

Description du thème

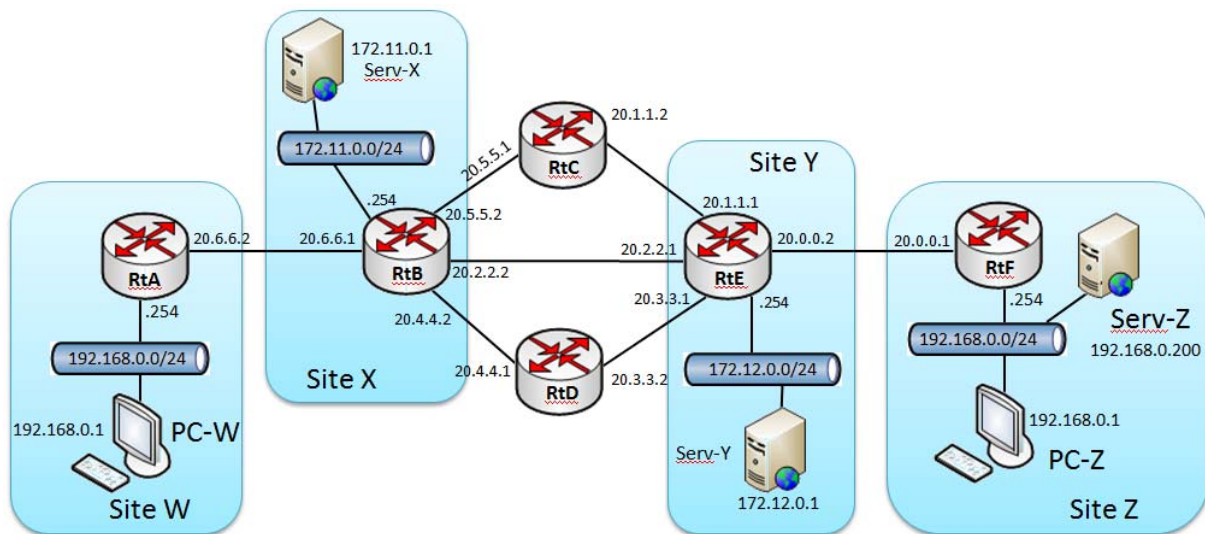
Propriétés	Description
Intitulé long	<p><i>Activité Packet-Tracer de découverte et de mise en pratique :</i></p> <p>Du NAT dynamique et statique Du routage dynamique avec le protocole RIP, en version 2</p> <p>Maquette à compléter par étape (au départ, équipements paramétrés au niveau des interfaces uniquement)</p>
Formation concernée	BTS Services Informatiques aux Organisations
Matière	SISR2 Conception des infrastructures réseaux SISR3 Exploitation des services
Présentation	Cette activité a pour but de découvrir le paramétrage et le fonctionnement du protocole de routage dynamique RIP et ceux de la translation d'adresse NAT.
Notions	<p>Activités</p> <p>A1.3.2 Définition des éléments nécessaires à la continuité d'un service A3.1.3 Prise en compte du niveau de sécurité nécessaire à une infrastructure A3.2.1 Installation et configuration d'éléments d'infrastructure</p> <p>Savoir-faire</p> <ul style="list-style-type: none"> • Configurer une maquette ou un prototype pour valider une solution • Installer et configurer les éléments nécessaires à la qualité et à la continuité du service
Transversalité	<p>SI5 : Activités :</p> <ul style="list-style-type: none"> - D3.3 - Administration et supervision d'une infrastructure - A3.3.1 Administration sur site ou à distance des éléments d'un réseau, de serveurs, de services et d'équipements <p>Savoir-faire :</p> <ul style="list-style-type: none"> - Installer, configurer et administrer un service
Pré-requis	Une connaissance de base de l'outil Packet Tracer pour créer la maquette et des notions de base sur le routage.
Outils	Packet Tracer Student 6.2.0
Mots-clés	Packet Tracer, Activité, Maquette, Cisco, Routage, NAT, RIP, RIPV2, NAT/PAT
Durée	2 à 3 heures
Niveau de difficulté	Moyenne (6/10) <i>avec une maîtrise préalable de Packet Tracer et une connaissance de base du routage.</i>
Auteur(es)	Denis Gallot / David Duron, avec le concours d'Eve-Marie Gallot et de Patrice Dignan, relecture Gaëlle Castel.
Version	v 1.0
Date de publication	Mars 2016
Contenu du package	Document présentant les objectifs et les paramétrages demandés ainsi que des éléments de correction. Fichiers .pkt de départ et .par étapes. Schéma de l'infrastructure. Configurations des routeurs. Fichier d'activité .pka.



