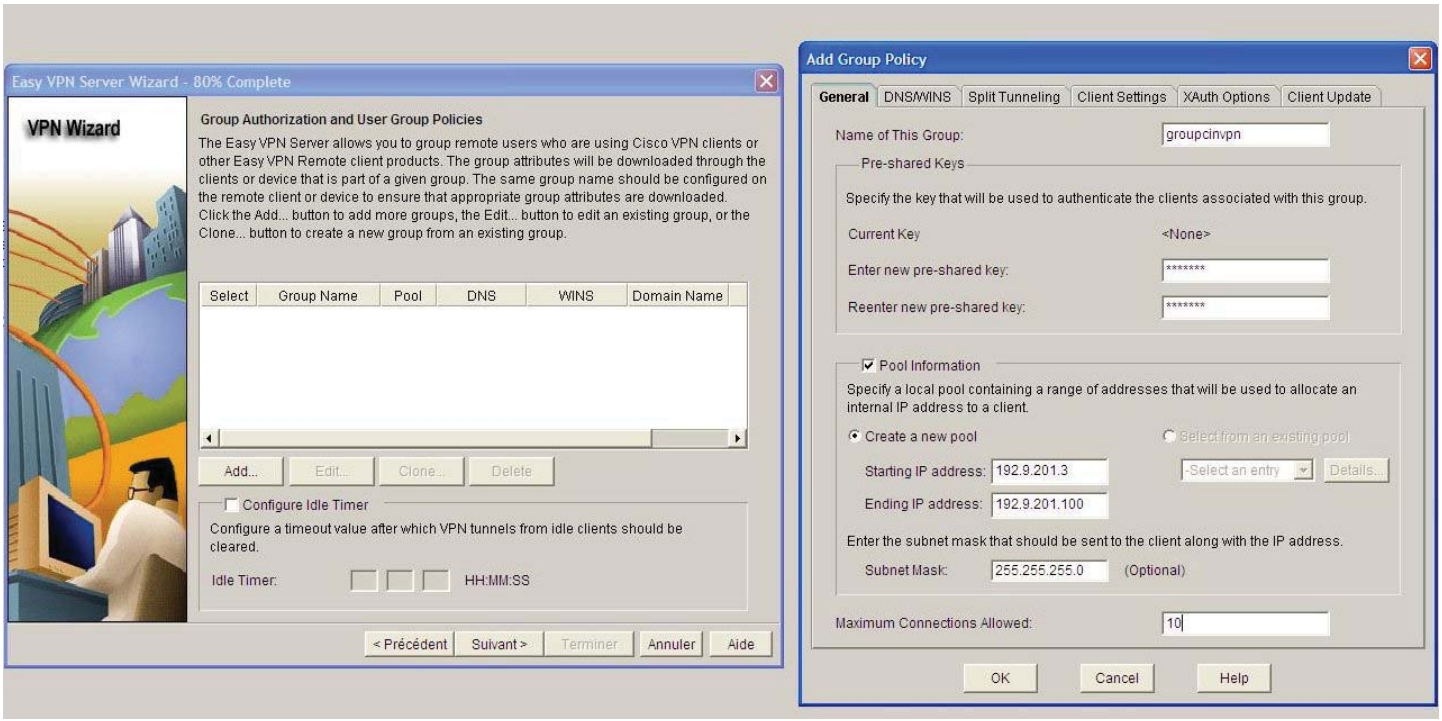
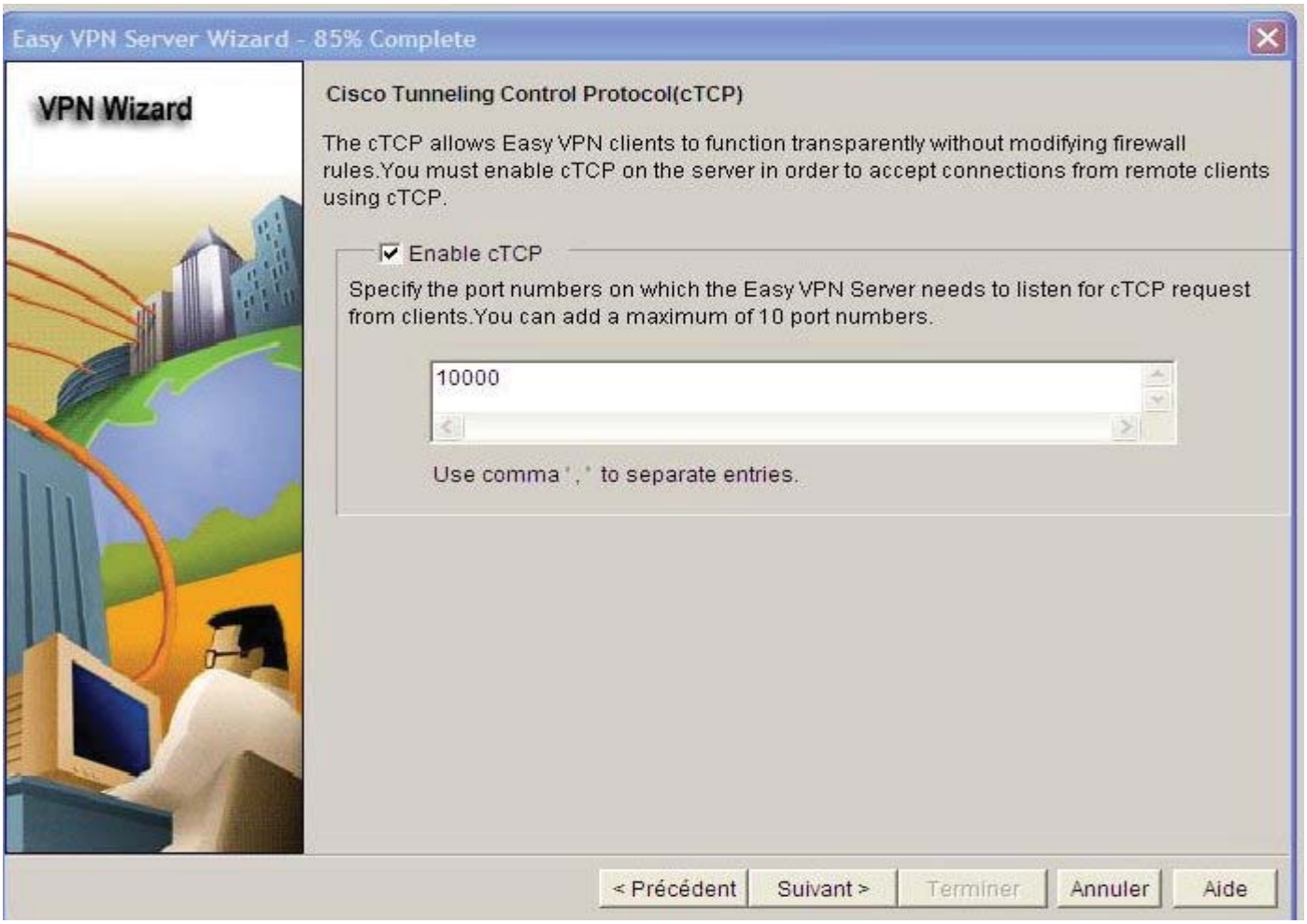
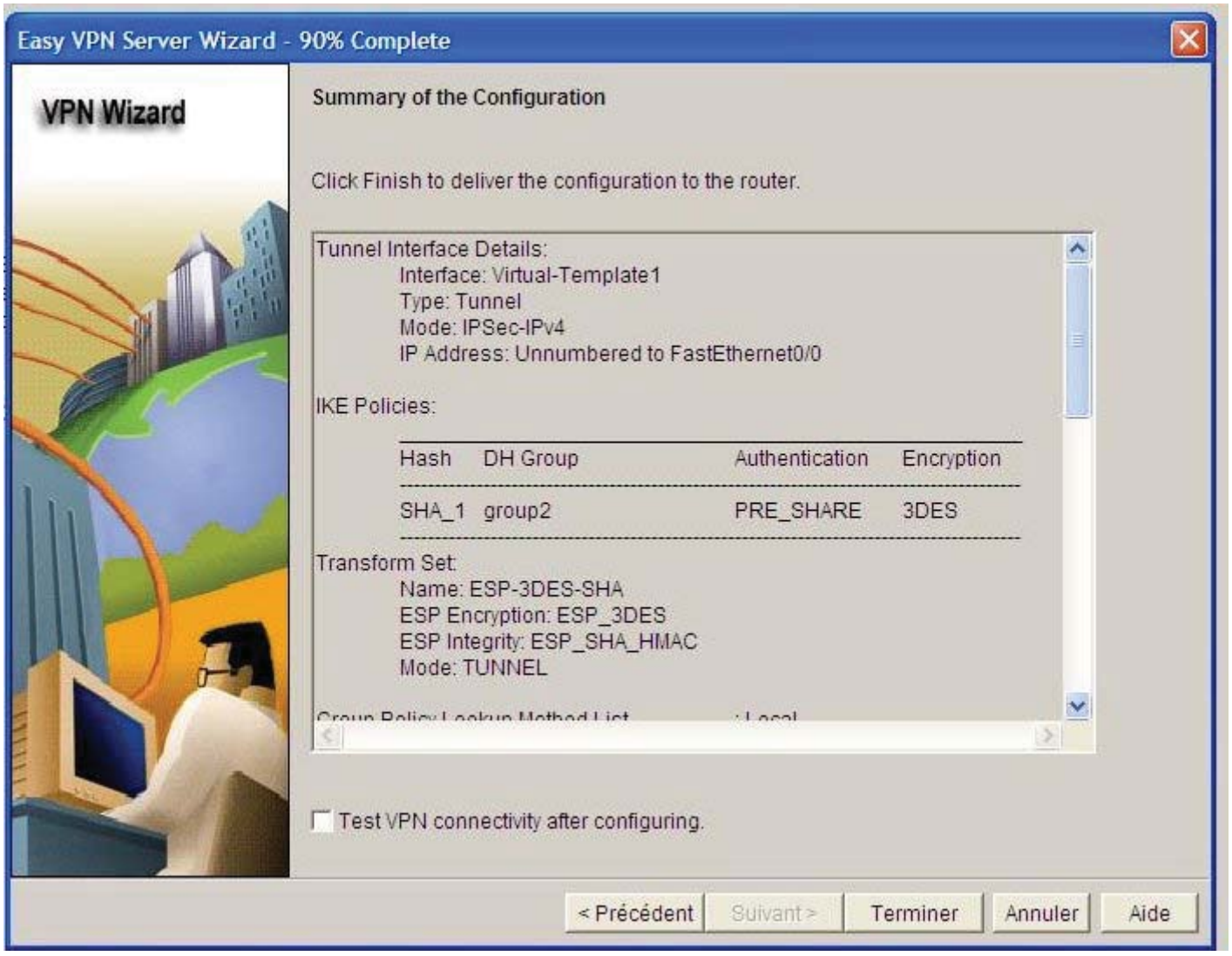
Créez un compte pour un groupe d'utilisateurs groupcinvpn ayant pour mot de passe password cisco1*



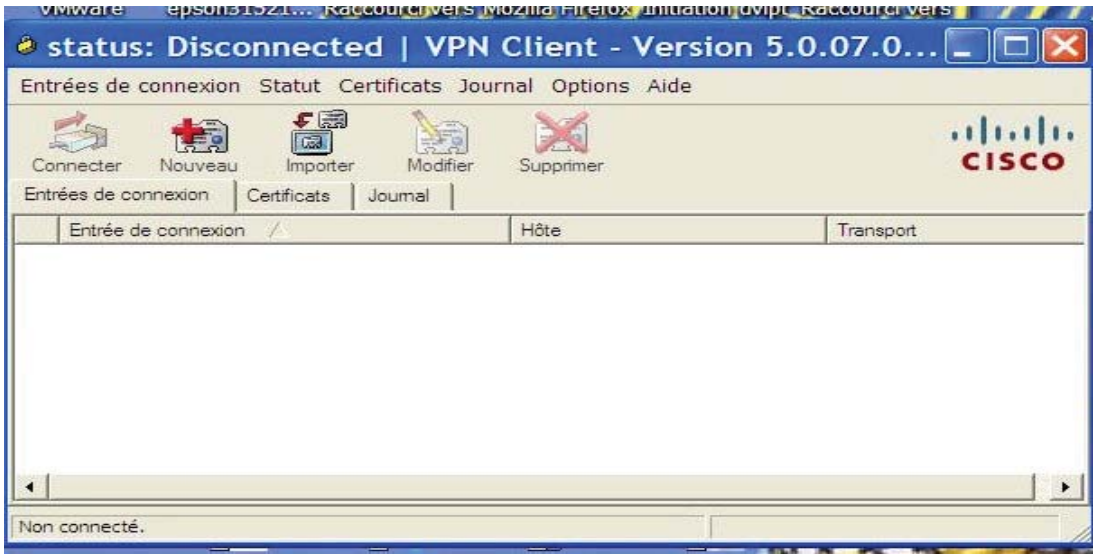


Gardez le port d'écoute TCP pour les clients qui se connectent

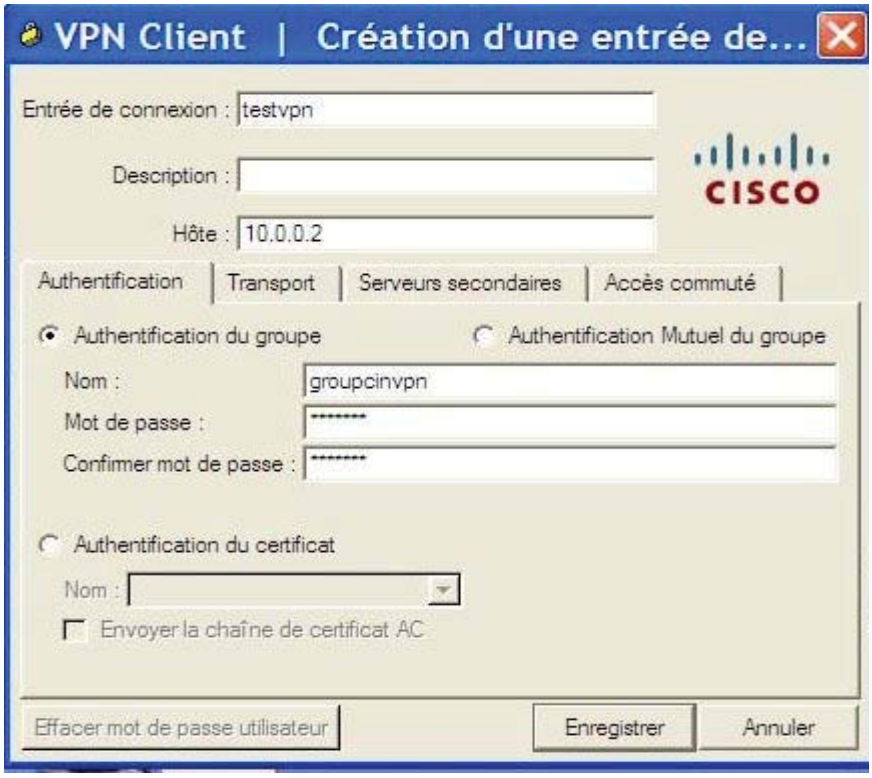
La page suivante nous donne la configuration adoptée pour cette liaison VPN :



