Commandes complémentaires des routeurs Cisco

I Présentation

Les routeurs Cisco permettent bien sûr d'interconnecter des réseaux entre eux.

L'aspect configuration des interfaces et du routage a été vu dans un autre document. Nous allons aborder d'autres fonctionnalités des routeurs Cisco.

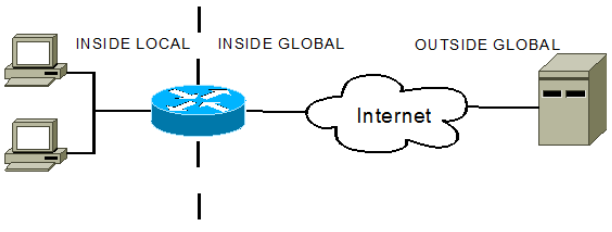
Les fonctionnalités sont :

- ✗ Mise en place du NAT pour connecter un réseau privé sur un réseau public ;
- ✗ Mise en place d'un serveur DHCP ;
- ✗ Utilisation du protocole SNMP pour superviser à distance les routeurs ;
- ✗ Mise en place d'un tunnel virtuel entre deux réseaux privés (VPN).

II Mise en place du NAT

Le NAT (Network Address Translation) permet de configurer des réseaux privés avec n adresses. Ces réseaux sont vus sur le réseau public (Internet par exemple) comme une ou quelques adresses publiques.

Cisco définit le réseau interne (privé) et le réseau externe (public) : le réseau utilisant les adresses IP privées est appelé le réseau interne (inside), tandis que la partie du réseau utilisant des adresses IP publiques (Internet) est appelé le réseau externe (outside).



Il faut préciser quel que soit le type de translation, les deux réseaux :

1. Pour l'interface interne :

```
ip nat inside
```

2. Pour l'interface externe :

```
ip nat outside
```

Exemple : fa0/0 est connecté sur le réseau externe (Internet) et fa1/0 est connecté au réseau interne (local)

```
Router#configure terminal
Router(config)#interface fa1/0                fa1/0 réseau privé
Router(config-if)#ip nat inside
Router(config-if)#interface fa0/0             fa0/0 réseau public
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#exit
```

II.1 Configuration NAT statique

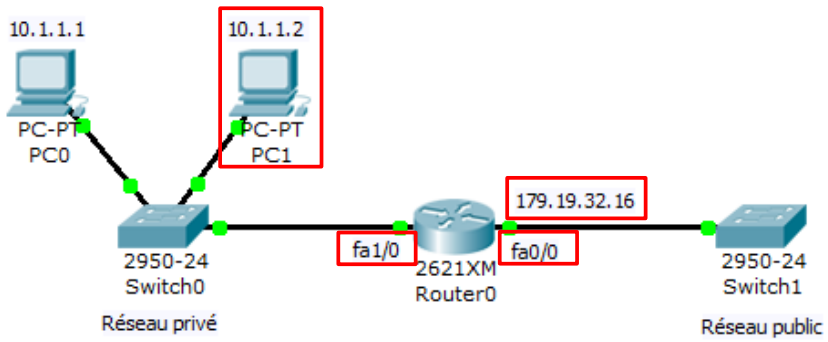
Dans cette configuration, l'adresse IP locale donnée sera translatée en une adresse globale. Il faut pour cela effectuer les deux actions suivantes :

1. Définir la traduction statique d'adresses :

```
ip nat inside source static <ip local> <ip global>
```

2. Déclarer les deux interfaces.

Exemple : L'adresse 10.1.1.2 du réseau interne (connecté à l'interface fa1/0) sera vue du réseau externe comme l'adresse 179.19.32.16.



L'ordinateur PC0 ne pourra pas sortir du réseau contrairement au PC1.

```
Router#configure terminal
Router(config)#ip nat inside source static 10.1.1.2  179.19.32.16
Router(config)#interface fa1/0
Router(config-if)#ip nat inside
Router(config-if)#interface fa0/0
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#exit
```

II.2 Configuration NAT dynamique

Cette configuration définit un certain nombre d'adresses (pool) du côté réseau global. Ces adresses pourront être utilisées pour la translation. Si par exemple, le pool est de 3 adresses, 3 ordinateurs du réseau interne pourront communiquer avec le réseau global en même temps. Le choix de ces trois adresses sera chronologique. Le premier arrivé est le premier servi. Pour cette configuration, les 4 étapes suivantes sont à réaliser :

