à  $R_1$
- message KEEPALIVE de  $R_2$  à  $R_1$
- message KEEPALIVE de  $R_1$  à  $R_2$
- message UPDATE de  $R_1$  à  $R_2$  qui transmet les informations des réseaux atteignables (en l'occurrence le réseau local de  $R_1$ ) et des informations supplémentaires comme le numéro de Système Autonome, les *next-hop*...
- messages KEEPALIVE périodiques pour vérifier l'état du lien

Les communications réseaux entre  $R_1$  et  $R_2$ , ainsi qu'entre  $R_3$  et  $R_4$  s'établissent automatiquement, mais il a fallu faire en sorte que les routeurs redistribuent ces routes avec BGP. Ceci peut se faire avec la commande (config-router) `redistribute connected`.

## 2 Réseau avec chemins multiples

### 2.1 Distribution des routes

La première étape est de faire en sorte que tous les routeurs BGP redistribuent leurs routes, en particulier pour les liens connectés ce qui est utile pour les routes locales de  $R_1$  et  $R_5$ . Une fois que cela est fait, on constate que les routeurs à l'intérieur des LAN qui ne sont pas des routeurs de bordure ne peuvent pas accéder à la totalité du réseau.

En configurant la redistribution des routes BGP via OSPF et RIP sur les routeurs de bordure des AS 2 et 4, on constate que les tables de routage des routeurs internes sont complétées avec les routes accessibles via les différents routeurs de bordure. Les commandes de redistribution et leur contexte sont explicitées en annexe.

La dernière étape est de faire en sorte que les routeurs de bordure exportent les routes qu'ils ont acquit par OSPF ou RIP avec BGP. De cette façon, l'ensemble du réseau peut accéder aux routeurs internes.

Pour tester, on peut essayer de faire un *ping* de  $R_1$  à  $R_5$  (P11 à P55). On constate que le réseau est bien accessible et avec *traceroute*, on obtient le résultat de la figure 1. Si on fait le *traceroute* inverse, de P55 à P11, on obtient une route différente qui passe par l'AS4 au lieu de l'AS2, le routage est donc asymétrique, comme le montre la figure 2.

Les routes entre P33 et P77 sont symétriques. On constate d'ailleurs une

particularité lorsque l'on fait un *traceroute* entre ces réseaux, plusieurs routes peuvent être empruntées pour s'atteindre mutuellement, comme le montre les figures 3 et 4.

## 2.2 Tolérance aux pannes

### 2.2.1 Liaison $R_6-R_7$

On va simuler une panne sur la liaison  $R_5-R_6$ . Pour cela, on supprime le lien de  $R_6$  à  $R_5$ .

Avec OSPF, des LS UPDATE permettent de mettre à jour les tables de routage. Par exemple, le réseau P55 est maintenant accessible depuis  $R_7$  de façon unique en passant par  $R_1$ . De fait, des LS UPDATE vont être envoyés avec un numéro de séquence incrémenté pour les ID concernés (55.0.0.0). Ainsi ces informations sont persistées dans la LSDB et les routes sont mises à jour sur  $R_7$ .

Avec BGP, dès que le routeur  $R_5$  détecte que le lien est coupé (grâce notamment aux paquets KEEPALIVE et SLARP), il transmet l'information au routeur  $R_4$  en précisant les routes qui sont *withdrawn* et  $R_4$  de son côté informe  $R_5$  que ces réseaux sont accessibles à travers lui-même (avec les NLRI).

$R_4$  de son côté utilise RIP pour retransmettre les informations qu'il a reçu depuis  $R_5$  avec BGP vers  $R_3$ . En particulier, les réseaux qui ne sont maintenant plus accessibles à travers  $R_4$  font l'objet de paquets RIP dans lesquels la métrique est à 16. Pour RIP, la métrique maximale est 15, au delà, les liens sont supprimés. Par exemple, pour le réseau 56.0.0.0 qui ne peut plus être accédé par  $R_5$ , un paquet RIP d'origine  $R_4$  contient ce réseau avec la métrique 16. De son côté,  $R_3$  retransmet les informations à  $R_4$  concernant l'accessibilité de ce réseau via  $R_2$ . La métrique utilisée est 1 puisque ce réseau est accessible depuis le routeur de bordure  $R_2$  qui se trouve à une distance de 1 *hop*.

D'un point de vue durée, la détection de la déconnexion du lien prend maximum 30 secondes avec la temporisation par défaut. Les informations sont ensuite redistribuées immédiatement à travers les différents protocole de routage. Une trentaine de secondes suffisent pour que les tables de routage soit stabilisées sur tous les routeurs et que la connectivité globale soit donc récupérée.