Objectifs

- **Mettre en place du NAT** sur deux sites distants « W » et « Z » qui communiquent via un réseau public de routeurs et qui sont dotés du même plan d'adressage en 192.168.0.0/24.
- **Mettre en place du routage dynamique** dans un réseau de routeurs et mettre en évidence la tolérance de panne.
- **Mettre en place du NAT Statique** pour permettre l'accès extérieur à un serveur privé

Point de départ



Le schéma comporte 4 sites. Les deux sites d'extrémité W et Z utilisent le même réseau 192.168.0.0/24. On peut assimiler ces deux sites à des particuliers dotés d'une « box ». Sur le site X, un réseau 172.11.0.0/24 comporte un serveur Web Serv-X d'adresse 172.11.0.1. Sur le site Y, un autre serveur Web (Serv-Y) est accessible à l'adresse 172.12.0.1 sur le réseau 172.12.0.0/24.

Conséquence :

Le protocole RIP n'est pas utilisable sur les sites terminaux W et Z, du fait de l'utilisation du même réseau IP par ces deux sites.

C'est tout l'objectif de cet Exolab, qui va justement montrer comment activer le NAT ou le NAT/PAT (*Network Address Translation/ Port Address Translation*) sur les routeurs CISCO, pour masquer les adresses privées et permettre l'acheminement des requêtes vers ou depuis le réseau « public ».

Le fichier Packet Tracer de départ comporte des routeurs 1841 et des routeurs 2811 auxquels on a ajouté des modules NM4E pourvus de 4 interfaces Ethernet. Dans ce fichier, seules les interfaces ont été paramétrées selon le schéma ci-dessus.

Tableau des réseaux de connexion entre routeurs

Du routeur vers le routeur	RtA	RtB	RtC	RtD	RtE	RtF
RtA	-	20.6.6.0/30	-	-	-	-
RtB	20.6.6.0/30	-	20.5.5.0/30	20.4.4.0/30	20.2.2.0/30	-
RtC	-	20.5.5.0/30	-	-	20.1.1.0/30	-
RtD	-	20.4.4.0/30	-	-	20.3.3.0/30	-
RtE	-	20.2.2.0/30	20.1.1.0/30	20.3.3.0/30	-	20.0.0.0/30
RtF	-	-	-	-	20.0.0.0/30	-

Première étape

TRAVAIL À FAIRE

1. Vérifier la connectivité des postes et des serveurs vers leur passerelle et la connectivité de chaque routeur avec son – ou ses – voisin(s).

Deuxième étape

Mise en place des routes par défaut

TRAVAIL À FAIRE

2. Mettre en place les routes par défaut dans les routeurs terminaux RtA et RtF. Consultez ensuite la table de routage et vérifiez la présence de ces routes par défaut sur chaque routeur.

Troisième étape

Sur les routeurs terminaux RtA et RtF, mettre en place le NAT. À l'issue de cette étape, les postes clients des sites W et Z devront être capables de joindre le second routeur (RtB pour le site W et RtE pour le site Z). **Vous comprenez pourquoi ?**

Utiliser les exemples ci-dessous pour effectuer cette étape.

Exemple de mise en place du NAT

Le service NAT permet de traduire les adresses IP internes non-unicas et souvent non routables d'un réseau local à un ensemble d'adresses externes unicas et routables. Ce mécanisme permet de faire correspondre une seule adresse externe publique visible sur Internet à toutes les adresses d'un réseau privé, et pallie ainsi l'épuisement des adresses IPv4.

Il existe deux types de NAT :

NAT statique : le principe du NAT statique consiste à associer une adresse IP publique à une adresse IP privée interne au réseau. La translation statique ne résout pas le problème de la pénurie d'adresse dans la mesure où N adresses IP routables sont nécessaires pour connecter N machines du réseau interne.