Du côté du client vpn lancez le logiciel clientvpn

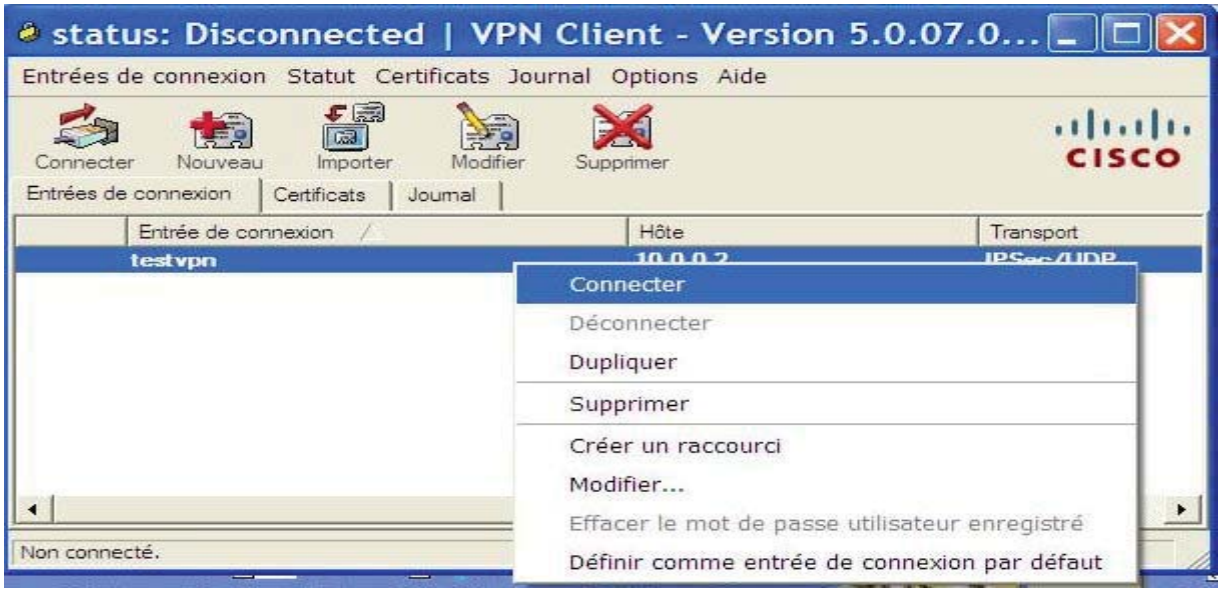


Puis cliquez sur nouveau :



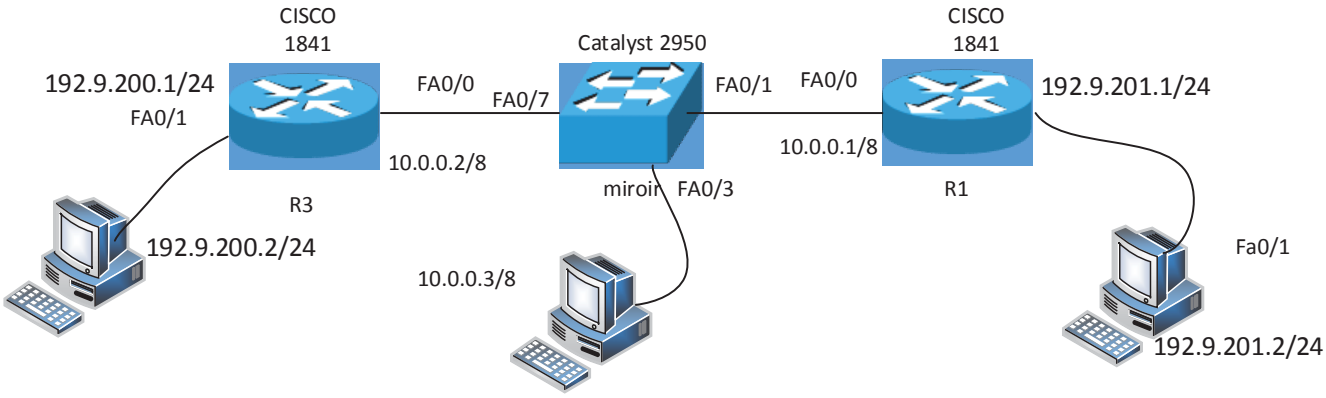
4.5.3) Les clés pré-partagés coté client et serveur doivent-elles être identiques ?

Fournir l'adresse IP du serveur VPN et gardez le même nom de groupe et le même mot de passe que le serveur. Cliquez droit sur testvpn et se connecter



4.5.4) Faire une capture de trame sur le poste 192.9.201.2. Quel est le protocole de transport utilisé sur IPSEC

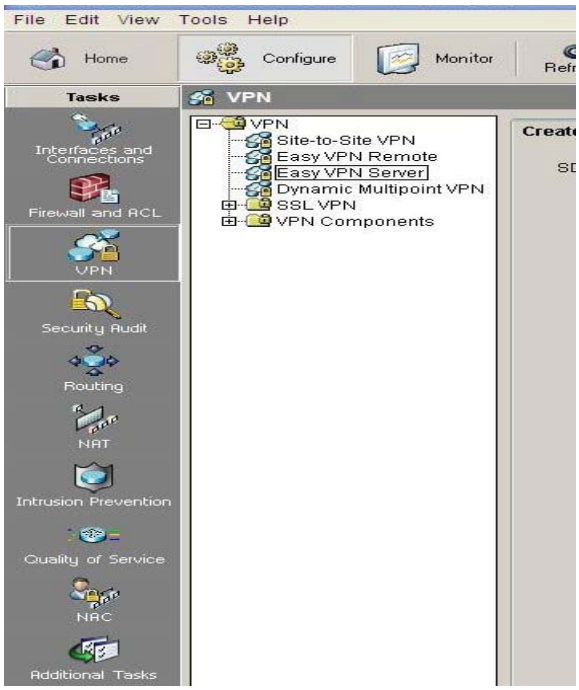
4.6) Mise en place d'un tunnel IPSEC entre deux routeurs



4.5.1) Mettre en place les différentes liaisons non sécurisées et vérifiez si les différents postes informatiques se voient sur le réseau.
4.5.2) Suivre la démarche suivante pour mettre en place une liaison sécurisée VPN entre les deux routeurs.

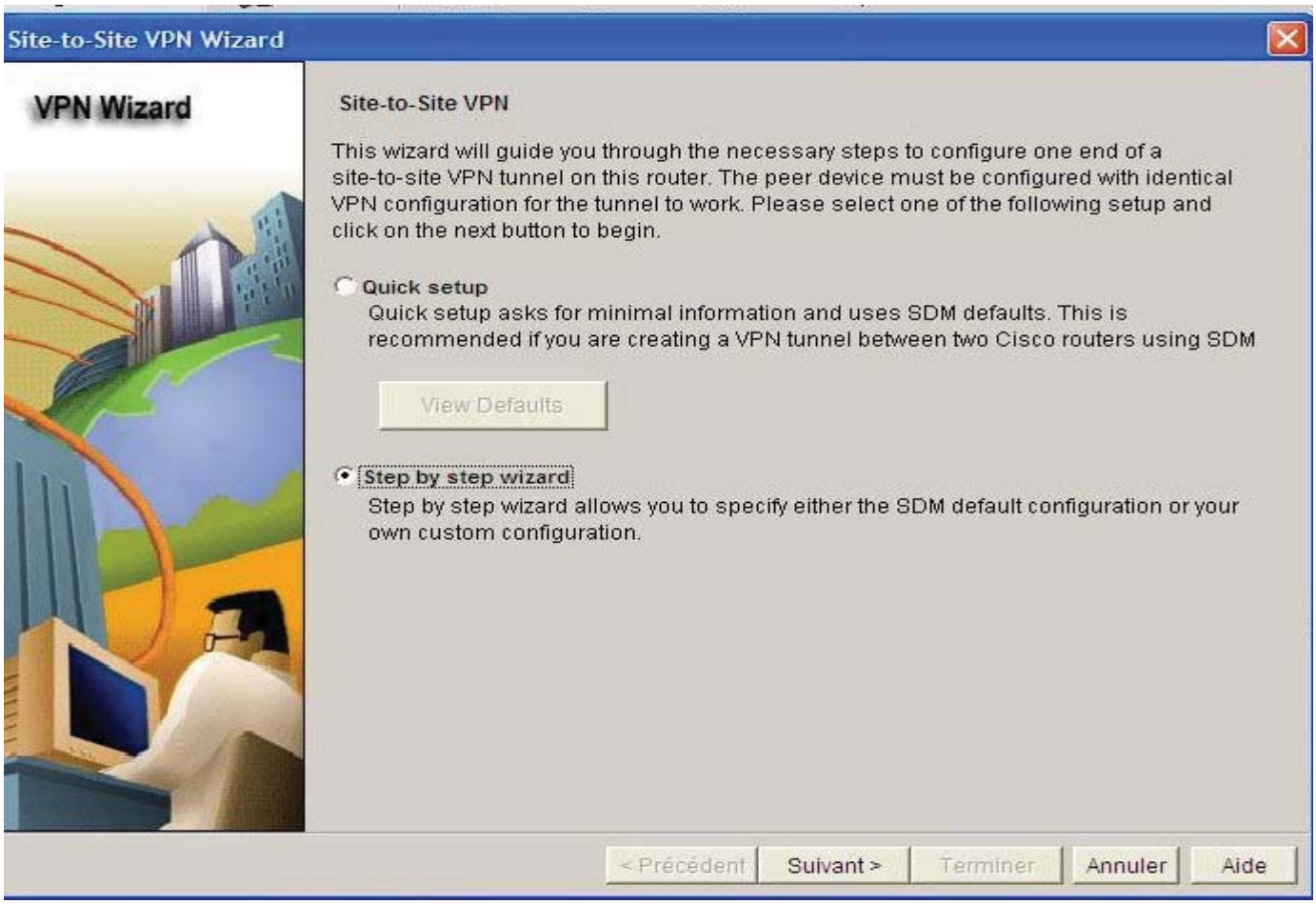
Grace à un client WEB (internet explorer), tapez l'adresse IP du routeur R1 afin de faire apparaitre la page de configuration.

Cliquez sur configure et sur VPN. La page suivante apparait :