Après la coupure du lien entre  $R_5$  et  $R_6$ , on constate bien entendu que le *traceroute* a évolué comme cela est montré sur la figure 5.

### 2.2.2 Liaison $R_6-R_8$

On peut faire une coupure de la même façon entre  $R_6$  et  $R_8$ . Soumis aussi à un *timeout* de 30 secondes, la détection de la coupure du lien entraîne l'émission de messages BGP et OSPF aux routeurs adjacents sous la forme énoncée précédemment. Pour l'AS 2, ce changement n'entraîne pas de modifications notables puisque la coupure s'est faite intra-AS 4. Cependant, la nouvelle route déterminée entre P11 et P55 passe systématiquement par l'AS 2 comme précédemment. Mais contrairement à un coupure empêchant la liaison, cette fois-ci c'est la perturbation des routes existantes passant par l'AS 4 qui entraîne la préférence pour les routes passant par l'AS 2.

### 2.2.3 Désactivation du préfixe P55

De la même façon, la désactivation du préfixe P55 va entraîner sa suppression dans l'AS 4 avec les LS UPDATE de OSPF, des *withdraw* avec BGP et des messages RIP avec une métrique de 16 pour la suppression à l'intérieur de l'AS 2.

## 3 Réseau avec chemins multiples et routage politique

### 3.1 Filtrage par route

On filtre la divulgation des préfixes P33 et P77 depuis les routeurs de bordure des AS 2 et 4. De fait, on isole ces deux derniers. Après avoir forcé le nettoyage des tables de routage, les adresses correspondant à ces préfixes ne sont plus présentes dans les AS 1 et 3, et sont donc cantonnées dans leur AS de définition.

### 3.2 Filtrage par AS

Pour filtrer par AS pour le routeur  $R_5$ , il faut associer une liste de filtrage en entrée pour ses voisins, en empêchant les routes originaires de l'AS 2 d'être apprises. Pour cela, on peut utiliser l'expression régulière `_2$`.

Les routes transmises par l'AS 2 mais qui ne sont pas originaires de ce dernier sont bien entendu distribuées et donc, on peut toujours accéder à P11 depuis l'AS 3.

### 3.3 Politique de routage

A ce stade, nous voulons que le trafic avec l'AS 1 passe (dans les 2 sens) par l'AS 2 mais en gardant la possibilité de passer par l'AS 4 en cas de coupure avec l'AS 2. Pour cela, il suffit que les routes émises par  $R_5$  vers l'AS 2 aient la métrique la plus faible. Ainsi, non seulement  $R_5$  privilégiera le passage par l'AS 2 pour accéder à P11, mais en retour,  $R_1$  privilégiera aussi ces routes pour accéder à  $R_5$ .

Maintenant, pour changer la politique de l'AS 3 afin que le trafic sortant depuis l'AS 1 passe via l'AS 2 alors que le trafic entrant de la même provenance passe via l'AS 4, il suffit d'attribuer la métrique la plus faible aux routes provenant de l'AS 2 en sortie ainsi que pour les routes provenant de l'AS 4 en entrée en utilisant les options `in` et `out`.

### 3.4 Des observateurs de tables BGP

Originellement, <http://www.routeviews.org> a pour objectif de rendre accessible les tables de routage BGP de certains routeurs du backbone pour différents objectifs, comme par exemple voir comment ces routeurs perçoivent certains préfixes exportés pour vérifier des configurations ou même analyser l'évolution des routes BGP.

## A *Traceroutes*

### A.1 Réseaux P11 et P55

```
R1#traceroute 55.0.0.1
 1 18.0.0.2 32 msec 20 msec 20 msec
 2 78.0.0.1 [AS 4] 20 msec 40 msec 44 msec
 3 67.0.0.1 [AS 4] 68 msec 64 msec 56 msec
 4 56.0.0.1 [AS 4] 52 msec 104 msec 84 msec
```

FIGURE 1 – *Traceroute* de P11 à P55

```
R5#traceroute 11.0.0.1
 1 45.0.0.1 36 msec 20 msec 20 msec
 2 34.0.0.1 [AS 2] 56 msec 68 msec 72 msec
 3 23.0.0.1 [AS 2] 68 msec 72 msec 72 msec
 4 12.0.0.1 [AS 1] 100 msec 68 msec 72 msec
```