1. Définir le ou les pool(s) d'adresses :

```
ip nat pool <name> <ip début> <ip fin> netmask <mask>
```

- ✗ **<name>** : nom du pool d'adresses utilisé lors des commandes suivantes
- ✗ **<ip début> <ip fin>** : première et dernière adresses à utiliser pour la translation dans le réseau externe

- **<mask>** : masque réseau pour le réseau externe

2. Définir une liste d'accès standard qui indiquera quelles adresses doivent être traduites :

```
access-list <numéro> permit <source> <wildcard_mask>
```

- **<numéro>** : numéro de la liste
- **<source>** : réseau interne
- **<wildcard_mask>** : masque où les 1 sont remplacés par des 0 et les 0 par des 1. Exemple : pour un réseau de classe A, le masque est 255.0.0.0, le wildcard_mask est 0.255.255.255.

3. Établir une traduction dynamique de la source en spécifiant l'access-list :

```
ip nat inside source list <numéro> pool <name>
```

- **<numéro>** : numéro de la liste ACL
- **<name>** : nom du pool créé précédemment.

4. Déclarer les deux interfaces.

Exemple : les adresses de 179.9.8.17 à 179.9.8.30 (avec un masque réseau 255.255.255.0) sont utilisées pour translater tout le réseau 10.0.0.0/8.

```
Router#configure terminal
Router(config)#ip nat pool my-internet 179.9.8.17 179.9.8.30 netmask 255.255.255.0
Router(config)#access-list 1 permit 10.0.0.0 0.255.255.255
Router(config)#ip nat inside source list 1 pool my-internet
Router(config)#interface fa1/0
Router(config-if)#ip nat inside
Router(config-if)#interface fa0/0
Router(config-if)#ip nat outside
Router(config)#exit
```

II.3 Configuration PAT avec une seule adresse globale

Dans cette configuration, toutes les adresses internes peuvent être translatées car ce ne sont pas les adresses qui sont transformées mais les ports. Les 3 étapes à réaliser sont :

1. Définir une liste d'accès standard pour déterminer quelles adresses doivent être traduites :

```
access-list <numéro> permit <source> <wildcard_mask>
```

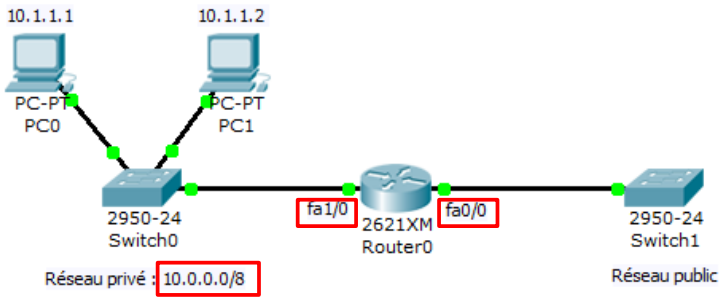
- **<numéro>** est un numéro inférieur à 100, 1 par exemple
- **<source>** est l'adresse de sous-réseau du réseau privé ;
- **<wildcard_mask>** est son masque de sous-réseau inversé.

2. Établir une traduction dynamique de la source en spécifiant l'access-list :

```
ip nat inside source list <numéro> interface <interface> overload
```

3. Déclarer les deux interfaces.

Exemple :




```
Router#configure terminal
Router(config)#interface fa1/0                    fa1/0 : interface réseau privé
```

```
Router(config-if)#ip nat inside
Router(config-if)#interface fa0/0              fa0/0 : interface réseau publique
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#access-list 1 permit  10.0.0.0  0.255.255.255  10.0.0.0/8 : réseau privé
Router(config)#ip nat inside source list 1 interface fa0/0 overload     fa0/0 : interface
sortante (publique)
Router(config)#exit
```

II.4 Configuration PAT avec un pool d'adresses publiques

La configuration PAT permet de translater les adresses en modifiant non pas les adresses IP mais les ports utilisés. Les 4 étapes sont :

1. Définir une liste d'accès standard pour déterminer quelles adresses doivent être traduites :

```
access-list <numéro> permit <source> <wildcard>
```

2. Définir le ou les pool(s) d'adresses pour les adresses du réseau global:

```
ip nat pool <name> <ip début> <ip fin> netmask <mask>
```