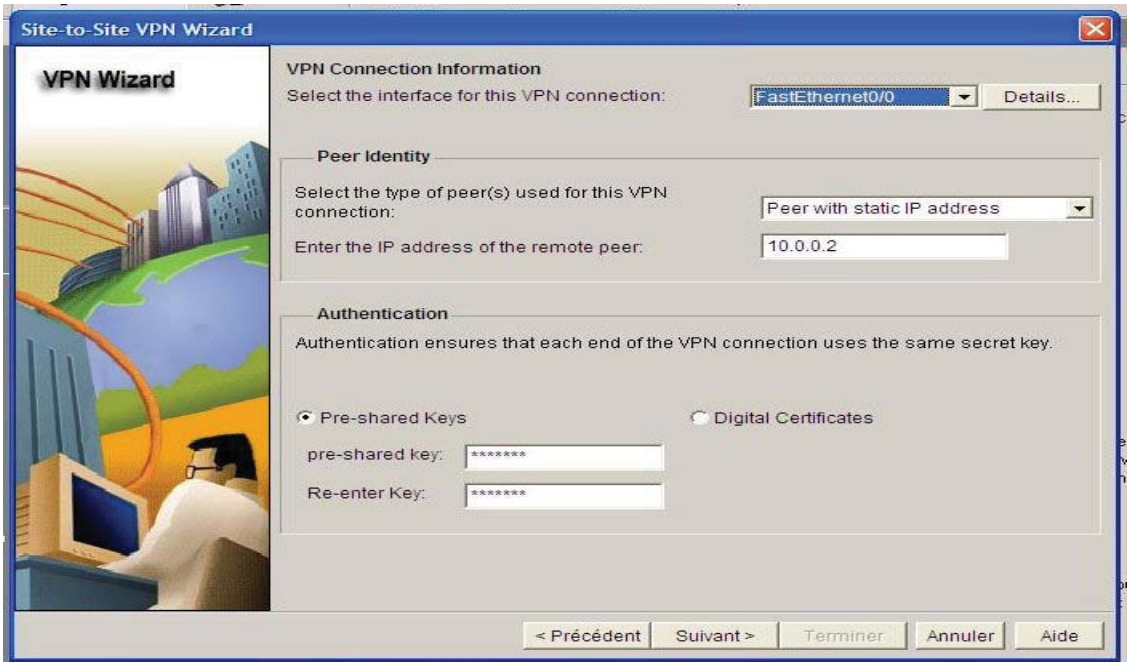


Cliquez sur site to site VPN

Choisir une configuration en mode pas à pas

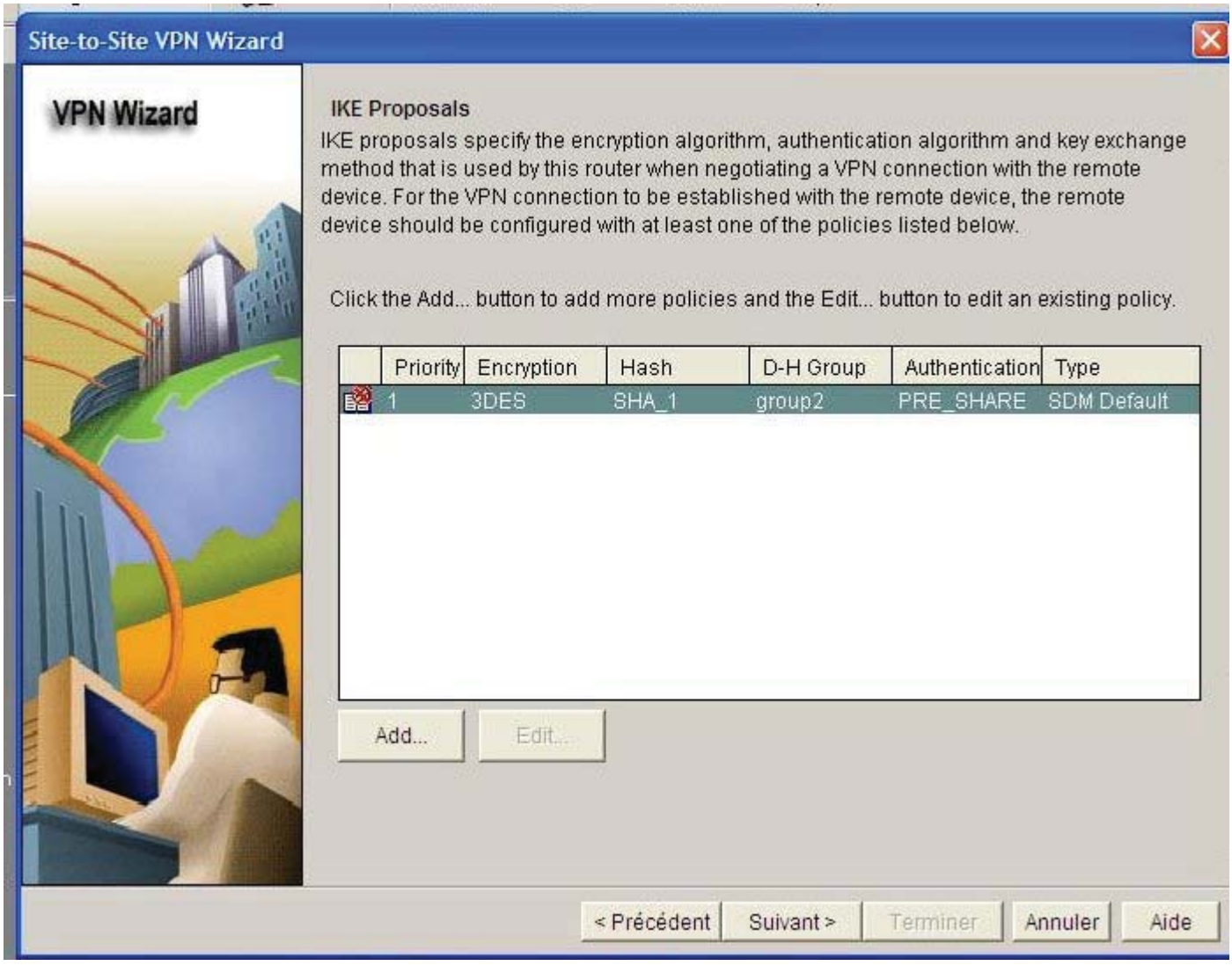


Sélectionnez l'interface sur laquelle le routeur met en place cette liaison VPN (chiffrement symétrique clés pré-partagés).
Indiquez l'adresse IP de l'interface distante du routeur ou se termine cette liaison VPN et fournir les clés pré-partagées :



```
crypto isakmp key cisco1* address 10.0.0.2
crypto map SDM_CMAP_1 1 ipsec-isakmp
 description Tunnel to10.0.0.2
 set peer 10.0.0.2
crypto isakmp key cisco1* address 10.0.0.2
```

La fenêtre suivante nous donne la configuration de IKE (Internet key echange) qui permet de mettre en place les protocoles pour l'échange des clés lors de l'ouverture du tunnel



Gardez la configuration par défaut :

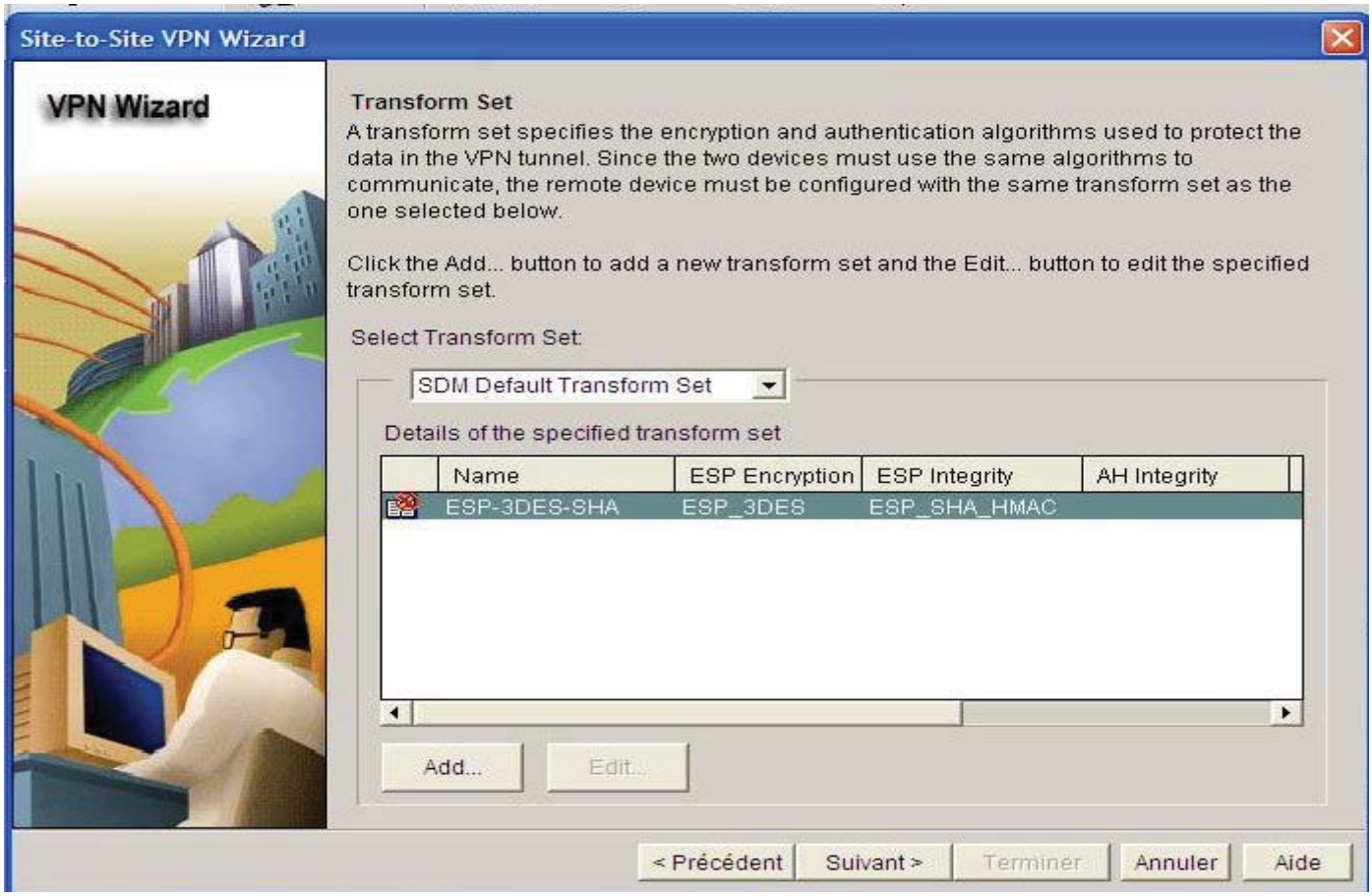
- ✓ L'échange de clé s'effectue grâce à un protocole diffie heffman groupe 2
- ✓ Une signature du message du type SHA version 1
- ✓ un chiffrement symétrique du type 3DES grâce aux clés pré-partagées

La page suivante transform set permet de configurer la façon dans les données vont être transformés et chiffrées dans le tunnel. On choisira la configuration par défaut suivante :

- ✓ un chiffrement symétrique du type ESP_3DES
- ✓ Une signature du type ESP_SHA_HMAC

On obtient les commandes suivantes pour le routeur :

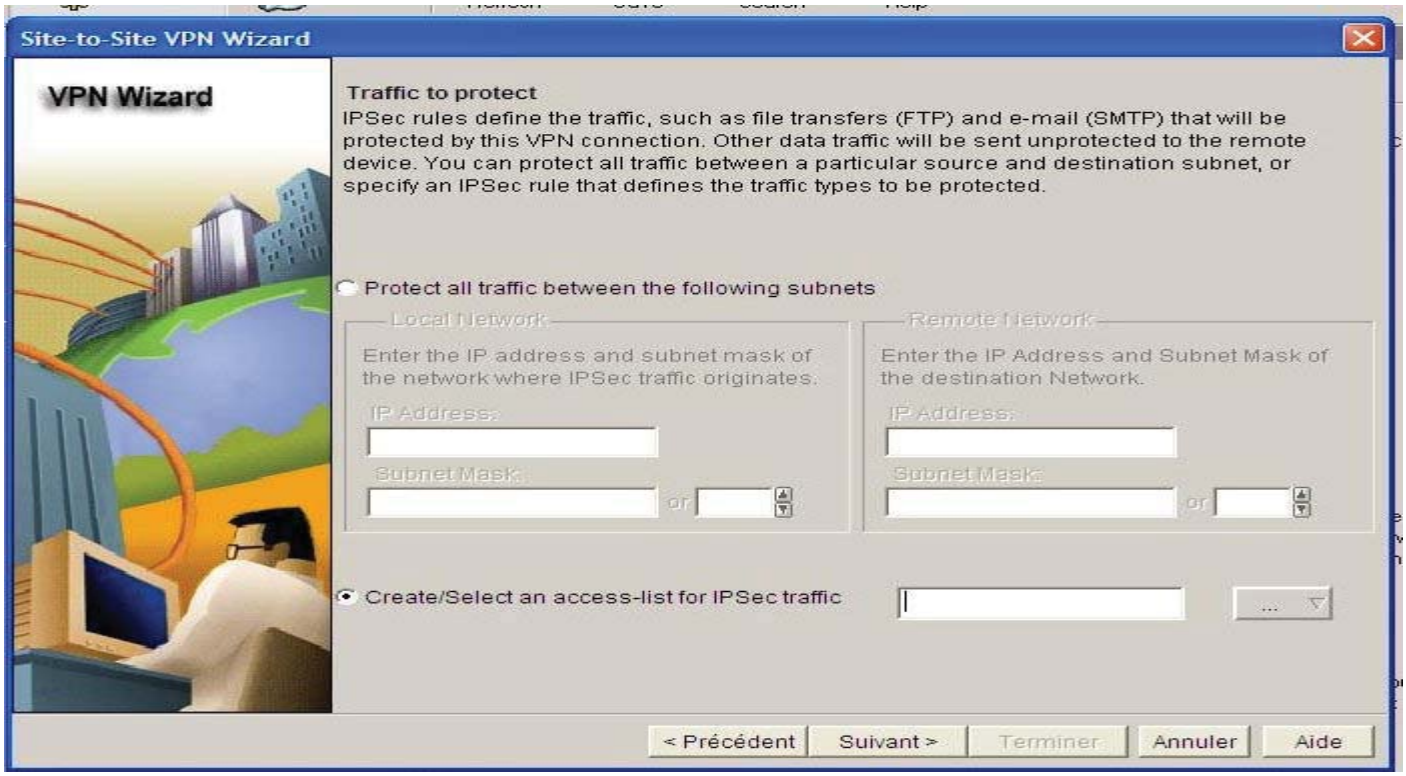
```
crypto isakmp policy 1
 encr 3des
 authentication pre-share
 group 2
```



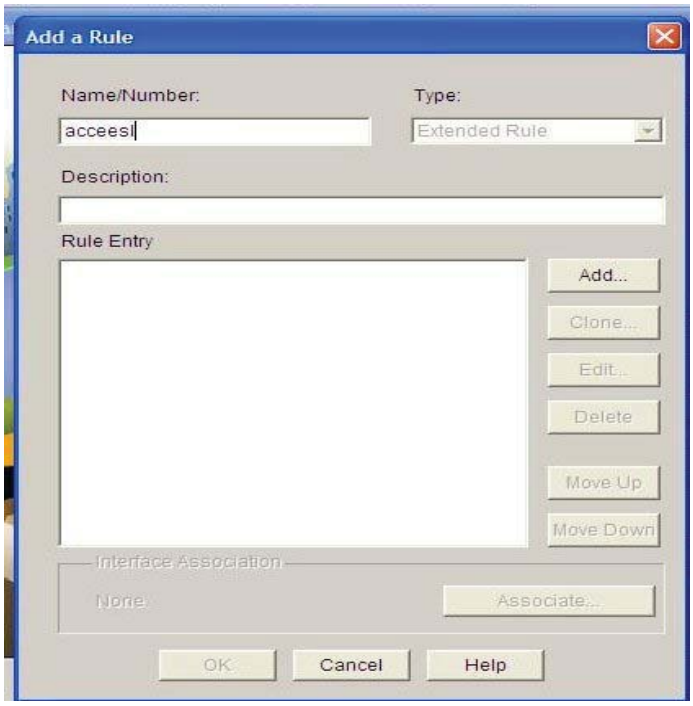
Ce qui donne la ligne de commande suivante :

```
set transform-set ESP-3DES-SHA
```

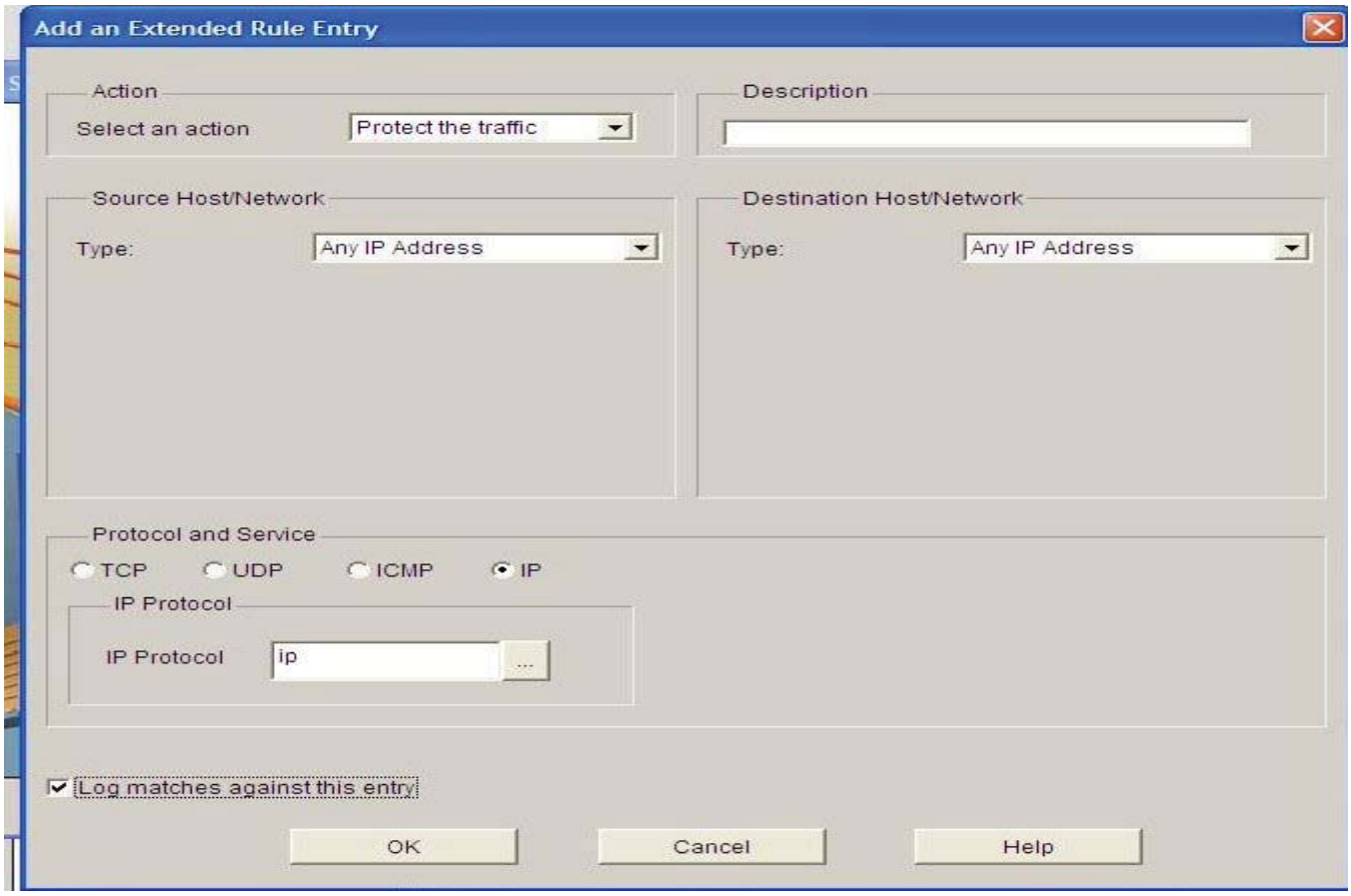
On va mettre en place une access list afin de chiffrer tout le contenu binaire entre les 2 routeurs :