FIGURE 2 – *Traceroute* de P11 à P55

### A.2 Réseaux P33 et P77

```
R3#traceroute 77.0.0.1
 1 34.0.0.2 68 msec
   23.0.0.1 60 msec
   34.0.0.2 64 msec
 2 12.0.0.1 64 msec
   45.0.0.2 64 msec
   12.0.0.1 96 msec
 3 56.0.0.2 88 msec
   18.0.0.2 92 msec
   56.0.0.2 88 msec
 4 78.0.0.1 128 msec
   67.0.0.2 124 msec
   78.0.0.1 120 msec
```

FIGURE 3 – *Traceroute* de P33 à P77

```
R7#traceroute 33.0.0.1
 1 78.0.0.2 8 msec
   67.0.0.1 20 msec
   78.0.0.2 20 msec
 2 56.0.0.1 20 msec
   18.0.0.1 40 msec
   56.0.0.1 40 msec
 3 12.0.0.2 72 msec
   45.0.0.1 48 msec
   12.0.0.2 64 msec
 4 34.0.0.1 48 msec
   23.0.0.2 72 msec
   34.0.0.1 68 msec
```

FIGURE 4 – *Traceroute* de P77 à P33

### A.3 Coupure $R_5 - R_6$

```
R5#traceroute 11.0.0.1
 1 45.0.0.1 12 msec 20 msec 20 msec
 2 34.0.0.1 [AS 2] 24 msec 40 msec 40 msec
 3 23.0.0.1 [AS 2] 68 msec 64 msec 60 msec
 4 12.0.0.1 [AS 2] 88 msec 84 msec 80 msec
```

FIGURE 5 – *Traceroute* de P55 à P11 avec coupure  $R_5 - R_6$

## B Tables de routage

```
B 34.0.0.0/8 [20/1] via 12.0.0.2, 00:01:38
B 68.0.0.0/8 [20/0] via 18.0.0.2, 00:01:38
B 33.0.0.0/8 [20/1] via 12.0.0.2, 00:01:38
C 18.0.0.0/8 is directly connected, Serial1/1
B 55.0.0.0/8 [20/1] via 12.0.0.2, 00:01:07
B 23.0.0.0/8 [20/0] via 12.0.0.2, 00:01:38
B 67.0.0.0/8 [20/0] via 18.0.0.2, 00:01:38
B 77.0.0.0/8 [20/0] via 18.0.0.2, 00:01:38
B 78.0.0.0/8 [20/0] via 18.0.0.2, 00:01:38
B 56.0.0.0/8 [20/0] via 18.0.0.2, 00:01:38
C 11.0.0.0/8 is directly connected, Loopback0
C 12.0.0.0/8 is directly connected, Serial1/0
B 45.0.0.0/8 [20/1] via 12.0.0.2, 00:01:38
```

FIGURE 6 – Table de routage de  $R_1$

```
R 34.0.0.0/8 [120/1] via 23.0.0.2, 00:00:15, Serial1/1
B 68.0.0.0/8 [20/0] via 12.0.0.1, 00:02:32
R 33.0.0.0/8 [120/1] via 23.0.0.2, 00:00:15, Serial1/1
B 18.0.0.0/8 [20/0] via 12.0.0.1, 00:02:32
R 55.0.0.0/8 [120/1] via 23.0.0.2, 00:00:15, Serial1/1
C 23.0.0.0/8 is directly connected, Serial1/1
B 67.0.0.0/8 [20/0] via 12.0.0.1, 00:02:32
B 77.0.0.0/8 [20/0] via 12.0.0.1, 00:02:32
B 78.0.0.0/8 [20/0] via 12.0.0.1, 00:02:32
B 56.0.0.0/8 [20/0] via 12.0.0.1, 00:02:32
B 11.0.0.0/8 [20/0] via 12.0.0.1, 00:02:32
C 12.0.0.0/8 is directly connected, Serial1/0
R 45.0.0.0/8 [120/1] via 23.0.0.2, 00:00:15, Serial1/1
```