3. Établir une traduction dynamique de la source en spécifiant l'access-list :

```
ip nat inside source list <numéro> pool <name> overload
```

4. Déclarer les deux interfaces.

Exemple : Le réseau interne est 10.0.0.0/8 et l'adresse 10.0.0.1 ne doit pas être translatée. Les adresses sur le réseau externe sont de 179.9.8.17 à 179.9.8.30 avec un masque réseau 255.255.255.0.

```
Router#configure terminal
Router(config)#access-list 1 permit 10.0.0.0  0.255.255.255
Router(config)#access-list 1 deny 10.0.0.1
Router(config)#ip nat pool my-internet 179.9.8.17 179.9.8.30 netmask 255.255.255.0
Router(config)#ip nat inside source list 1 pool my-internet overload
Router(config)#interface fa1/0                 fa1/0 : interface réseau privé
Router(config-if)#ip nat inside
Router(config-if)#interface fa0/0              fa0/0 : interface réseau publique
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#exit
```

II.5 Autres commandes

- **show ip nat translations :** affiche des informations sur chaque translation en cours en particulier le temps depuis lequel elle est active.
- **clear ip nat translation <entrées> :** efface la table de translations (* pour toutes les adresses)
- **ip nat translation timeout <time_out seconds> :** configure la durée de validité d'une traduction

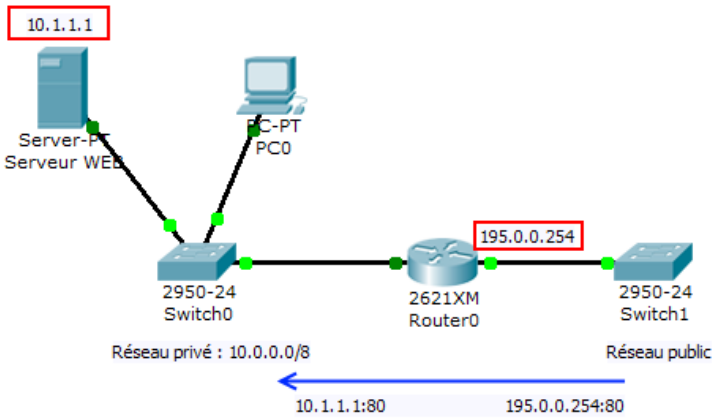
II.6 Translation de port

Toute connexion du réseau public (outside) vers le réseau privé (inside) est impossible. Or il est intéressant parfois de permettre des accès à un serveur interne à partir d'Internet, par exemple un serveur web. Pour cela, il faut faire une translation de port (ou port forwarding). Tout accès à l'adresse publique avec un port spécifique (port TCP ou UDP) sera renvoyé vers l'adresse privée avec le numéro de port indiqué.

```
ip nat inside source static { tcp | udp } <localaddr> <localport> <globaladdr>
<globalport>
```

a Translation simple

Exemple : Les paquets arrivants depuis l'extérieur à l'adresse 195.0.0.254 sur le port 80 seront envoyés au port 80 de la machine 10.1.1.1 (réseau privé).




```
Router#configure terminal
Router(config)#ip nat inside source static tcp  10.1.1.1  80  195.0.0.254  80
Router(config)#exit
```

Remarque : pour des ports UDP, il suffit de remplacer tcp par udp.

b Translation avec translation de port

Si votre routeur possède un serveur WEB intégré et que vous souhaitez y accéder aussi à partir d'Internet, vous pouvez effectuer les accès suivants :

- ✗ Serveur WEB du routeur : 195.0.0.254 avec le port 80 (accès normal à partir d'un navigateur WEB) ;
- ✗ Serveur WEB interne (10.1.1.1) : 195.0.0.254 avec le port 8080 (ajout de « :8080 » dans l'URL du navigateur WEB).

Dans ce cas, vous remplacez la commande ci-dessus par :

```
Router(config)#ip nat inside source static tcp  10.1.1.1  80  195.0.0.254  8080
```

c Translation sur plusieurs ports

Si vous devez rediriger un ensemble de ports vers une seule adresse IP, il faut préciser les ports grâce au mot **range**.