Donnez comme nom à l'access-list accessl puis cliquez sur add



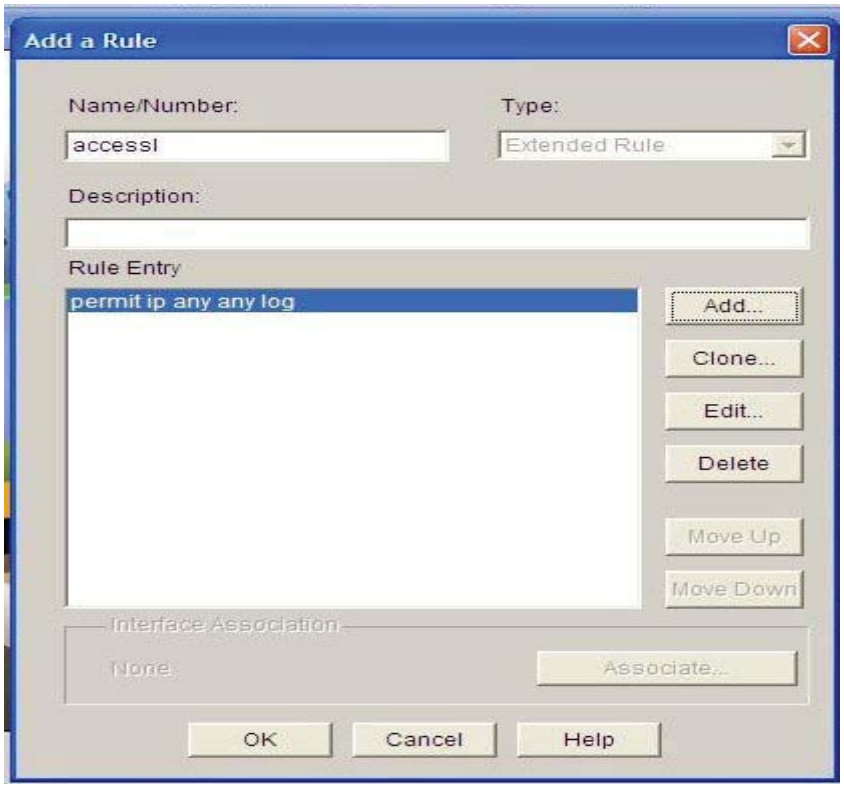
Protégez tout le trafic entre les 2 routeurs (page par défaut) faire next :
(évitez de cocher log matches against this entry)

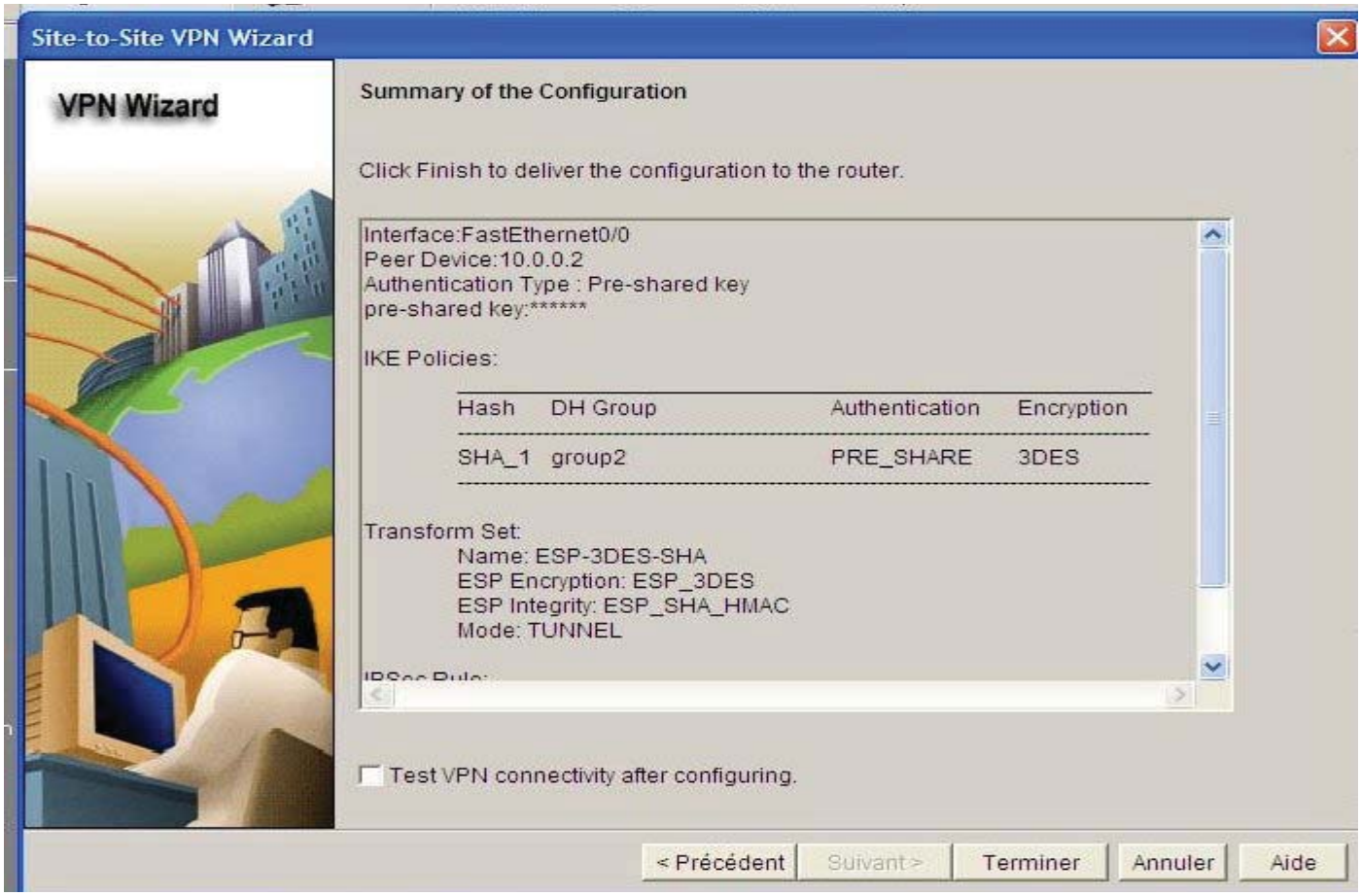


Ce qui donne la ligne de commande suivante :

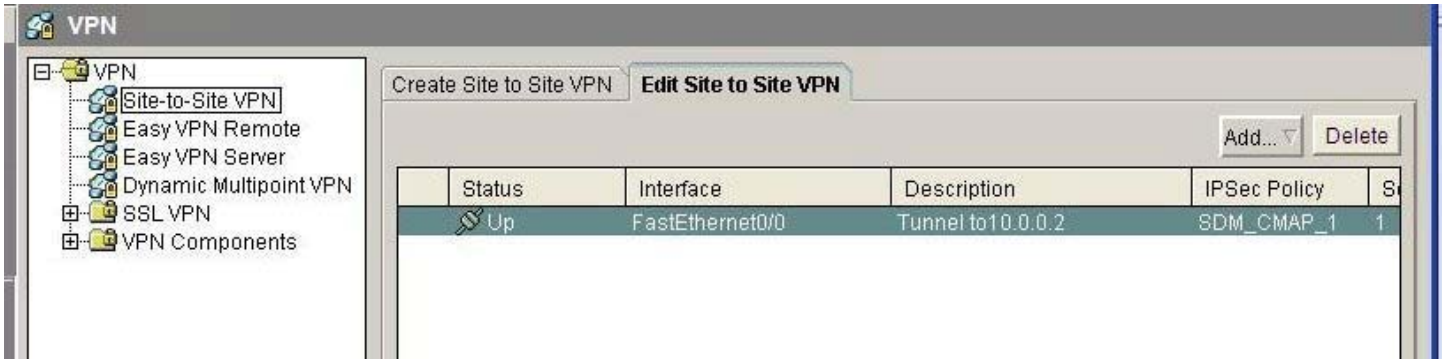
```
match address accessl
```

Les feuilles suivantes résument nos choix :

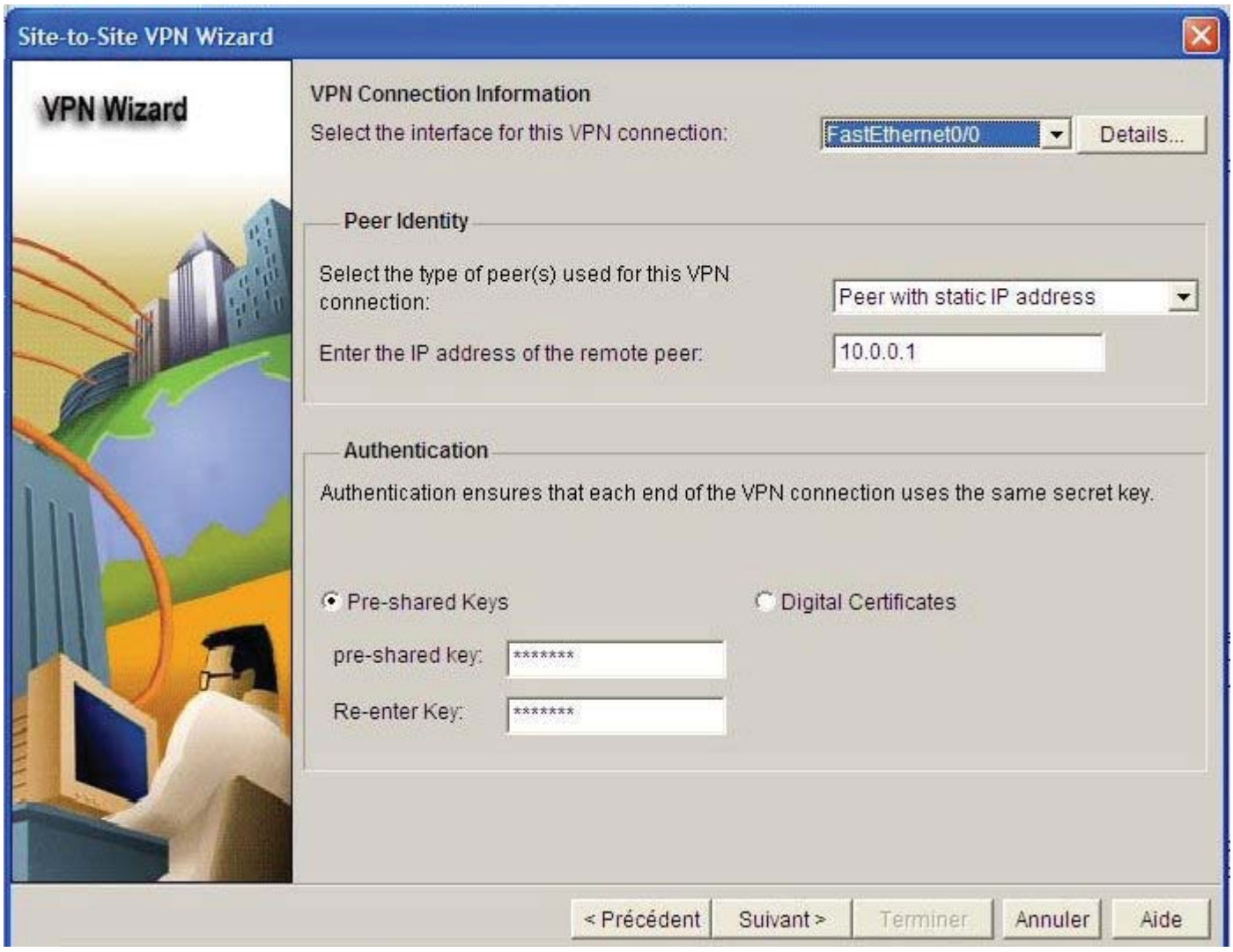




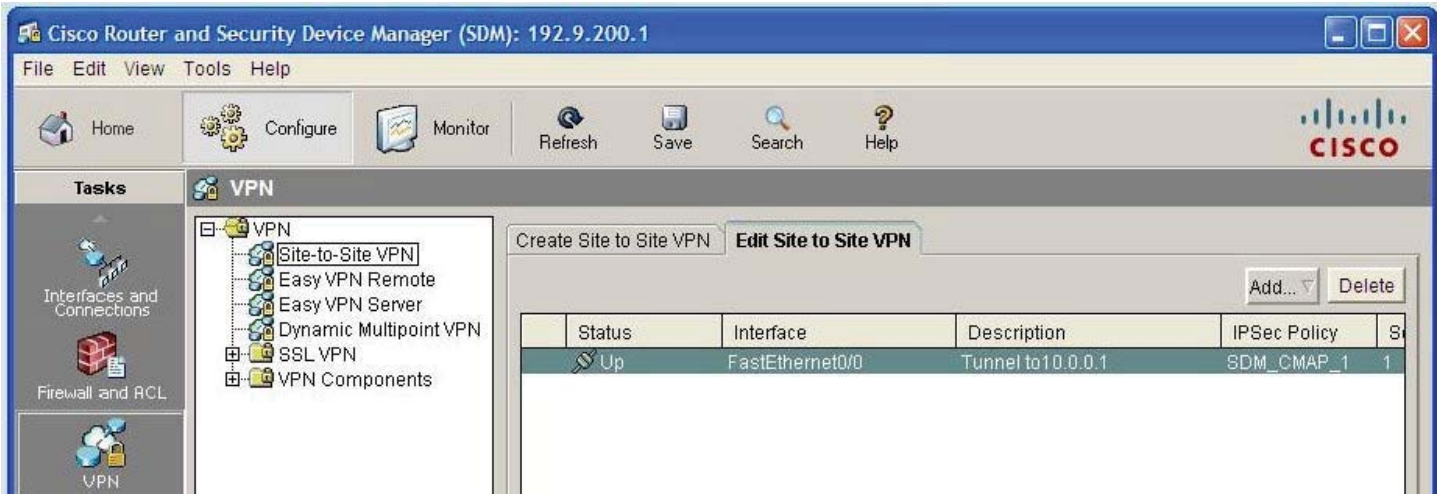
Le tunnel est pour l'instant down et passera UP lors de la fin de la configuration de R3



Grace à un client WEB (internet explorer), tapez l'adresse IP du routeur R3 afin de faire apparaitre la page de configuration.
La démarche reste identique au routeur R1, la seule modification se trouve sur la page suivante :



A la fin de la configuration de R1, les deux tunnels doivent passer à l'état UP :



Dossier annexe

1) Les commandes CISCO pour un switche

- Démarrer +programme+accessoire +Communication + Hyperterminal:
  - Nom = nom du switch +OK
    - Port Com = l + OK
    - Vitesse = 9600 bauds + Appliquer + OK
- Après Password
    - SwitchX> **en** ou **enable**
    - SwitchX#

- **Les commandes générales**

Switch ou router# erase startup-config (pour effacer la configuration)
Switch ou router#copy running-config startup-config
Switch ou router# reload (pour redémarrer)
Switch ou router(config)# no ip domain-lookup
(Pour éviter que le routeur tente de traduire des commandes incorrectement entrées , dévalidez la recherche DNS.