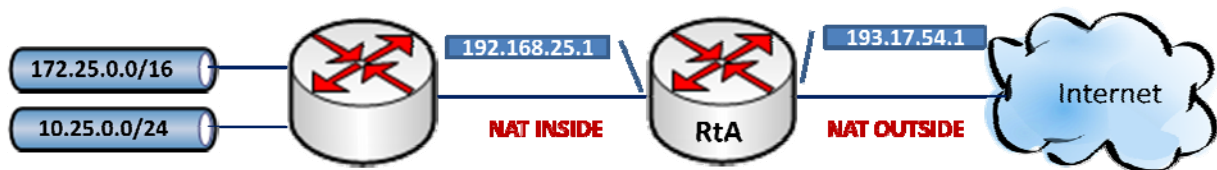
NAT dynamique : Le NAT dynamique fonctionne sur le même principe que le NAT statique mis à part que l'adresse publique attribuée ne sera pas toujours la même. Cette adresse sera choisie dans un "pool d'adresses" à la disposition du routeur et attribuée pour un temps défini.

Le NAT dynamique permet donc de partager un nombre réduit d'adresses IP routables entre plusieurs machines en adressage privé.

Dans le cas du partage d'une seule adresse publique routable, toutes les machines du réseau interne possèdent virtuellement, vu de l'extérieur, la même adresse IP. C'est le NAT/PAT.

Le NAT/PAT (ou NAPT) est donc un cas particulier du NAT dynamique, dans lequel on utilise une seule adresse publique (et pas un pool) ; cette adresse sera associée à un port choisi dynamiquement par le routeur pour identifier chaque demande sortante.

Exemple de mise en œuvre du NAT dynamique - Attention : ce n'est pas – et c'est volontaire – l'exemple de la maquette)



1. Définition des côtés interne et externe du NAT sur les interfaces:

```
RTA(CONFIG)# INTERFACE FA0  
RTA(CONFIG-IF)# IP ADDRESS 192.168.25.1 255.255.255.0  
RTA(CONFIG-IF) # IP NAT INSIDE  
RTA(CONFIG)# INTERFACE FA1  
RTA(CONFIG-IF)# IP ADDRESS 193.17.54.1 255.255.255.0  
RTA(CONFIG-IF) # IP NAT OUTSIDE
```

2. Création d'une ACL (*access list*) autorisant les deux réseaux internes

```
RTA(CONFIG)# ACCESS-LIST 1 PERMIT 172.25.0.0 0.0.255.255  
RTA(CONFIG)# ACCESS-LIST 1 PERMIT 10.25.0.0 0.255.255.255
```

Cette commande utilise des masques génériques : l'effet du masque générique 0.0.0.255 indique au routeur de ne prendre en compte que les 3 premiers octets et d'ignorer le dernier octet. Pour les *access-lists*, on utilise en quelque sorte les masques de sous-réseau inversés : les bits positionnés sont les bits que l'on ne vérifie pas.

3. Activation effective du NAT (avec la solution 1)

On utilise donc l'adresse externe (côté *outside*) pour la translation d'adresses :

```
RTA(CONFIG)# IP NAT INSIDE SOURCE LIST 1 INTERFACE FA1 OVERLOAD
```

Bien comprendre : on autorise les machines [internes] respectant l'*access-list* 1 à utiliser le NAT/PAT en sortie sur l'interface Fa1. Le mot-clé *overload* signifie qu'on utilise le PAT (*Port Address Translation*), appelé parfois aussi « *single-address NAT* » (on utilise une seule adresse publique associée à différents ports).

ou Solution 2 : création d'une étendue d'adresses utilisées pour la translation d'adresses :

```
RTA(CONFIG)# IP NAT POOL SORTIE 193.17.54.100 193.17.54.150 NETMASK 255.255.255.0
```

Cette commande ajoute un "pool" d'adresses appelé *sortie* et utilisant la plage d'adresses 193.17.54.100 à 193.17.54.150. Cette plage permet 51 translations simultanées sur le routeur.