Exemple : Les paquets TCP arrivants depuis l'extérieur pour les ports de 1024 à 2048 sont envoyés à la machine 10.0.0.1 (le port est inchangé), un pool myPool et une nouvelle liste d'accès (100) par exemple sont créés.

```
Router#configure terminal
Router(config)#ip nat pool myPool 10.0.0.1  10.0.0.1 netmask 255.255.255.0 type rotary
Router(config)#ip nat inside source list 100 pool myPool
Router(config)#access-list 100 permit tcp any any range 1024 2048
outer(config)#exit
```

III Configuration pour plusieurs VLANs

Supposons que vos commutateurs ont été configurés pour avoir plusieurs VLAN (réseaux virtuels). Les systèmes connectés sur ces VLANs peuvent communiquer entre eux. Mais il faut utiliser un routeur pour réaliser le routage entre ces réseaux virtuels.

Vous devez donc configurer le routeur pour interconnecter différents VLANs. Il faut pour chaque VLAN sur le commutateur, créer une sous interface. Par exemple, pour l'interface fa0/0, on crée des sous-interfaces qui s'appelleront fa0/0.1, fa0/0.2, ...

Ces sous-interfaces doivent être configurées grâce à la commande :

```
encapsulation dot1q 20
```

où 20 est le numéro du VLAN.

Exemple : les 2 VLANs sont connectés sur le port fa0/0 ; l'adresse IP sur le premier VLAN 10 est 160.54.0.254 et celle sur le second VLAN 20 est 160.54.1.254 (masque des 2 réseaux : 255.255.255.0)

```
Router#configure terminal
Router(config)#int fa0/0                              fa0/0 : interface du trunk
Router(config-if)#no shutdown
Router(config-if)#int fa0/0.1                         fa0/0.1 : 1ère interface
Router(config-subif)#encapsulation  dot1q  10         pour le vlan 10
Router(config-subif)#ip  address  160.54.0.254  255.255.255.0     a pour adresse
160.54.0.254/24
Router(config-subif)#exit
Router(config)#int fa0/0.2                            fa0/0.2 : 2nde interface
Router(config-subif)#encapsulation  dot1q  20         pour le vlan 20
Router(config-subif)#ip  address  160.54.1.254  255.255.255.0     a pour adresse
160.54.1.254/24
Router(config-subif)#exit
Router(config)#exit
```

Pour visualiser le résultat :

- ✗ Pour vérifier la configuration des interfaces :

```
Router# show ip route
```

- ✗ Pour vérifier l'association du VLAN et de l'interface :

```
Router# show interfaces
```

IV Configuration DHCP

On utilise souvent dans chaque réseau, un serveur DHCP qui permet d'affecter une adresse IP à chaque ordinateur du réseau et cela de façon dynamique. Il peut être intéressant de ne pas ajouter un serveur spécifiquement pour cela, mais d'utiliser le routeur déjà présent dans le réseau.

Pour configurer un routeur pour qu'il soit un serveur DHCP, il faut :

1. Créer un pool DHCP sur le routeur :

```
ip dhcp pool <nom_du_pool>
```

2. Spécifier le réseau à écouter avec la commande :

```
network <adresse_ip> <masque>
```

3. Définir les options du DHCP :
   - ✗ Passerelle par défaut :

```
default-router <ip_du_router>
```

   - ✗ Serveur DNS primaire :

```
dns-server <ip_du_serveur_DNS>
```

   - ✗ Durée du bail :

```
lease <jour> <heure> <minute>
```

   - ✗ Nom de domaine associé à votre configuration DHCP :

```
domain-name <nom du domaine>
```

   - ✗ Nom du serveur Wins :

```
netbios-name-server <nom du serveur Wins>
```

4. Définir quelles adresses ne seront pas distribuées par le serveur, pour éviter les conflits en utilisant la commande en mode de configuration globale :

```
ip dhcp excluded-address <ip_interdite>
```

ou

```
ip dhcp excluded-address <première adresse ip interdite> <dernière adresse ip interdite>
```

Exemple : On crée un pool pour le réseau Labo, réseau 172.16.128.0/17.

```
Router#configure terminal
Router(config)#ip dhcp pool Labo
Router(dhcp-config)#network 172.16.128.0 255.255.128.0   Réseau concerné : 172.16.128.0/17
Router(dhcp-config)#default-router 172.16.255.254        Passerelle par défaut :
172.16.255.254
Router(dhcp-config)#dns-server 172.16.0.250              Serveur DNS : 172.16.0.250
Router(dhcp-config)#exit
Router(config)#ip dhcp excluded-address 172.16.255.254   Adresse 172.16.255.254 non
attribuée par le serveur
Router(config)#lease 0 5 30                              Durée du bail 5h30
Router(config)#exit
```

Pour voir les baux distribués par le serveur, il faut taper la commande « show ip dhcp binding » en mode privilégié.