- **configuration de SSH**

Switch(config)#username cin privilege 15 secret cisco
Switch(config)#ip domain-name cin.com (création d'un domaine)
Switch(config)#crypto key generate rsa
Switch(config)#line vty 5 15 (10 connexions autorisées)
Switch(config-line)#login local
Switch(config-line)#transport input ssh
Switch(config)#line vty 0 4 (5 connexions autorisées)
Switch(config-line)#login local
Switch(config-line)#transport input telnet ssh (mode Telnet et SSH)

- **configuration d'un serveur web**

conf t
Router ou switche (config)#ip http server (validation serveur web)
Router ou switche (config)#username "cisco" privilege 15 secret "cisco"
Login cisco password cisco
Router(config)# ip http secure-server (Necessaire SSH site https)

Router ou switche (config)#ip http authentication local

- Création de VLAN (Virtual Local Area Network) :
  - SwitchX#**vlan database**
  - SwitchX(VLAN)#**vlan** *«numéro du vlan»* **name** *«nom du vlan n »*

vlan *10* **name** *cin1*

  - SwitehX(VLAN)#**apply**
  - SwitehX(VLAN)#**exit**
  - SwitchX#**sh vlan**
- Suppression un VLAN :
  - SwitchX#**vlan database**
  - SWitehX(VLAN)#**no vlan** *«numéro du vlan »*
  - SwitchX(VLAN)#**apply**
  - SwitchX(VLAN)#**exit**
- Affectation des interfaces aux vlan :
  - SwitchX#**conf t**
  - SwitchX(config)#**int** *« numéro de l'interface»*
  - SwitchX(config-if)#**speed 100**
  - SwitchX(config-if)#**duplex half**
  - SwitchX(config-if)#**switchport access vlan** *« numéro du vlan n»*
- Affectation **d'un port avec une adresse MAC** :

On peut spécifier une @Mac par port pour le bloquer si une autre adresse Mac est connectée

  - SwitchX#**conf t**
  - SwitchX(config)#**int** *« numéro de l'interface »*
  - SwitchX(config-if)#**switchport mode access**
  - SwitchX(config-if)#**switchport port-security**
  - SwitchX(config-if)#`switchport port-security maximum` *nombre*

```
switchport port-security maximum 2
```

  - SwitchX(config-if)#**switchport port-security mac-address** *«@mac»*

```
switchport port-security mac-address 0018.DED7.C6CA
```

o Pour trouver l'@ mac d'un PC : sous msdos : ipconfig –all

Pour visualiser les interfaces qui sont protégées :

o SwitchX# show port-security interface name

```
Switch#show port-security interface fa0/13
```

o Switch#show port-security address

Pour désactiver les sécurités :

SwitchX(config-if)#No switchport port-security

SwitchX(config-if)#No switchport port-security violation protect

SwitchX(config-if)#No switchport mode access

#No switchport port-security maximum

#No switchport port-security mac-address 0018.ded7.c6ca

- Port mirroring :

Le port mirroring est très utile il permet de faire une image du flux binaire qui arrive sur un port et de renvoyer cette image (miroir) vers un autre port

Port-mirroring sur port(s) source .On définit le port source dont le trafic sera copié (both, rx pour réception ou tx pour transmission):

Switch#

Switch#conf t

Switch(config)#monitor session 1 source interface fa0/12 both

Switch(config)#monitor session 1 destination interface fa0/10

pour plusieurs ports :

switch(config)#monitor session 1 source interface range fa0/12 - 19 both

Visualiser le port-mirroring

switch#show monitor session 1

pour effacer le miroir

switch(config)#no monitor session 1

- Affectation d'une adresse IP à un VLAN pour administration à distance :

Pour l'administration à distance d'un switch, il faut l'emploi de telnet.

- SwitchX#**conf t**
- SwitchX(config)#**int vlan 1**
- SwitchX(config-if)#**ip add** « *adresse IP du vlan 1* » « *masque par défaut*»
- SwitchX(config-if)#**no shut**

Pour avoir accès au switch, il faut faire la commande suivante sous DOS :

telnet « *adresse IP du vlan1*>>

- Sécurisation de l'accès du switch à distance :

Pour cela, il faut créer un access-list. II faut créer une règle et l'appliquer à une interface.

- Création de la règle :
- SwitchX#**conf t**
- SwitchX(config)#**ip access-list standard** « *numéro de la règle de l à 99*>>
- SwitchX(config-std-nacl)#**permit host** « *@IP du poste qui peul administrer à distance* »
- SwitchX(config-std-nacl)#**deny any**
- SwitchX(config-std-nacl)#**exit**
- Application de la règle à une interface :
- SwitchX(config)#**int** vlan 1
- SwitchX(config-if)#**ip access-group** « *numéro de la règle n*>>

Pour supprimer cet access-list, il faut taper les lignes de commandes suivantes :

SwitchX(config)#**no ip access-list standard** « *numéro de la règle n*>>

SwitchX(config-it)#**no ip access-group** « *numéro de la règle*>> in

- Les commandes CISCO pour l'agrégation de liens

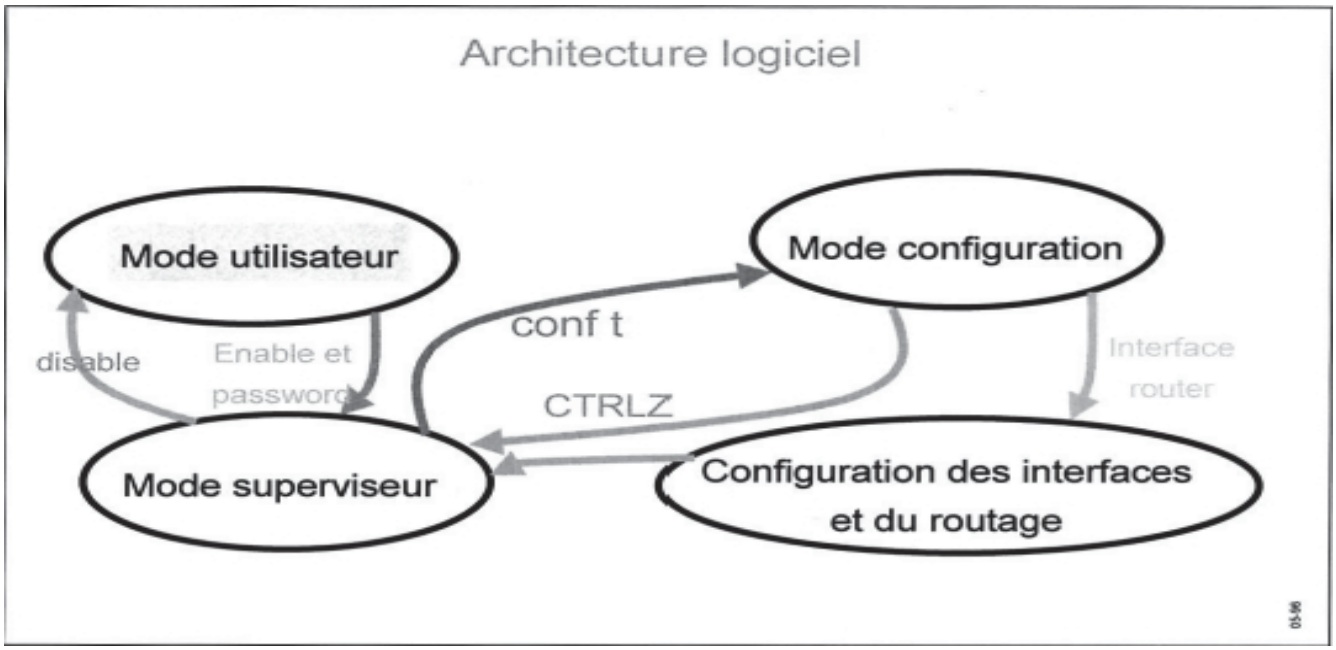
conf t (sur le premier switche en mode actif)
Switch#conf t
Switch(config)#port-channel load-balance src-mac
Switch(config)#int gi0/1

```
Switch(config-if)#channel-protocol lacp
Switch(config-if)#channel-group 1 mode active
Switch(config)#int gi0/2
Switch(config-if)#channel-protocol lacp
Switch(config-if)#channel-group 2 mode active

Switch#conf t (sur le deuxième switche en mode passif)
Switch(config)#port-channel load-balance src-mac
Switch(config)#int gi0/1
Switch(config-if)#channel-protocol lacp
Switch(config-if)#channel-group 1 mode passive
Switch(config)#int gi0/2
Switch(config-if)#channel-protocol lacp
Switch(config-if)#channel-group 2 mode passive
```

2) Les commandes CISCO pour un routeur

Les différents modes :



- Démarrer + **programme** + accessoire + Communication + Hyperterminal :
  - Nom = nom du routeur +OK
  - Port Com l + OK
  - Vitesse = 9600 bauds + Appliquer + 0K

- Après Password :

Router1> **en** ou **enable** -->passage en mode superviseur
Router1 # **sh run** --> visualise la configuration
Router1 # **sh ip route** -->pour voir les tables de routages

Configuration des interfaces :

- Routeur1# **conf t** Configuration du terminal
- Routeur1 (config)**#int** « nom de l'interface » *ex : int.ea0/0 ou int fa0/0*
- Routeur1 (config-if)**#ip add** « @IP de l'interface » « Masque du réseau de l'interface» *ex: ip add 192.168.2.202 255.255.255.0*

• Pour les interfaces Séries DTE :
- Routeur1(config-if)**#encap lapb**

. Pour les interfaces Séries DCE :
- Routeur1(config-if)**#encap lapb DCE**

. Pour les interfaces séries DTE
- Routeur1(config-if)**#encap lapb DTE**
- Routeur1(config-if)**#clock rate 1300000**

oRouteur1(config-if)#**no shut** => pour activer l'interface (l'interface est montée)

Pour faire tomber l'interface :

- Routeur1(config-if)#**shutdown**

Pour une interface LAN fast ethernet

o Routeur1 (config-if)#**ip add** « @IP de l'interface » « Masque du réseau de l'interface» *ex: ip add 192.168.2.202 255.255.255.0*
Router(config-if)#speed 100
Router(config-if)#duplex half
Router(config-if)#no shut
Router(config-if)#

- **Configuration du Routage:**

oRoutage statique :

- Routeur1 **#conf t**
- Routeur1(config)#**ip routing** => Validation du routage
- Routeur1 (config)#**ip route** « @IP du réseau à atteindre » « Masque » « @IP de l'interface d'en face par ou l'on veut passer »
  Ex : ip route 192.168.2.0 255.255.255.0 192.168.20.201

o Supprimer le routage statique :

- Routeur1(config)#**no ip route** « ligne de routage à supprimer » => supprime une ligne de routage
  Ex :no ip route 192.168.1.0 255.155.255.0 192.168.20.201
- Routeur1(config)#**no ip routing** => supprime toutes les lignes de routage

o Routage dynamique :

- Routeur1 **#conf t**
- Routeur1(config)#**no ip routing**
- Routeur1(config)#**ip routing**
- Routeur1(config)#**router rip**
- Routeur1 (config-router)#**net** «@IP réseau du LAN(s)»
- Routeur1(config-router)#**net** «@IP réseau du WAN(s)» => si elle existe
- Routeur1 (config-router)#**net** « @IP du WAN2 » => si elle existe

o Supprimer un routage dynamique :

- Routeur1(config-router)#**no net** « @IP réseau du LAN(s) »
- Routeur1(config-router)#**no net** « @IP réseau du WAN(s) »

- **Tunneling :** (sert à protéger les infos par encapsulation)

o Montage d'un tunnel

- Routeur1 **#conf t**
- Routeur1(config)**#int** «t0 (nom de l'interface virtuelle) »
- Rouleur1(config-if)**#ip add** « @ IP de l'interface virtuelle » « Masque »
- Routeur1 (config-if)**#tun source** « nom de l'interface physique réelle »
- Routeur1(config-if)**#tun dest** << @IP de l'interface physique réelle d'en Face»
- Routeur1 (config-if)**#tun key** «chiffre (ex :1234) »
- R1(config)#no interface tunnel 1

1 numéro du tunnel

o On force les infos à passer par le tunnel :

- Routeur1**#conf t**
- Routeur1 (config)**#router rip**
- Routeur1 (config-router)**#no net** « @IP du réseau ou l'on à placer le tunnel »
- Routeur1 (config-router)**#net** « @IP du réseau du tunnel »

o Pour une interface WAN GHDSL

✓ Configuration pour une interface du type CPE

Routeur1#conf t pour l'interface cpe

Routeur1 (config)#interface atm0/0/0

Routeur1 (config-if) #ip address « @ IP de l'interface WAN CPE» « Masque »

Permet de spécifier une adresse IP pour l'interface WAN

Routeur1 (config-if) #pvc «vpi»/«vci»

Entre dans le mode de configuration de circuit virtuel ATM et configure un nouveau circuit virtuel permanent (PVC) en assignant un numéro VPI/VCI

Router(config-if-vc)#protocol ip « @ IP de l'interface WAN d'en face CO» broadcast

Valide une connectivité IP et créé une adresse IP point à point pour le circuit virtuel

Router(config-if-vc)#vbr-rt «debit max» «debit moyen»

Router(config-if-vc)#encapsulation aal5snap

Choix de la classe de service

Routeur1 (config-if-vc) #exit

Routeur1 (config-if) #dsl operating-mode gshdsl symmetric annex B

spécifie un mode de fonctionnement du GSHDSL pour une interface ATM A(format US) B (format UE)

Routeur1 (config-if) #dsl equipment-type cpe

Configure l'interface ATM DSL pour fonctionner comme matériel Co ou CPE.