On associe le tout pour mettre en œuvre le NAT.

```
RTA(CONFIG)# IP NAT INSIDE SOURCE LIST 1 POOL SORTIE
```

Notes :

- Attention, dans la solution 2, il faut laisser de la place pour la plage d'adresses de translation dans le réseau « *Outside* ». On ne peut donc pas utiliser un réseau en /30.
- En Packet Tracer, la commande IP NAT POOL utilise le mot-clé « NETMASK » alors qu'on trouve « PREFIX » en IOS natif.

Exemple de mise en œuvre du NAT statique :

Cela permet de mettre en œuvre un mappage statique par exemple, d'un serveur web interne vers l'adresse IP publique du routeur.

```
RTA(CONFIG)# IP NAT INSIDE SOURCE STATIC TCP 20.25.0.1 80 193.17.54.1 80
```

Ici, il s'agit en fait d'une *redirection de port*. Le NAT statique permet éventuellement de joindre une machine privée, pour l'ensemble de ses communications, mais ici le « *mapping* » ne concerne qu'un seul port, le port 80 : tout le trafic qui arrive sur l'interface publique 193.17.54.1 à destination du port 80 sera redirigé sur la machine d'adresse privée 20.25.0.1, sur le même port 80.

Cette commande permettra donc d'accéder au serveur HTTP intranet depuis l'extérieur du réseau : les requêtes HTTP (port 80) envoyées à l'adresse publique du routeur sont traduites vers une adresse privée du réseau interne sur lequel est installé le serveur HTTP.

Note : vous pourrez l'utiliser - dans l'étape n°6 - pour atteindre Serv-Z, depuis PC-W.

Affichage des translations actives

RTA# **SHOW IP NAT TRANSLATIONS**

PRO	INSIDE GLOBAL	INSIDE LOCAL	OUTSIDE LOCAL	OUTSIDE GLOBAL	
ICMP	193.17.54.1:714	172.25.5.2:714	201.27.52.3:714	201.27.52.3:714	translations dynamiques
ICMP	193.17.54.1:715	172.25.5.2:715	201.27.52.3:715	201.27.52.3:715	
ICMP	193.17.54.1:1041	172.25.6.44:716	201.27.52.3:697	201.27.52.3:1041	
ICMP	193.17.54.1:1042	172.25.6.44:717	201.27.52.3:698	201.27.52.3:1042	
TCP	193.17.54.1:1302	172.25.7.3:714	85.27.4.69:80	85.27.4.69:80	
...					
TCP	193.17.54.1:80	10.25.0.1:80	---	---	Translation statique



INSIDE		OUTSIDE	
LOCAL	GLOBAL	LOCAL	GLOBAL
172.25.7.3:714	85.27.4.69:80	193.17.54.1:1302	85.27.4.69:80

Afin de pouvoir partager les différentes adresses IP sur une adresse IP routable, le NAT dynamique utilise le mécanisme de translation de port (PAT - *Port Address Translation*), c'est-à-dire l'affectation d'un port source différent à chaque requête, de manière à pouvoir maintenir une correspondance entre les requêtes provenant du réseau interne et les réponses des machines sur Internet, toutes adressées à l'adresse IP du routeur (normalement les ports dynamiquement ouverts devraient tous être supérieurs à 1024, mais Packet-Tracer ne respecte pas tout à fait cela).

RTA# **DEBUG IP NAT**

Cette commande permet d'afficher la gestion des translations NAT.

TRAVAIL À FAIRE

3. Mise en place d'un NAT dynamique sur les routeurs terminaux RtA et RtF
NB : utiliser la solution 1 puisque les réseaux inter-routeurs sont en préfixe /30.

Vérifiez, comme suggéré au début de cette étape, que PC-W et PW-Z peuvent *pinguer* respectivement RtB et RtE (sur n'importe laquelle de leurs interfaces d'ailleurs).