FIGURE 7 – Table de routage de  $R_2$

```

C 34.0.0.0/8 is directly connected, Serial1/1
R 68.0.0.0/8 [120/1] via 34.0.0.2, 00:00:05, Serial1/1
          [120/1] via 23.0.0.1, 00:00:03, Serial1/0
C 33.0.0.0/8 is directly connected, Loopback0
R 18.0.0.0/8 [120/1] via 34.0.0.2, 00:00:05, Serial1/1
          [120/1] via 23.0.0.1, 00:00:04, Serial1/0
R 55.0.0.0/8 [120/1] via 34.0.0.2, 00:00:05, Serial1/1
C 23.0.0.0/8 is directly connected, Serial1/0
R 67.0.0.0/8 [120/1] via 34.0.0.2, 00:00:05, Serial1/1
          [120/1] via 23.0.0.1, 00:00:04, Serial1/0
R 77.0.0.0/8 [120/1] via 34.0.0.2, 00:00:05, Serial1/1
          [120/1] via 23.0.0.1, 00:00:04, Serial1/0
R 78.0.0.0/8 [120/1] via 34.0.0.2, 00:00:05, Serial1/1
          [120/1] via 23.0.0.1, 00:00:04, Serial1/0
R 56.0.0.0/8 [120/1] via 34.0.0.2, 00:00:07, Serial1/1
          [120/1] via 23.0.0.1, 00:00:05, Serial1/0
R 11.0.0.0/8 [120/1] via 23.0.0.1, 00:00:06, Serial1/0
R 12.0.0.0/8 [120/1] via 23.0.0.1, 00:00:06, Serial1/0
R 45.0.0.0/8 [120/1] via 34.0.0.2, 00:00:08, Serial1/1

```

FIGURE 8 – Table de routage de  $R_3$

```

C 34.0.0.0/8 is directly connected, Serial1/0
B 68.0.0.0/8 [20/0] via 45.0.0.2, 00:03:39
R 33.0.0.0/8 [120/1] via 34.0.0.1, 00:00:00, Serial1/0
B 18.0.0.0/8 [20/0] via 45.0.0.2, 00:03:39
B 55.0.0.0/8 [20/0] via 45.0.0.2, 00:03:39
R 23.0.0.0/8 [120/1] via 34.0.0.1, 00:00:00, Serial1/0
B 67.0.0.0/8 [20/0] via 45.0.0.2, 00:03:39
B 77.0.0.0/8 [20/0] via 45.0.0.2, 00:03:39
B 78.0.0.0/8 [20/0] via 45.0.0.2, 00:03:39
B 56.0.0.0/8 [20/0] via 45.0.0.2, 00:03:39
R 11.0.0.0/8 [120/1] via 34.0.0.1, 00:00:00, Serial1/0
R 12.0.0.0/8 [120/1] via 34.0.0.1, 00:00:00, Serial1/0
C 45.0.0.0/8 is directly connected, Serial1/1

```

FIGURE 9 – Table de routage de  $R_4$

```

B 34.0.0.0/8 [20/0] via 45.0.0.1, 00:03:54
B 68.0.0.0/8 [20/0] via 56.0.0.2, 00:03:54
B 33.0.0.0/8 [20/1] via 45.0.0.1, 00:03:54
B 18.0.0.0/8 [20/0] via 56.0.0.2, 00:03:54
C 55.0.0.0/8 is directly connected, Loopback0
B 23.0.0.0/8 [20/1] via 45.0.0.1, 00:03:54
B 67.0.0.0/8 [20/0] via 56.0.0.2, 00:03:54
B 77.0.0.0/8 [20/20] via 56.0.0.2, 00:03:54
B 78.0.0.0/8 [20/0] via 56.0.0.2, 00:03:54
C 56.0.0.0/8 is directly connected, Serial1/1
B 11.0.0.0/8 [20/1] via 45.0.0.1, 00:03:24
B 12.0.0.0/8 [20/1] via 45.0.0.1, 00:03:54
C 45.0.0.0/8 is directly connected, Serial1/0

```

FIGURE 10 – Table de routage de  $R_5$

```

B 34.0.0.0/8 [20/0] via 56.0.0.1, 00:04:13
C 68.0.0.0/8 is directly connected, Serial1/2
B 33.0.0.0/8 [20/0] via 56.0.0.1, 00:04:13
O E2 18.0.0.0/8 [110/20] via 67.0.0.2, 00:05:00, Serial1/1
B 55.0.0.0/8 [20/0] via 56.0.0.1, 00:04:13
B 23.0.0.0/8 [20/0] via 56.0.0.1, 00:04:13
C 67.0.0.0/8 is directly connected, Serial1/1
O E2 77.0.0.0/8 [110/20] via 67.0.0.2, 00:05:00, Serial1/1
O IA 78.0.0.0/8 [110/128] via 67.0.0.2, 00:05:00, Serial1/1
C 56.0.0.0/8 is directly connected, Serial1/0
O E2 11.0.0.0/8 [110/1] via 67.0.0.2, 00:04:12, Serial1/1
O E2 12.0.0.0/8 [110/1] via 67.0.0.2, 00:04:12, Serial1/1
B 45.0.0.0/8 [20/0] via 56.0.0.1, 00:04:13