Routeur1 (config-if) #dsl linerate 264

configure le débit de la ligne en Kbits/s **72**, **136**, **200**, **264**, **392**, **520**, **776**, **1032**, **1160**, **1544**, **2056**, and **2312**.

Routeur1 (config-if) #no shut

Routeur1 (config)#exit

Routeur1#exit

Routeur1#show atm vc

Routeur1#clear interface atm0/0/0

Routeur1#show controllers atm0/0/0

- ✓ Configuration pour une interface du type CO

Routeur1#conf t pour l'interface cpe

Routeur1 (config)#interface atm0/0/0

Routeur1 (config-if) #ip address « @ IP de l'interface WAN CO » « Masque »

Permet de spécifier une adresse IP pour l'interface WAN

Routeur1 (config-if) #pvc «vpi»/«vci»

Entre dans le mode de configuration de circuit virtuel ATM et configure un nouveau circuit virtuel permanent (PVC) en assignant un numéro VPI/VCI

Router(config-if-vc)#protocol ip « @ IP de l'interface WAN d'en face CPE» broadcast

Valide une connectivité IP et créé une adresse IP point à point pour le circuit virtuel

Router(config-if-vc)#vbr-rt «debit max» «debit moyen»

Router(config-if-vc)#encapsulation aal5snap

Choix de la classe de service

Routeur1 (config-if-vc) #exit

Routeur1 (config-if) #dsl operating-mode gshdsl symmetric annex B

spécifie un mode de fonctionnement du GSHDSL pour une interface ATM A(format US) B (format UE)

Routeur1 (config-if) #dsl equipment-type cpe

Configure l'interface ATM DSL pour fonctionner comme matériel Co ou CPE.

Routeur1 (config-if) #dsl linerate 264

configure le débit de la ligne en Kbits/s **72**, **136**, **200**, **264**, **392**, **520**, **776**, **1032**, **1160**, **1544**, **2056**, and **2312**.

Routeur1 (config-if) #no shut

Routeur1 (config)#exit

Routeur1#exit

```
Routeur1#show atm vc
Routeur1#clear interface atm0/0/0
Routeur1#show controllers atm0/0/0
```

Le chiffrement

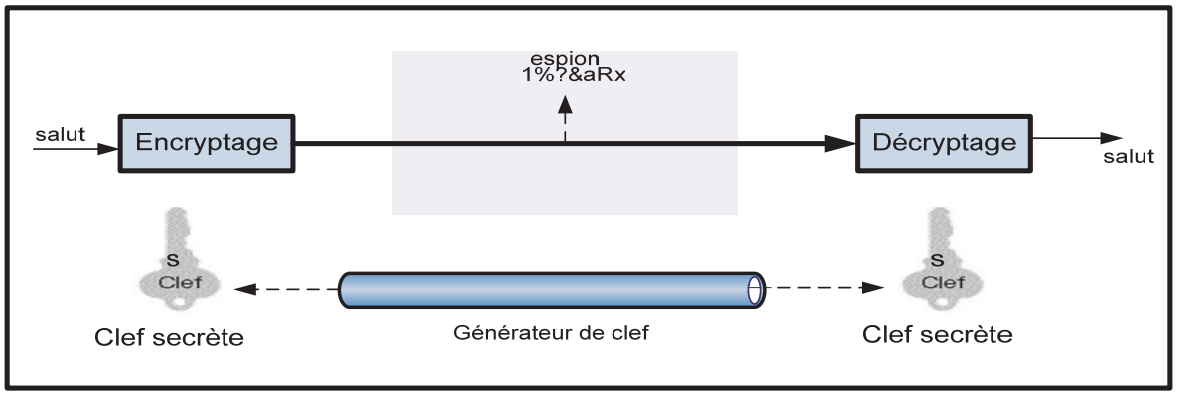
Un chiffrement moderne permet d'assurer :

- La **Confidentialité** des informations
- L'**Intégrité** des données en s'assurant que l'information n'a pas subi de modification
- **l'authentification** afin de vérifier que le message correspond bien à l'expéditeur.

Avant d'aborder les techniques modernes de chiffrement sur les réseaux nous allons rappeler les différents principes de chiffrement : le chiffrement symétrique, asymétrique ou mixte.

Cryptographie à clé secrète (chiffrement symétrique)

On utilise une clef secrète commune aux 2 systèmes qui veulent communiquer ensemble.
Les algorithmes symétriques actuels sont nettement plus rapides que les algorithmes asymétriques. Une clef symétrique est toutefois potentiellement peu sûre. Il suffit qu'un individu malintentionné se procure la clé secrète et il n'y a plus d'information protégée. La clef symétrique doit donc être transmise au correspondant sans être interceptée.



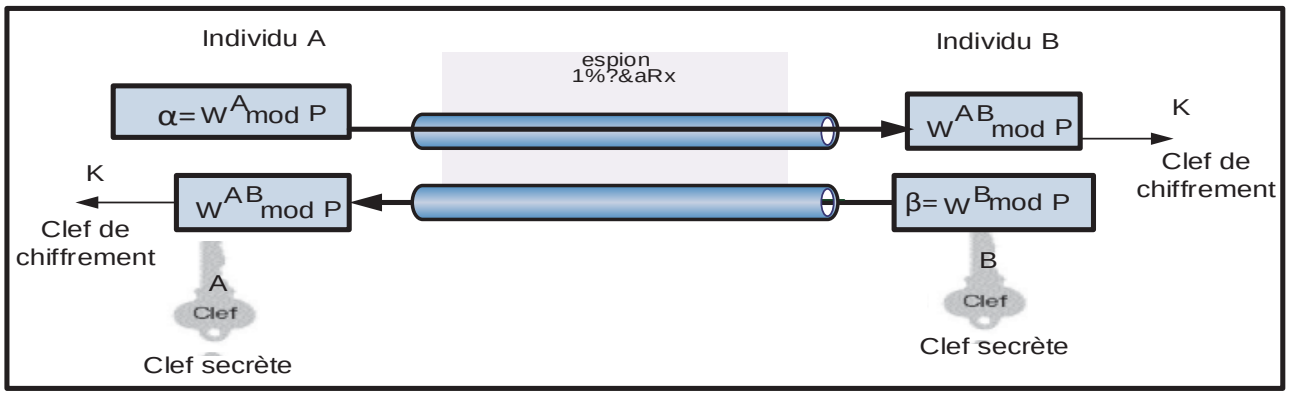
Les algorithmes de cryptage symétriques les plus répandus sont le **DES et le 3DES**.

La vulnérabilité de la cryptographie symétrique repose sur l'échange de clé.
Diffie et Hellman ont proposé une méthode afin de convenir d'un secret partagé sans le faire circuler entre les participants, en d'autres termes une fonction mathématique telle que les participants puissent échanger des informations dont eux seuls pourraient déduire le secret.

Le protocole repose sur la fonction suivante :

$$K = w^{A \times B} mod\ P$$

Cette fonction est publique, ainsi que les valeurs de W et P

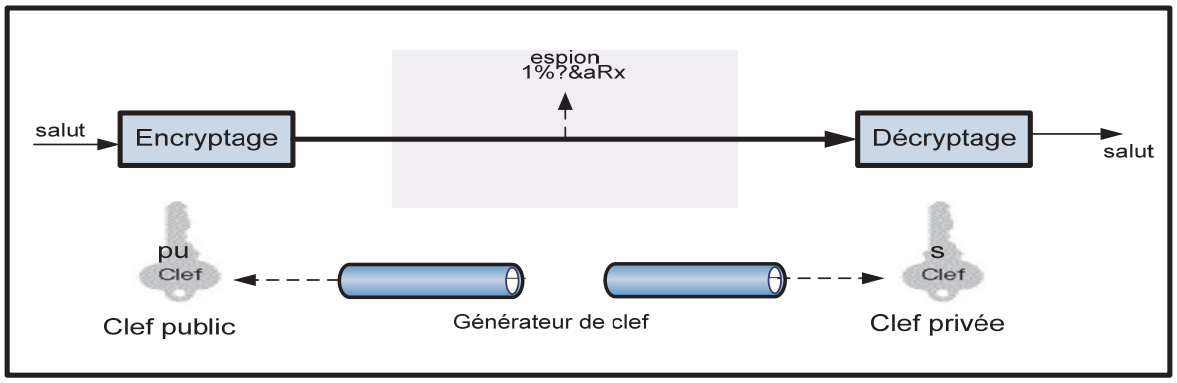


Chaque individu A et B dispose de sa propre clé secrète respectivement A et B. Chacun individu envoi une partie du message β et a sans directement transmettre A et B mais qui permet à A ou B de trouver la clé secrète K.

La clé K n'a jamais été directement transmise et il est très difficile à un espion de trouver la clé K connaissant que β et a. Le système reste fiable si P a un format de 512 octets.

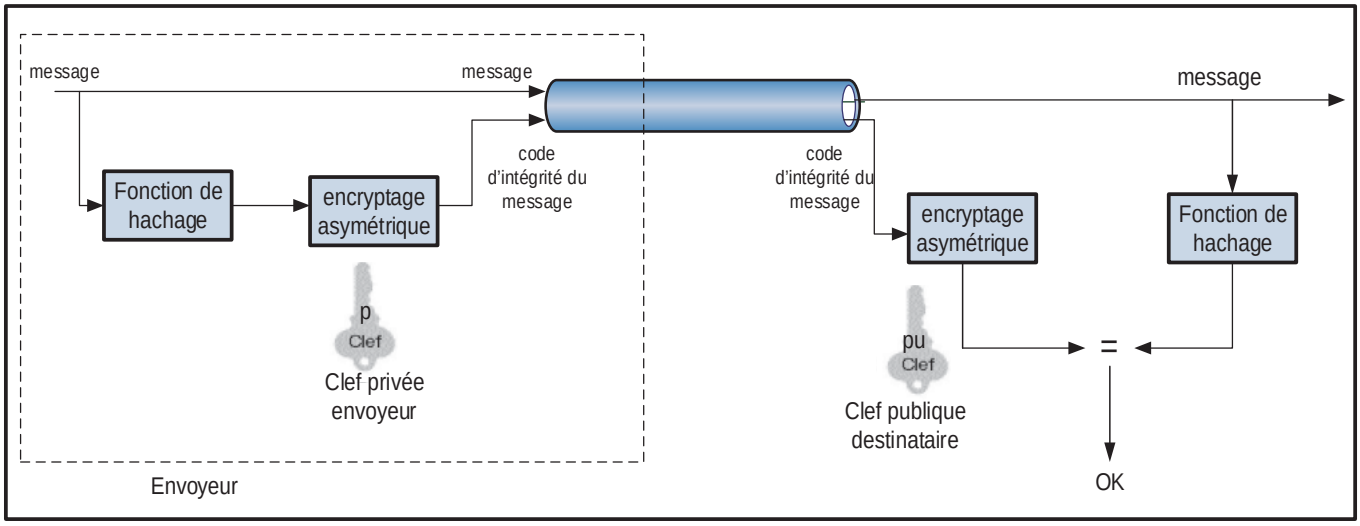
Cryptographie à clé publique (chiffrement asymétrique)

Le chiffrement à base de clef publique garantit qu'une donnée chiffrée par une clef, **publique ou privée**, ne peut être déchiffrée que par la clef associée. Les clefs sont de même nature et leurs rôles peuvent être inversés : ce que l'un chiffre est déchiffrable par l'autre. La paire de clef repose sur des nombres premiers.
L'astuce consiste à conserver une clef secrète (la clef privée) et à distribuer l'autre clef (la clef publique) à tout le monde. N'importe qui peut vous envoyer un message chiffré mais personne d'autre que vous ne peut le déchiffrer tant que vous êtes le seul à conserver un des deux membres de la paire de clefs.



La signature d'un message consiste à associer une identité à son contenu. L'identité du message est représentée par une suite binaire que l'on appelle le checksum. On applique le message à l'entrée d'une fonction de hachage qui fournit en sortie un résultat appelé haché. Le haché est ensuite crypté par la clef privée. Le destinataire de l'ensemble (message + code

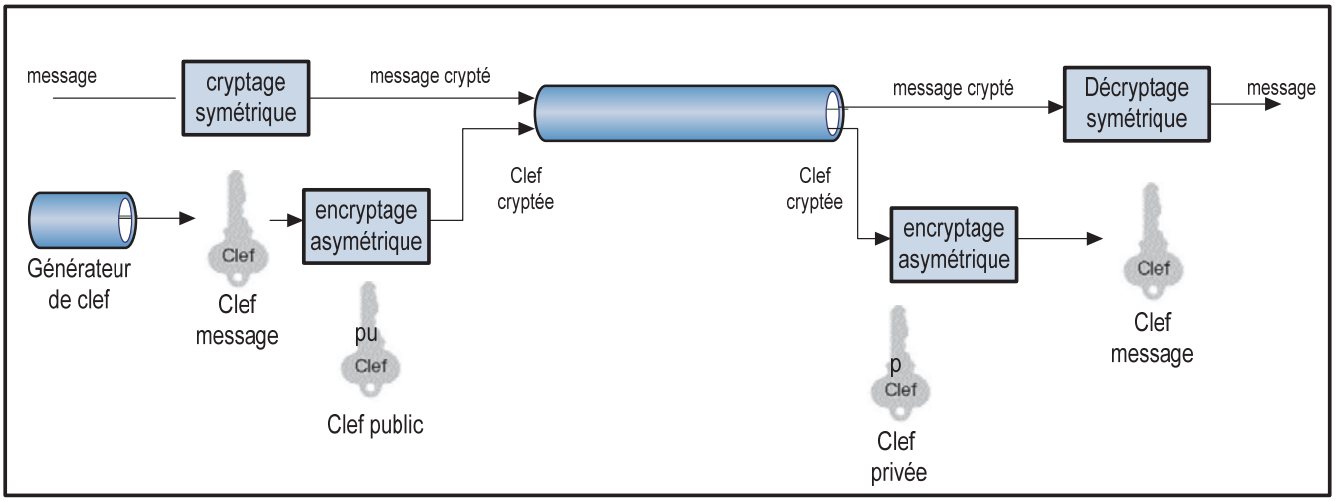
d'intégrité) recalcule le haché et le compare au résultat du déchiffrement de la signature ou code d'intégrité avec la clef publique.