Exemple :

```
Router#show ip dhcp binding
IP address       Client-ID/              Lease expiration        Type
                 Hardware address
192.168.20.1     0009.7CE7.8A94           --                     Automatic
```

V Configuration DNS

Vous pouvez configurer votre routeur pour que vous puissiez exécuter des commandes ping en mettant le nom des machines au lieu de leur adresse IP.

La première étape est d'activer l'utilisation d'un DNS :

```
routeur-dns# configure terminal
routeur-dns(config)# ip domain-lookup
```

Vous devez ensuite configurer les paramètres du serveur, c'est-à-dire le domaine par défaut :

```
routeur-dns(config)# ip domain-name snir.lan
```

Puis il faut définir les noms d'hôtes de votre réseau local :

```
routeur-dns(config)# ip host routeur-dns.snir.lan 192.168.0.254
routeur-dns(config)# ip host serveur.snir.lan 192.168.0.253
```

Pour les autres adresses, vous pouvez définir le serveur DNS à utiliser :

```
routeur-dns(config)# ip name-server 192.168.0.253
```

VI Mise en place du protocole SNMP

Le protocole SNMP permet de superviser les routeurs Cisco grâce à des logiciels de supervision. Il faut activer le protocole SNMP sur ceux-ci. En effet, le protocole SNMP permet notamment de récupérer des informations statistiques sur les équipements réseaux.

La commande de base est :

```
snmp-server community string [view view-name] [ro | rw] [access-list-number]
```

Elle configure la communauté string.

ro | rw : lecture seule ou lecture/écriture

Exemple :

```
Router#configure terminal
Router(config)#snmp-server community essai RO            Nom de la communauté en lecture
Router(config)#snmp-server community essaiRoot RW        Nom de la communauté en lecture/
écriture
Router(config)#end
```

La communauté essai permet de lire les informations sur le routeur, la communauté essaiRoot de les lire et de les modifier aussi.

Ensuite après avoir configuré ces deux communautés dans un « MIB browser », vous pourrez lire ou modifier des informations telles que le nom du routeur, la configuration des interfaces, …

VII Mise en place du VPN

VII.1 Présentation

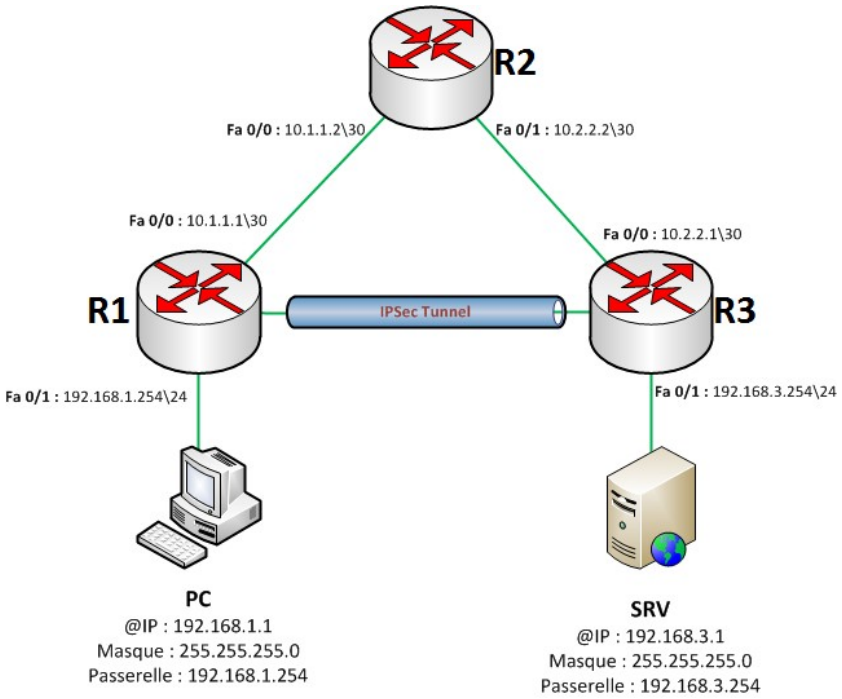
VPN est l'acronyme de « Virtual Private Network » soit réseau privé virtuel.