```

FIGURE 11 – Table de routage de  $R_6$

```

0 E2 34.0.0.0/8 [110/1] via 78.0.0.2, 00:04:36, Serial1/1
              [110/1] via 67.0.0.1, 00:04:36, Serial1/0
0 E2 68.0.0.0/8 [110/20] via 78.0.0.2, 00:05:24, Serial1/1
              [110/20] via 67.0.0.1, 00:05:24, Serial1/0
0 E2 33.0.0.0/8 [110/1] via 78.0.0.2, 00:04:36, Serial1/1
              [110/1] via 67.0.0.1, 00:04:36, Serial1/0
0 E2 18.0.0.0/8 [110/20] via 78.0.0.2, 00:05:24, Serial1/1
0 E2 55.0.0.0/8 [110/1] via 67.0.0.1, 00:04:37, Serial1/0
0 E2 23.0.0.0/8 [110/1] via 78.0.0.2, 00:04:36, Serial1/1
              [110/1] via 67.0.0.1, 00:04:36, Serial1/0
C   67.0.0.0/8 is directly connected, Serial1/0
C   77.0.0.0/8 is directly connected, Loopback0
C   78.0.0.0/8 is directly connected, Serial1/1
0 E2 56.0.0.0/8 [110/20] via 67.0.0.1, 00:05:25, Serial1/0
0 E2 11.0.0.0/8 [110/1] via 78.0.0.2, 00:04:38, Serial1/1
0 E2 12.0.0.0/8 [110/1] via 78.0.0.2, 00:04:33, Serial1/1
0 E2 45.0.0.0/8 [110/1] via 67.0.0.1, 00:04:39, Serial1/0

```

FIGURE 12 – Table de routage de  $R_7$

```

B   34.0.0.0/8 [20/0] via 18.0.0.1, 00:05:10
C   68.0.0.0/8 is directly connected, Serial1/0
B   33.0.0.0/8 [20/0] via 18.0.0.1, 00:05:10
C   18.0.0.0/8 is directly connected, Serial1/2
0 E2 55.0.0.0/8 [110/1] via 78.0.0.1, 00:05:11, Serial1/1
B   23.0.0.0/8 [20/0] via 18.0.0.1, 00:05:10
0 IA 67.0.0.0/8 [110/128] via 78.0.0.1, 00:05:58, Serial1/1
0 E2 77.0.0.0/8 [110/20] via 78.0.0.1, 00:05:58, Serial1/1
C   78.0.0.0/8 is directly connected, Serial1/1
0 E2 56.0.0.0/8 [110/20] via 78.0.0.1, 00:05:58, Serial1/1
B   11.0.0.0/8 [20/0] via 18.0.0.1, 00:05:10
B   12.0.0.0/8 [20/0] via 18.0.0.1, 00:05:10
0 E2 45.0.0.0/8 [110/1] via 78.0.0.1, 00:05:11, Serial1/1

```

FIGURE 13 – Table de routage de  $R_8$

## C *Cheatsheet*

### C.1 Configuration d'une interface

```
(config)# interface <interface_name>
(config-if)# ip address <ip> <netmask>
(config-if)# no shutdown
```

### C.2 Informations de configuration

```
# show interfaces
# show ip route
# show cdp neighbors
```

### C.3 Configuration BGP

```
(config)# router bgp <as>
(config-router)# network <net>
(config-router)# neighbor <remote_ip> remote-as <as>
    Pour enlever un voisin :
# clear ip bgp <neighbor_ip>
    Pour enlever la configuration du routeur :
(config)# no router bgp
```

### C.4 Configuration RIP

```
(config)# router rip
(config-router)# network <net>
```

### C.5 Configuration OSPF

```
(config)# interface <ifname>
(config-if)# ip ospf <pid> area <ar>
(config)# router ospf <pid>
(config-router)# network <net> 0.0.0.0 area <ar>
(config-router)# redistribute bgp
(config-router)# redistribute connected
```

### C.6 Enregistrer la configuration

```
# copy run start
```

## C.7 Liste de divulgation

```
(config)# config bgp <as>
(config-router)# neighbor <remote_ip> distribute-list <id> out
(config-router)# exit
(config)# access-list <id> deny <netw> <wildcard netmask>
(config)# access-list <id> permit all
```