La caractéristique principale d'une fonction de hachage est de produire un haché des données, c'est-à-dire un condensé de ces données.

La solution mixte

On utilise un système asymétrique (comme **RSA** ou **Diffie-Hellman**) pour crypter les clés de Message. Ces clés de message permettront de chiffrer le message grâce à un chiffrement symétrique (DES, 3DES, IDEA, RC4...).



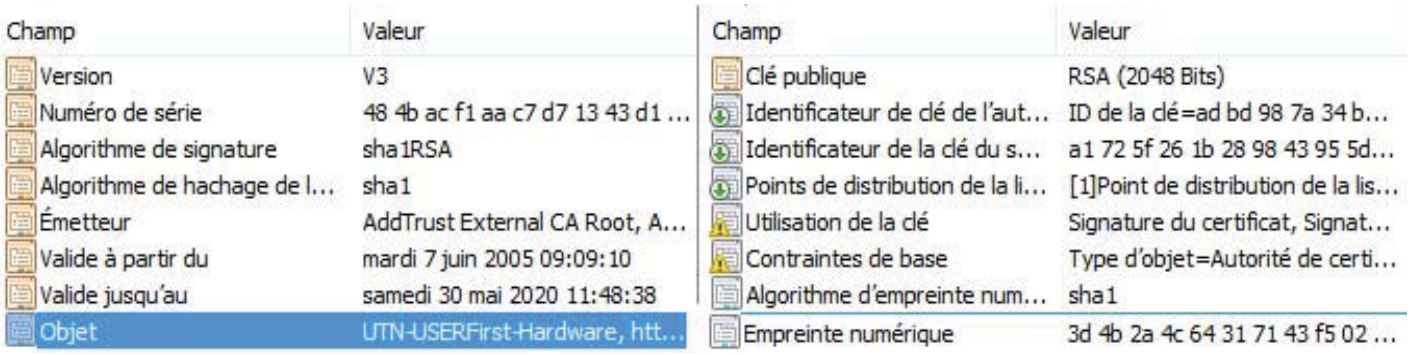
Le problème du chiffrement asymétrique vient de la transmission de la clé publique. Si celle-ci n'est pas sécurisée, une personne malveillante peut se positionner entre l'entité et son public en diffusant de fausses clés publiques (par le biais d'un faux site internet par exemple) puis

intercepter toutes les communications, lui permettant d'usurper l'identité du diffuseur de clés publique.
Pour un chiffrement asymétrique le passage de clé s'effectue par un certificat. Plus précisément, un certificat est une structure de données contenant une clé publique, des informations sur le propriétaire du certificat, l'ensemble étant signé par une autorité tierce appelée *Autorité de Certification (AC,ou Certificate Authority : CA).*
Le format de certificats le plus utilisé dans le cadre d'une *PKI (Public Key Infrastructure ou IGC)* est X.509v3. Ce dernier permet l'utilisation des protocoles normalisés tels que SSL ou encore IPSec.

Un certificat numérique contient les informations suivantes :

• La *clé cryptographique publique* de la paire de clés du sujet du certificat.
• Des *informations sur le sujet* qui a demandé le certificat.
• Des *informations sur l'Autorité de certification* qui a émis le certificat.

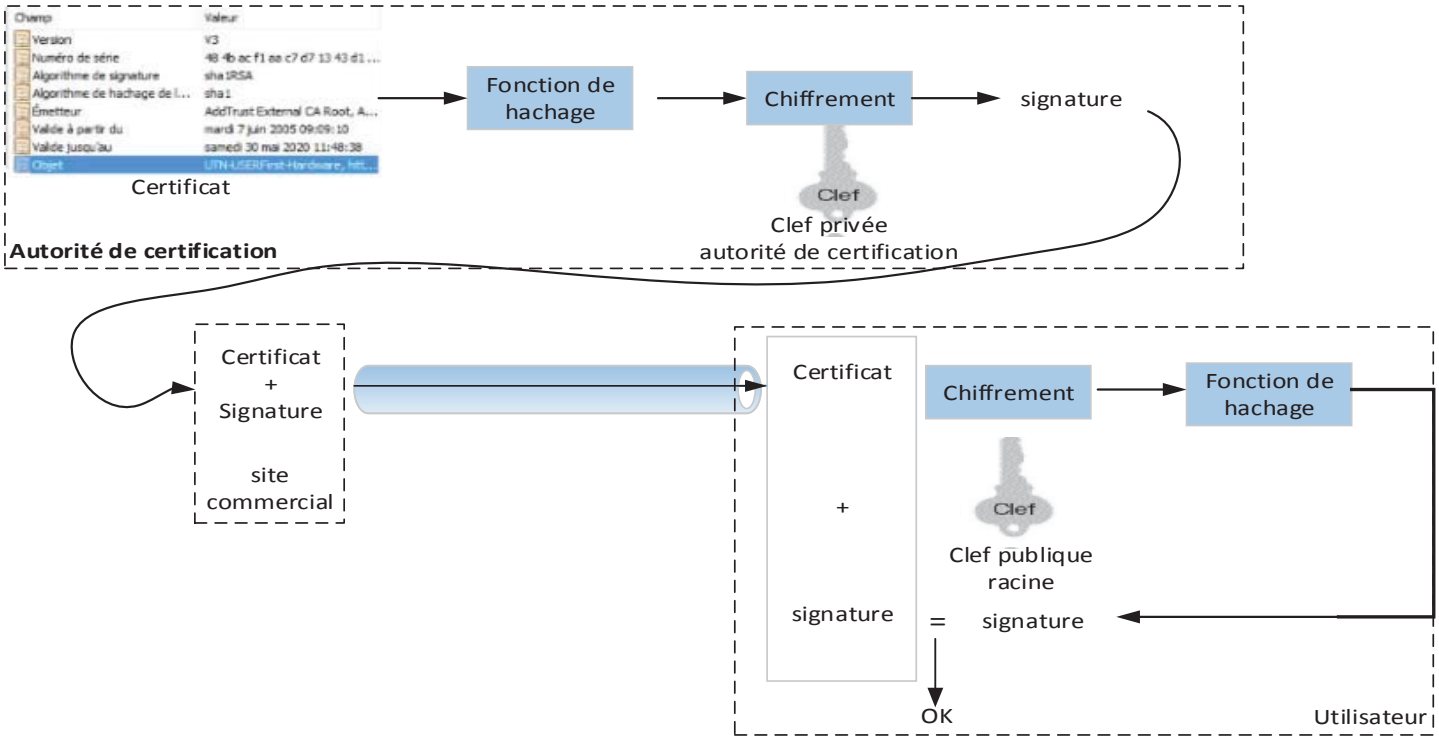
Avant d'émettre un certificat, l'Autorité de certification vérifie l'identité du demandeur. Cette vérification peut inclure une enquête de fond manuelle du demandeur ou un examen de la liste de contrôle d'accès discrétionnaire (DACL, *Discretionary Access Control List*) du modèle de certificat demandé, et ce pour garantir que l'utilisateur ou l'ordinateur à l'origine de la demande dispose des autorisations nécessaires pour inscrire le certificat demandé.



Les champs de base sont les suivants :

• **Version :** indique à quelle version de X509 correspond ce certificat.
• **Numéro de série :** Numéro de série du certificat.
• **Algorithme de signature :** identifiant du type de signature utilisée.
• **Emetteur :** DN de l'autorité de certification qui a émis ce certificat.
• **Valide à partir de :** la date de début de validité de certificat.
• **Valide jusqu'à :** la date de fin de validité de certificat.
• **Objet :** Distinguished Name (DN) de l'objet de ce certificat.
• **Clé publique de l'objet :** infos et clé publique de ce certificat.

La création d'un certificat et son identification s'effectue de la façon suivante:



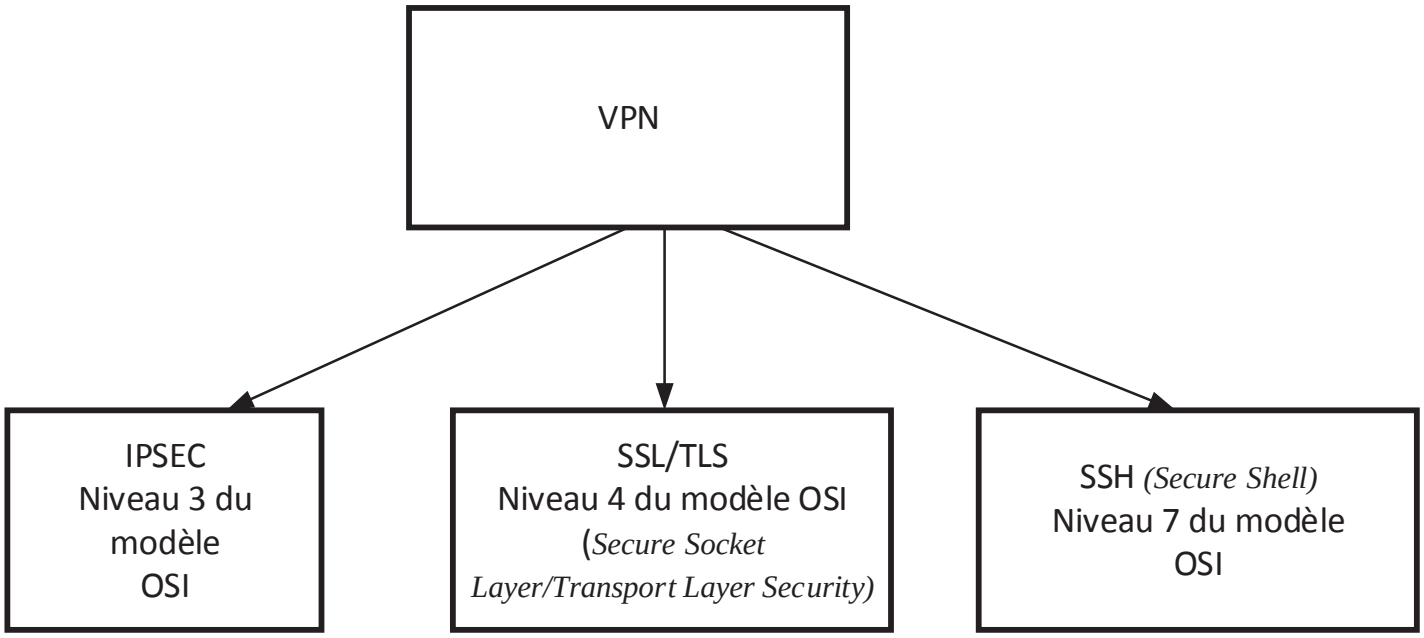
Les autorités de certification sont des organismes enregistrés et certifiés auprès d'autorités publiques ou de gouvernance de l'internet qui établissent leur viabilité comme intermédiaire fiable. Ces organismes diffusent leurs propres clés publiques. Étant certifiées fiables ces autorités sont en contact direct avec les principaux producteurs de système d'exploitation et de navigateurs internet (tels que Mozilla Firefox, Internet Explorer, Google Chrome…) qui incluent nativement les listes de clés des autorités de certification. Ces clés sont appelées ***clés publiques racines*** ou *clé root*, elles sont utilisées pour identifier les clés publiques d'autres organismes.

Lorsqu'un site commercial veut diffuser une clé publique dans un certificat, il contacte une autorité de certification. Celle-ci va recevoir la clé publique et l'identité du diffuseur. Après avoir vérifié l'identité du demandeur par des moyens conventionnels, l'autorité de certification place l'identité et la clé publique fournie dans un conteneur qu'elle signe en utilisant sa clé privée. Le fichier résultant est le certificat. Il est rendu au diffuseur ou au site commercial.

Le site commercial utilise le certificat pour ses communications (par exemple sur son site internet): un utilisateur reçoit le certificat, il le vérifie avec la **clé racine publique** de l'organisme de certification incluse dans son navigateur. Le déchiffrement authentifie le contenu du certificat et prouve que la clé publique et l'identité contenus dans le certificat sont certifiés liés par l'autorité de certification. L'utilisateur sait donc qu'il pourra récupérer les informations envoyées en utilisant la clé publique et qu'elles ne seront pas interceptées par un tiers.

VPN (virtual private network)

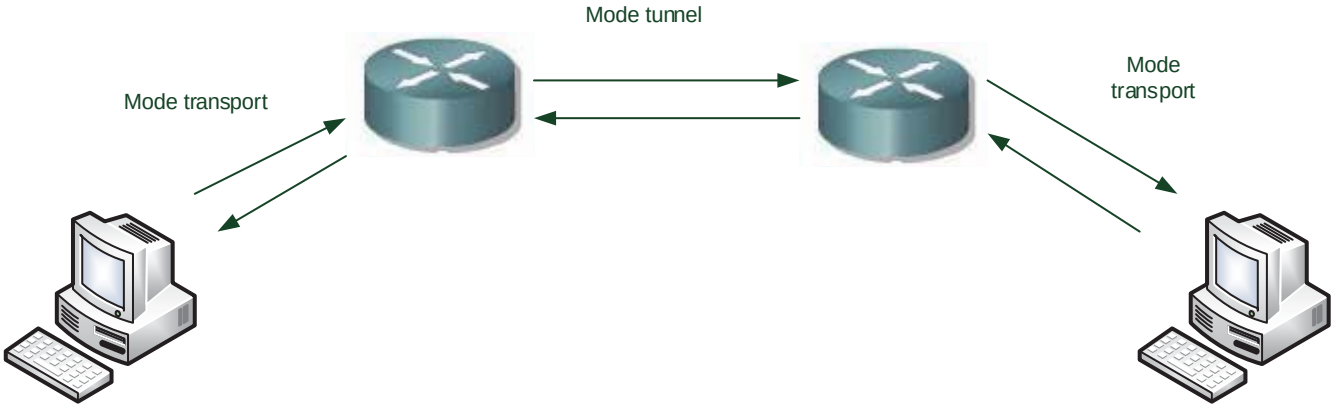
Le chiffrement du flux de l'ensemble du trafic sur un ou plusieurs itinéraires donnés constitue un *réseau privé virtuel*, ou VPN, (*Virtual Private Network)*. Il s'agit d'établir un canal chiffré entre deux noeuds quelconques de l'Internet, ces nœuds pouvant eux-mêmes être des routeurs d'entrée de réseaux. On aura ainsi établi une sorte de tunnel qui, à travers l'Internet, reliera deux parties éloignées l'une de l'autre du réseau d'une même entreprise pour donner l'illusion de leur contiguïté.
Le chiffrement est généralement utilisé pour les VPN de la façon suivante : l'algorithme de Diffie-Helmann est utilisé pour procéder au choix d'un secret partagé, qui constituera une clé de session pour chiffrer le trafic, et qui sera renouvelé à intervalles réguliers.
Il y a en revanche une assez grande variété de solutions pour introduire le VPN dans l'architecture du réseau :



IPSec

IPSec (Internet Protocol Security) est un protocole de sécurité basé sur des règles qui protège la transmission des données au niveau de la couche réseau. IPSec fournit l'authentification et le cryptage à la demande des communications IP entre deux points de terminaison. Les paquets IPSec sont signés, vérifiés, cryptés et décryptés au niveau de la couche de transport, ce qui rend IPSec transparent pour les applications, contrairement à SSL.