Vérifiez également le contenu de la table NAT/PAT :

- Soit par la commande Cisco : `show ip nat translations`

```

Routeur-A#sh ip nat translations
Pro  Inside global      Inside local      Outside local      Outside global
icmp 20.6.6.2:10         192.168.0.1:10   20.6.6.1:10       20.6.6.1:10
icmp 20.6.6.2:11         192.168.0.1:11   20.6.6.1:11       20.6.6.1:11
icmp 20.6.6.2:12         192.168.0.1:12   20.6.6.1:12       20.6.6.1:12
icmp 20.6.6.2:9          192.168.0.1:9    20.6.6.1:9        20.6.6.1:9

```

- Le mode *debug* fonctionne aussi (ne pas oublier de l'annuler : **no debug ip nat**) :

```

Routeur-A#debug ip nat
IP NAT debugging is on
Routeur-A#
NAT: s=192.168.0.1->20.6.6.2, d=20.6.6.1 [21]

NAT*: s=20.6.6.1, d=20.6.6.2->192.168.0.1 [73]

```

- Soit par l'interface graphique de Packet Tracer : outil loupe / clic sur Router-A / ARP Table

Protocol	Inside Global	Inside Local	Outside Local	Outside Global
icmp	20.6.6.2:10	192.168.0.1:10	20.6.6.1:10	20.6.6.1:10
icmp	20.6.6.2:11	192.168.0.1:11	20.6.6.1:11	20.6.6.1:11
icmp	20.6.6.2:12	192.168.0.1:12	20.6.6.1:12	20.6.6.1:12
icmp	20.6.6.2:9	192.168.0.1:9	20.6.6.1:9	20.6.6.1:9

Quatrième étape

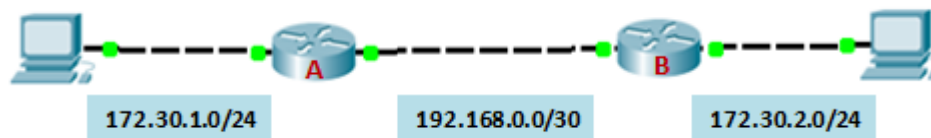
Mettre en place le routage dynamique dans les routeurs de liaison RtB, RtC, RtD et RtE.

Le principe est que chaque routeur annonce à ses voisins les réseaux qui lui sont connectés directement et les réseaux connus par RIP. Au final, tous les routeurs connectés par RIP connaissent tous les réseaux.

Note :

RIPv1 ne prenant en charge les masques de sous-réseau de longueur variable (on dit qu'il est *classfull*) ni l'authentification des routeurs. RIPv2 (*classless*) a été conçu pour permettre au protocole de répondre aux contraintes des réseaux actuels (découpages des réseaux IP en sous-réseaux, authentification par mot de passe ...).

Exemple de paramétrage de RIPv2 :



```

A(CONFIG)# ROUTER RIP
A(CONFIG-ROUTER)# VERSION 2
A(CONFIG-ROUTER)# NETWORK 172.30.1.0
A(CONFIG-ROUTER)# NETWORK 192.168.0.0
A(CONFIG-ROUTER)# NO AUTO-SUMMARY
A#SH IP ROUTE
C 172.30.1.0 IS DIRECTLY CONNECTED, FA0/0

```

```

B(CONFIG)# ROUTER RIP
B(CONFIG-ROUTER)# VERSION 2
B(CONFIG-ROUTER)# NETWORK 172.30.2.0
B(CONFIG-ROUTER)# NETWORK 192.168.0.0
B(CONFIG-ROUTER)# NO AUTO-SUMMARY
B#SH IP ROUTE
R 172.30.1.0 [120/1] VIA 192.168.0.1, FA0/1

```

R 172.30.2.0 [120/1] VIA 192.168.0.2, FA0/1 C 192.168.0.0 IS DIRECTLY CONNECTED, FA0/1	C 172.30.2.0 IS DIRECTLY CONNECTED, FA0/0 C 192.168.0.0 IS DIRECTLY CONNECTED, FA0/1
---	---

Note : le « NO AUTO SUMMARY » permet de travailler avec des masques non « standards ».

TRAVAIL À FAIRE

4. Vous devez donc, pour chaque routeur de liaison RtB, RtC, RtD et RtE activer le protocole RIPv2 et déclarer les réseaux qui lui sont directement connectés.
5. À l'issue de cette étape, vous devez vérifier que chaque routeur connaît la route vers tous les réseaux de liaison « 20.x.x.x » et les réseaux des serveurs « 172.x.x.x ».

NB : Comme indiqué précédemment, les réseaux privés (192.168.0.0/24) ne seront pas déclarés au niveau de RIP dans notre cas de figure.