Par exemple, cela va nous permettre depuis notre domicile, d'avoir accès au réseau local d'un autre site distant (notre entreprise par exemple) à travers une connexion internet sécurisée (IpSec). On parle alors de VPN Remote-Access.

Le VPN va aussi nous permettre de relier deux sites, par exemple, une grande entreprise a deux locaux situés à deux endroits bien distinct, à l'instar des lignes dédiées où l'on devait passer par un opérateur ou encore au lieu de mettre en place des liens physiques entre ces deux sites (fibre optique…), on va passer par internet pour relier ces deux sites.

On parle alors de VPN Site-To-Site comme dans l'exemple ci-dessous.

La protection des données transmises sur Internet est faite grâce à la mise en place de cryptage de celles-ci.



VII.2 Préparation initiale

Vous devez tout d'abord vérifier que l'IOS de vos routeurs supporte le VPN.

Ensuite vous devez configurer les routeurs en mettant en œuvre le NAT. Les étapes sont :

- ✗ Configuration de base des routeurs (noms, bannière et mots de passe) ;
- ✗ Configuration des interfaces réseau ;
- ✗ Configuration du routage (la route par défaut est celle par le routeur R2 dans l'exemple ci-dessus) ;
- ✗ Test de cette configuration (les routeurs peuvent « pinguer » les autres routeurs) ;
- ✗ Mise en œuvre du NAT entre les réseaux privés et les réseaux publics, NAT du type PAT avec une seule adresse globale ;
- ✗ Test du NAT (les ordinateurs peuvent « pinguer » les adresses du routeur R2).

Maintenant si tous les tests sont concluants, vos routeurs sont prêts pour entrer les commandes pour le VPN.

Attention : il faut absolument vérifier que les tests ont bien effectués et sont corrects avant de continuer.

Afin de faciliter votre travail et d'éviter des erreurs, il est recommandé de prendre en compte la configuration ci-dessus et celle dans votre cas. Ensuite, vous devez trouver la correspondance entre les adresses, les interfaces et les masques des deux configurations. Il suffit alors de remplacer dans l'exemple ci-dessous ces valeurs (valeurs indiquées en rouge) par celles de votre configuration.

VII.3 Mise en œuvre du VPN

La mise en place du VPN nécessite un certain nombre de commandes. Celles en gras seront à adapter en fonction du routeur sur lequel on lance les commandes. Elles seront à exécuter sur chaque routeur pour que le VPN fonctionne : le routeur local est celui sur lequel les commandes sont lancées, le routeur distant est l'autre.

L'exemple suivant vous aidera dans votre travail.

Vérifiez bien chaque commande avant de l'exécuter car il est facile de se tromper.

1. On active ensuite les fonctions crypto du routeur :

```
Router#configure terminal
Router(config)#crypto isakmp enable
```

2. Nous allons configurer les règles IKE entre les deux routeurs qui détermine quelle méthode de cryptage on utilise, quel Hash, quel type d'authentification, ...

```
Router(config)#crypto isakmp policy 1
Router(config-isakmp)#encr 3des
Router(config-isakmp)#hash md5
Router(config-isakmp)#authentication pre-share
Router(config-isakmp)#group 2
Router(config-isakmp)#lifetime 86400
Router(config-isakmp)#exit
```

group 2 : Spécifie l'échange de clés Diffie-Hellman

lifetime : Spécifie le temps de validité de la connexion avant une nouvelle négociation des clefs

3. Ensuite nous devons configurer la clef de cryptage pré-partagée :

```
Router(config)#crypto isakmp key motdepassevpn address ip1.ip2.ip3.ip4

mot de passe de la connexion VPN
                                        adresse publique du routeur distant
```

Sur certains routeurs avec certains IOS, la commande ne fonctionne pas car le routeur demande si le mot de passe doit être chiffré ou pas, tapez cette commande :

```
Router(config)#crypto isakmp key 6 motdepassevpn address ip1.ip2.ip3.ip4
```

4. Configurons le processus de transformation :

```
Router(config)#crypto ipsec transform-set TS esp-3des esp-md5-hmac
```

esp : Signifie Encapsulation Security Protocol

N'oubliez pas d'utiliser les mêmes protocoles d'encryption et de Hash utilisés dans la première étape.