Cinquième étape

Vérification du comportement du protocole RIP.

TRAVAIL À FAIRE

6. À l'aide de commandes (PING, TRACERT), vous mettrez en évidence le comportement du protocole RIP et la tolérance de panne qu'il permet.

Sixième étape

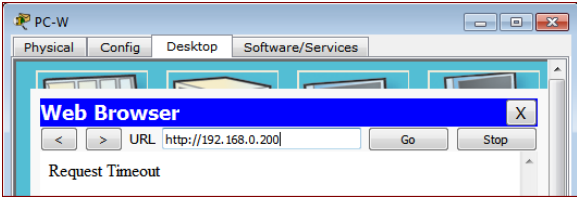
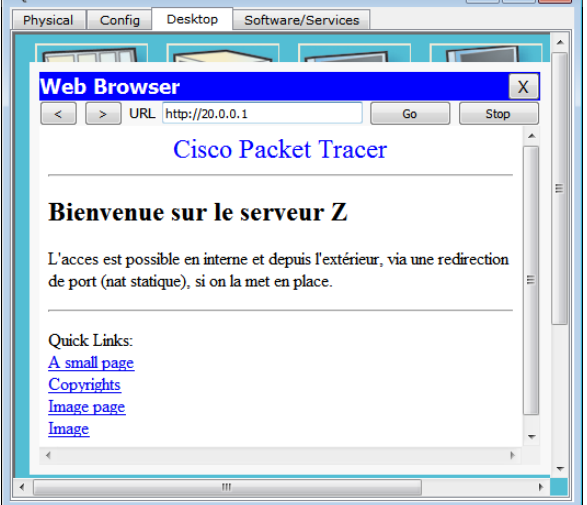
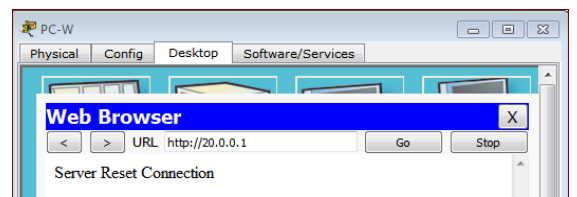
Mettre en place une redirection de port sur RtF, permettant d'atteindre Serv-Z depuis tout le réseau et notamment PC-W, en invoquant l'adresse publique de RtF.

TRAVAIL À FAIRE

7. Mettre en place la redirection nécessaire pour atteindre Serv-Z depuis l'extérieur du site Z.
8. Vérifier ensuite que l'accès HTTP fonctionne, par exemple depuis PC-W,

NB : Vous utilisez bien sûr le NAT statique comme indiqué plus haut, pour le port n° 80.

Accès Web depuis PC-W « AVANT / APRÈS »

	
	<p>Avant la mise en place de la redirection cela ne fonctionne pas, qu'on utilise l'adresse privée ou l'adresse publique.</p> <p>Après la mise en place de la redirection, on a bien accès à la page d'accueil sur Serv-Z, en utilisant bien entendu l'adresse publique du routeur.</p>

Éléments de correction commentés

Les première et deuxième étapes sont élémentaires.

Troisième étape

3. Mise en place d'un NAT dynamique sur les routeurs terminaux RtA et RtF
NB : Utiliser la solution 1 puisque les réseaux inter-routeurs sont en préfixe /30.

Exemple sur le Routeur RtA.

```
Routeur-A#conf term
```

Définition des interfaces INSIDE et OUTSIDE

```
Routeur-A(config)#interface fa0/1
Routeur-A(config-if)#ip nat inside
Routeur-A(config-if)#exit
Routeur-A(config)#interface fa0/0
Routeur-A(config-if)#ip nat outside
Routeur-A(config-if)#exit
```

Autorisation du réseau interne à sortir par translation

```
Routeur-A(config)#access-list 1 permit 192.168.0.0 0.0.0.255
```

Définition de l'interface sur laquelle s'effectue la translation

```
Routeur-A(config)#ip nat inside source list 1 interface fa0/0 overload
Routeur-A(config)#
```

Affichage de la table des translations suite à une commande PING 172.12.0.1 depuis PCW

```
Routeur-A#sh ip nat translations
Pro  Inside global      Inside local      Outside local     Outside global
icmp 20.6.6.2:1         192.168.0.1:1    172.12.0.1:1     172.12.0.1:1
icmp 20.6.6.2:2         192.168.0.1:2    172.12.0.1:2     172.12.0.1:2
icmp 20.6.6.2:3         192.168.0.1:3    172.12.0.1:3     172.12.0.1:3
icmp 20.6.6.2:4         192.168.0.1:4    172.12.0.1:4     172.12.0.1:4
icmp 20.6.6.2:5         192.168.0.1:5    172.12.0.1:5     172.12.0.1:5
icmp 20.6.6.2:6         192.168.0.1:6    172.12.0.1:6     172.12.0.1:6
icmp 20.6.6.2:7         192.168.0.1:7    172.12.0.1:7     172.12.0.1:7
icmp 20.6.6.2:8         192.168.0.1:8    172.12.0.1:8     172.12.0.1:8
```

Quatrième étape

4. Vous devez donc, pour chaque routeur de liaison RtB, RtC, RtD et RtE activer le protocole RIPv2 et déclarer les réseaux qui lui sont directement connectés.
5. À l'issue de cette étape, vous devez vérifier que chaque routeur connaît la route vers tous réseaux de liaison « 20.x.x.x » et les réseaux de serveurs « 172.x.x.x ».

NB : Comme indiqué précédemment les réseaux privés (192.168.0.0/24) ne seront pas déclarés au niveau de RIP dans notre cas de figure

Exemple sur le routeur RtB à 5 routes connectées

Passage en routage RIP Version 2

```
Routeur-B#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Routeur-B(config)#router rip
Routeur-B(config-router)#version 2
```

Déclaration des routes connectées

```
Routeur-B(config-router)#network 20.6.6.0
Routeur-B(config-router)#network 172.11.0.0
Routeur-B(config-router)#network 20.5.5.0
Routeur-B(config-router)#network 20.2.2.0
Routeur-B(config-router)#network 20.4.4.0
Routeur-B(config-router)#exit
```

Lorsque les déclarations sont faites sur chacun des routeurs RtB à RtE, un affichage de la table de routage doit donner (par exemple sur le routeur RtB) :

```
Routeur-B#sh ip route
Codes: C - connected, S - static, I - IGRP, 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, E - EGP
i - IS-IS, 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
```

```
Gateway of last resort is not set
```

```
20.0.0.0/30 is subnetted, 7 subnets
R 20.0.0.0 [120/1] via 20.2.2.1, 00:00:22, FastEthernet0/1
R 20.1.1.0 [120/1] via 20.2.2.1, 00:00:22, FastEthernet0/1
[120/1] via 20.5.5.1, 00:00:22, Ethernet1/0
C 20.2.2.0 is directly connected, FastEthernet0/1
R 20.3.3.0 [120/1] via 20.4.4.1, 00:00:22, Ethernet1/1
[120/1] via 20.2.2.1, 00:00:22, FastEthernet0/1
C 20.4.4.0 is directly connected, Ethernet1/1
C 20.5.5.0 is directly connected, Ethernet1/0
C 20.6.6.0 is directly connected, FastEthernet0/0
172.11.0.0/24 is subnetted, 1 subnets
C 172.11.0.0 is directly connected, Ethernet1/2
172.12.0.0/24 is subnetted, 1 subnets
R 172.12.0.0 [120/1] via 20.2.2.1, 00:00:22, FastEthernet0/1
```

On retrouve donc les routes connectées (lignes « C »)

- 20.2.2.0
- 20.4.4.0
- 20.5.5.0
- 20.6.6.0
- 172.11.0.0

On retrouve aussi les routes connectées aux routeurs voisins et « rippées » (lignes « R »)