Dans notre cas : *Encryption : 3des* et *hash : md5*

5. On fixe ensuite une valeur de Lifetime (durée de vie d'une session cryptée) :

```
Router(config)#crypto ipsec security-association lifetime seconds 1800
```

6. La 5éme étape consiste à créer une ACL qui va sélectionner le trafic à crypter.

```
Router(config)#ip access-list extended VPN-TRAFFIC
Router(config-ext-nacl)#permit ip  i.i.i.i m.m.m.m   ii.ii.ii.ii mm.mm.mm.mm

Adresse du routeur local et son masque inversé côté réseau privé
                       Adresse du routeur distant et son masque inversé côté réseau privé
Router(config-ext-nacl)#exit
```

7. Dans cette dernière étape, nous configurons la crypto map qui va associer l'access-list, le trafic, et la destination dans une « carte de cryptage appelée CMAP » :

```
Router(config)#crypto map CMAP 10 ipsec-isakmp
Router(config-crypto-map)#set peer ip1.ip2.ip3.ip4    Adresse publique du routeur distant

Router(config-crypto-map)#set transform-set TS
Router(config-crypto-map)#match address VPN-TRAFFIC
Router(config-crypto-map)#exit
```

8. La configuration de R1 est presque terminée nous devons appliquer la carte de cryptage sur l'interface de sortie :

```
Router(config)#interface NomDeLInterfaceSortante   Nom de l'interface publique du routeur
distant
Router(config-if)#crypto map CMAP
```

Un message vous indique que la crypto map fonctionne.

9. Il ne reste plus qu'à activer le NAT si nécessaire avec suppression de l'éventuel NAT précédent. Cette règle permet de rejeter le trafic destiné au VPN dans les règles du NAT.

```
Router(config)#no access-list 1
Router(config)#access-list 100 deny ip  i.i.i.i m.m.m.m  ii.ii.ii.ii mm.mm.mm.mm

Adresse du routeur local et son masque inversé côté réseau privé
                      Adresse du routeur distant et son masque inversé côté réseau privé

Router(config)#access-list 100 permit ip  i.i.i.i m.m.m.m  any
Router(config)#no ip nat inside source list 1
Router(config)#ip nat inside source list 100 interface NomdeLInterfaceSortante overload
Router(config)#exit
```

Exemple pour le schéma ci-dessus (routeur 1) :

```
Router(config)#crypto isakmp enable
Router(config)#crypto isakmp policy 1
Router(config-isakmp)#encr 3des
Router(config-isakmp)#hash md5
Router(config-isakmp)#authentication pre-share
Router(config-isakmp)#group 2
Router(config-isakmp)#lifetime 86400
Router(config-isakmp)#exit
Router(config)#crypto isakmp key motDePasse address 10.2.2.1
Router(config)#crypto ipsec transform-set TS esp-3des esp-md5-hmac
Router(cfg-crypto-trans)#exit
Router(config)#crypto ipsec security-association lifetime seconds 1800
Router(config)#ip access-list extended VPN-TRAFFIC
Router(config-ext-nacl)#permit ip 192.168.1.254  0.0.0.255  192.168.3.254  0.0.0.255
Router(config-ext-nacl)#exit
Router(config)#crypto map CMAP 10 ipsec-isakmp
Router(config-crypto-map)#set peer 10.2.2.1
Router(config-crypto-map)#set transform-set TS
Router(config-crypto-map)#match address VPN-TRAFFIC
Router(config-crypto-map)#exit
Router(config)#interface fa0/0
Router(config-if)#crypto map CMAP
```

```
Router(config)#no access-list 1     Si une access-list a été configurée pour le NAT
Router(config)#access-list 100 deny ip 192.168.1.254  0.0.0.255  192.168.3.254  0.0.0.255
Router(config)#access-list 100 permit ip 192.168.1.254  0.0.0.255 any
Router(config)#no ip nat inside source list 1 interface fa0/0
Router(config)#ip nat inside source list 100 interface fa0/0 overload
Router(config)#exit
```

Après configuration des deux routeurs, si vous n'avez pas fait d'erreur, vous pouvez communiquer entre les ordinateurs des deux réseaux privés.