- 20.0.0.0
- 20.1.1.0
- 20.3.3.0
- 172.12.0.0

Une commande DEBUG IP RIP permet de voir les (nombreux) échanges avec les routeurs voisins

```
Routeur-B#debug ip rip
RIP protocol debugging is on
Routeur-B#RIP: received v2 update from 20.5.5.1 on Ethernet1/0
    20.0.0.0/30 via 0.0.0.0 in 2 hops
    20.1.1.0/30 via 0.0.0.0 in 1 hops
    20.3.3.0/30 via 0.0.0.0 in 2 hops
    172.12.0.0/24 via 0.0.0.0 in 2 hops
RIP: sending v2 update to 224.0.0.9 via FastEthernet0/1 (20.2.2.2)
RIP: build update entries
    20.4.4.0/30 via 0.0.0.0, metric 1, tag 0
    20.5.5.0/30 via 0.0.0.0, metric 1, tag 0
    20.6.6.0/30 via 0.0.0.0, metric 1, tag 0
    172.11.0.0/24 via 0.0.0.0, metric 1, tag 0
RIP: sending v2 update to 224.0.0.9 via Ethernet1/1 (20.4.4.2)
RIP: build update entries
    20.0.0.0/30 via 0.0.0.0, metric 2, tag 0
    20.1.1.0/30 via 0.0.0.0, metric 2, tag 0
    20.2.2.0/30 via 0.0.0.0, metric 1, tag 0
    20.5.5.0/30 via 0.0.0.0, metric 1, tag 0
    20.6.6.0/30 via 0.0.0.0, metric 1, tag 0
    172.11.0.0/24 via 0.0.0.0, metric 1, tag 0
    172.12.0.0/24 via 0.0.0.0, metric 2, tag 0
RIP: sending v2 update to 224.0.0.9 via Ethernet1/0 (20.5.5.2)
RIP: build update entries
    20.0.0.0/30 via 0.0.0.0, metric 2, tag 0...
```

Cinquième étape

6. À l'aide de commandes (PING, TRACERT), vous mettrez en évidence le comportement du protocole RIP et la tolérance de panne.

- Sur le poste PC-W, passer la commande ping -t 172.12.0.1
- Couper par exemple l'interface Fa0/1 (20.2.2.1) du routeur RtE
- Observer les résultats du PING et mesurer le temps de rétablissement de la liaison entre le PC et le serveur Web Serv-Y du site Y via une route détournée.
- Remettre en fonctionnement l'interface Fa0/1 du routeur RtE
- Sur le PC-W, passer une commande tracert 172.12.0.1

Tracert 172.12.0.1

Tracing route to 172.12.0.1 over a maximum of 30 hops:

1	1 ms	0 ms	0 ms	192.168.0.254
2	0 ms	0 ms	0 ms	20.6.6.1
3	0 ms	0 ms	0 ms	20.2.2.1
4	*	0 ms	0 ms	172.12.0.1

Trace complete.

- On voit qu'on passe en ligne directe : PC-W → RtB → RtE → Serv-Y
- On coupe de nouveau l'interface Fa0/1 du routeur RtE, et on exécute de nouveau le même TRACERT

Tracert 172.12.0.1

Tracing route to 172.12.0.1 over a maximum of 30 hops:

1	0 ms	0 ms	0 ms	192.168.0.254
2	0 ms	0 ms	0 ms	20.6.6.1
3	0 ms	0 ms	0 ms	20.5.5.1
4	0 ms	0 ms	0 ms	20.1.1.1
5	0 ms	1 ms	1 ms	172.12.0.1

Trace complete.

- On voit qu'on passe maintenant par une route détournée : PC-W → RtB → RtD → RtE → Serv-Y

Sixième étape

7. Mettre en place la redirection nécessaire pour atteindre Serv-Z depuis l'extérieur du site Z.
8. Vérifier ensuite que l'accès HTTP fonctionne, par exemple depuis PC-W,

```
Routeur-F(config)#nat inside source static tcp 192.168.0.200 80 20.0.0.1 80
```

Après une consultation du site web d'adresse 20.0.0.1 à partir du PC0, on peut afficher la table des translations d'adresses du routeur F.

```
Routeur-F#show ip nat translations
Pro Inside global Inside local Outside local Outside global
tcp 20.0.0.1:80 192.168.0.200:80 --- ---
tcp 20.0.0.1:80 192.168.0.200:80 20.6.6.2:1034 20.6.6.2:1034
```

On voit ici que la demande émane de l'adresse 20.6.6.2, c'est-à-dire l'adresse extérieure du routeur RtA qui fait la demande pour le client PC-W.