Le mode tunnel et le mode transport

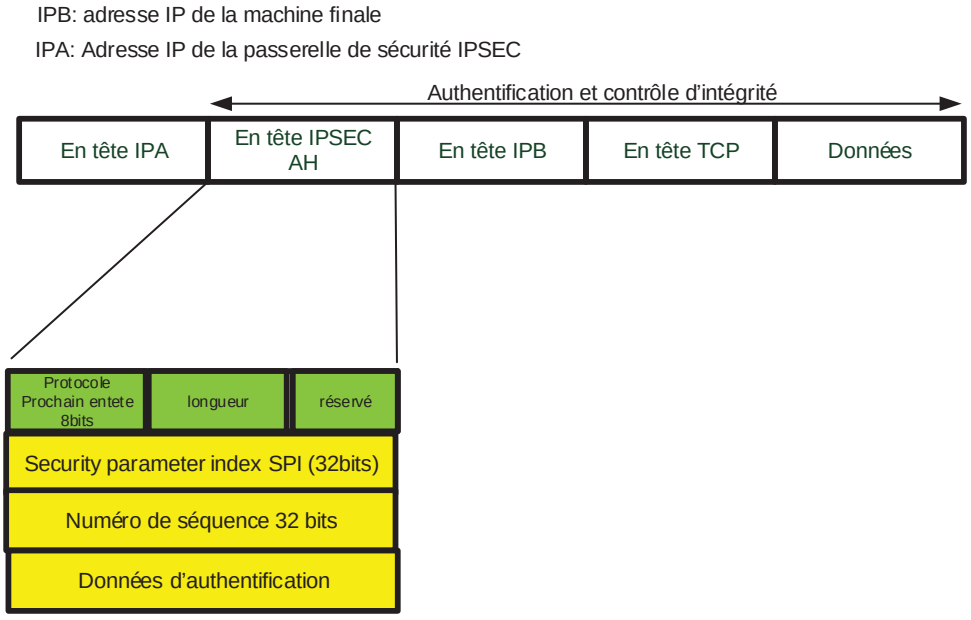


Le mode tunnel est utilisé entre deux passerelles de sécurité (routeur, firewall, …) alors que **le mode transport** se situe entre deux hôtes. **Le protocole IPSEC** est le plus utilisé pour réaliser un tunnel.
L'encapsulation IPsec en mode tunnel permet le masquage d'adresses et la sécurité dans le transport.

IPSec comporte essentiellement deux protocoles :

• **le protocole AH *(Authentification Header)*** assure l'authenticité et l'intégrité des données acheminées ; c'est un protocole réseau, de couche 3 donc, que l'on peut voir comme une option d'IP ;

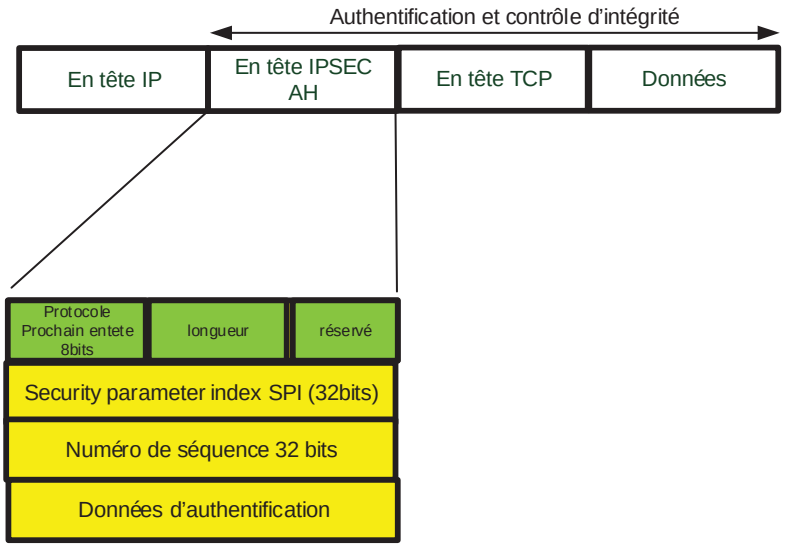
AH en mode tunnel



Le mode AH peut utiliser des algorithmes SHA-1 ou MD5.

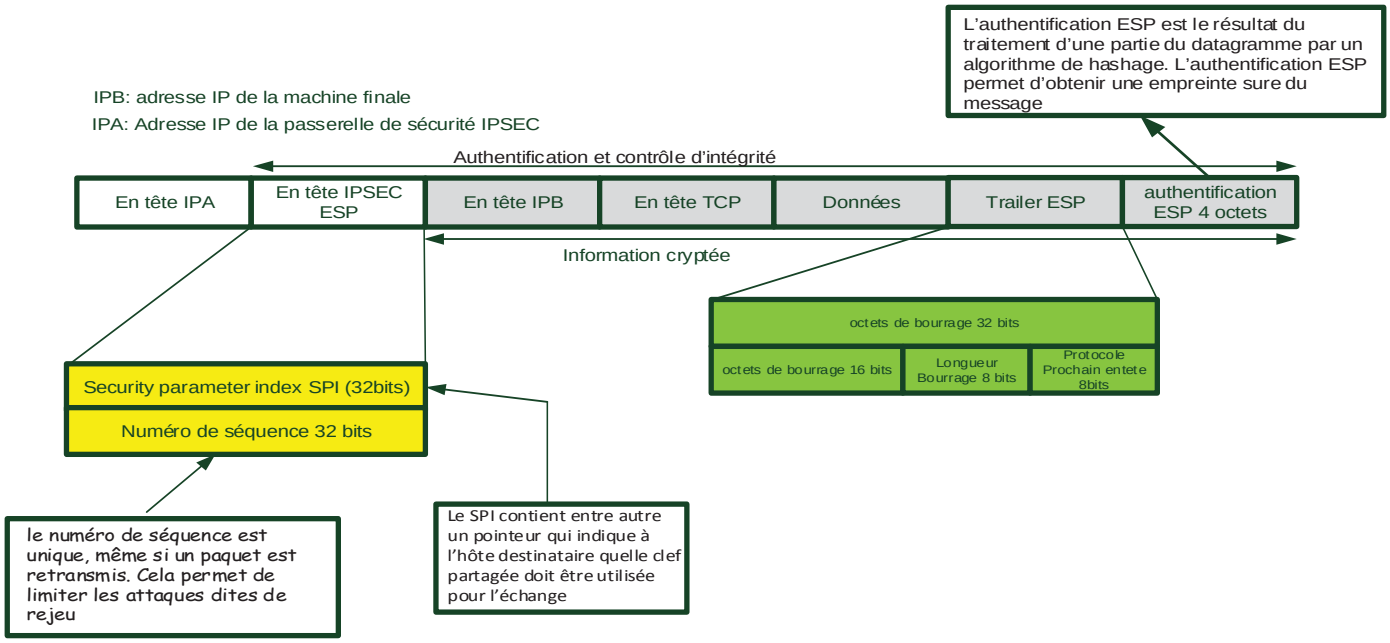
Toutefois, l'algorithme le plus couramment utilisé pour le mode IPSec AH est MD5. MD5 est également appelé *MD5-HMAC*. HMAC est l'acronyme de *Hashed Message Authentication Code (code d'authentification de message haché)* et fait référence à une implémentation de MD5 qui fournit des fonctionnalités de signature sans utiliser d'algorithme de clé publique.

IPSEC AH en mode transport

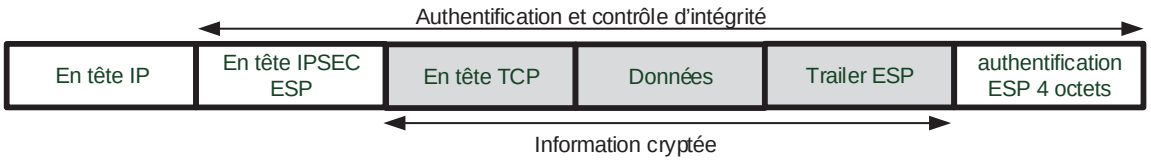


• **le protocole de transport ESP (couche 4) *(Encapsulating Security Payload)*** assure la confidentialité et l'intégrité des données, leur authenticité étant assurée de façon optionnelle.

ESP en mode tunnel



ESP en mode transport



Les protocoles AH et ESP sont complétés par le protocole d'échange de clés IKE *(Internet Key Exchange)* et par le protocole de gestion de clés ISAKMP *(Internet Security Association and Key Management Protocol),*

IKE (Internet Key Exchange)

Lors d'une communication sécurisée par *IPSec*, la source et la destination vont devoir se mettre d'accord sur un certain nombre de paramètres, et ainsi générer les clés de cryptage qui vont permettre l'échange de données de façon sécurisée. Bien évidemment, tout ce processus est effectué de façon à ce que seules la source et la destination soient capables de déchiffrer le contenu des données.

IKE est un standard de *l'IETF* permettant à deux systèmes d'établir des associations de sécurité (SA) afin de se mettre d'accord sur la façon d'échanger et de protéger les données.

IKE est lui-même un ensemble de deux protocoles : *ISAKMP* et *OAKLEY*.

La négociation porte entre autres sur les éléments suivants :

- *L'algorithme de hachage à utiliser (MD5, SHA-1) ;*
- *l'algorithme de chiffrement à utiliser (DES, 3DES) ;*
- *le groupe Diffie-Hellman à utiliser pour la génération de la clé principale (groupe 1 : 768 bits, groupe 2 : 1024 bits, groupe 3 : 2048 bits) ;*
- *l'authentification des différentes parties par : certificat, ou un secret partagé.*

Principe SSH

SSH est un moyen de connexion en ligne de commande crypté de la couche 7 du modèle OSI
SSH utilise le principe de clé asymétrique RSA (Clé privée + clé publique) pour créer une clé de session de cryptage (DES en SSH-1), principe du chiffrement mixte
La clé publique peut être librement publiée tandis que la clef privée doit rester secrète. La connaissance de la clef publique ne permet pas d'en déduire la clé privée.
Le serveur SSH (dans notre cas le switch) dispose du couple de clefs RSA stocké et généré lors de l'installation du serveur (sans certificat en SSH-1)

1/ Connexion Client

2/ Envoi clé publique du serveur au client (Vérification "fingerprint" éventuelle la première fois pour éviter un faux serveur)

3/ Génération puis crytage d'une clé de session avec clé publique

4/ Envoi clé de session cryptée au serveur

5/ Décryptage clé de session par clé privée (ce qui prouve que c'est le bon client !)

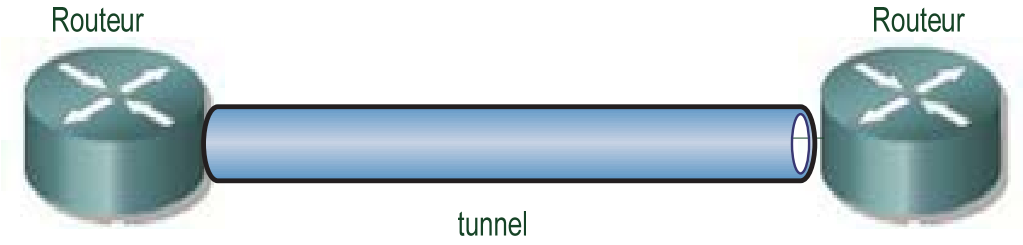
6/ Envoi d'un message du serveur crypté avec clé de session puis décryptage par le client avec la même clé (ce qui prouve que c'est le bon serveur :)

6/ Authentification du client finale en login/mot de passe



tunneling

Le tunneling est une méthode qui permet de transporter un protocole (et ses données) dans un autre protocole. Lorsque le protocole d'encapsulation permet de protéger les données qu'il transporte (d'une manière ou d'une autre, sans parler forcément de chiffrement), on préfèrera employer le terme de tunnel.



Le protocole le plus utilisé pour sécuriser TCP/IP au niveau de la couche 3 est le protocole IPSEC contrairement à SSL/TLS ou SSH qui sécurisent respectivement les couches 6 et 7 (ce qui évite de rattacher IPSec à un port donné 22 pour SSH ou 443 pour HTTPs).

Le mode ESP (Encapsulating Security Payload) du protocole IPSEC fournit les services de sécurité suivants :

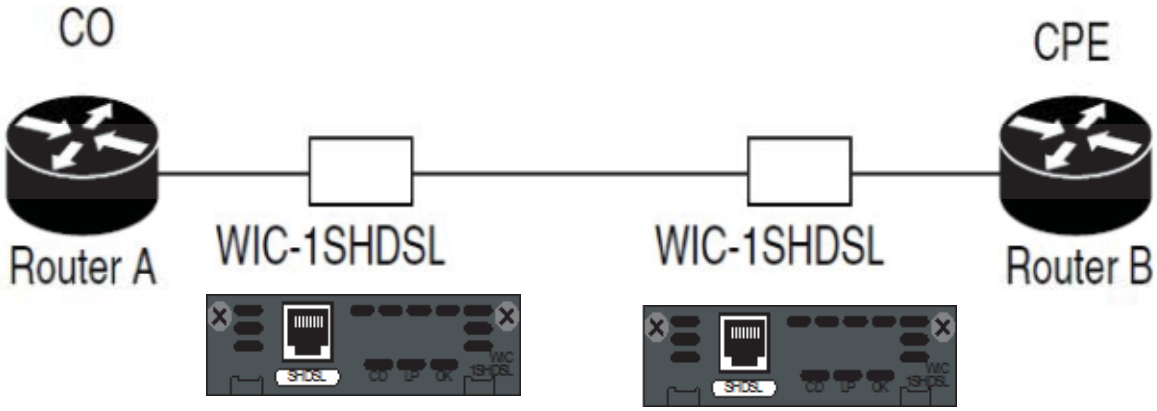
- **Confidentialité des données** : IPSec permet si on le désire de chiffrer le contenu de chaque paquet IP pour éviter la lecture de ceux-ci par quiconque. Elle est assurée par un chiffrement symétrique des données.
- **Intégrité des données** : IPSec permet de s'assurer qu'aucun paquet n'a subit de modification quelconque durant son trajet en rajoutant à chaque paquet IP le résultat d'un calcul de hachage (SHA-1 ou MD5) portant sur tout ou partie du datagramme.
- **Protection contre les écoutes et analyses de trafic** : IPsec permet de chiffrer les adresses IP réelles de la source et de la destination, ainsi que toute l'en-tête IP correspondant.
- **Protection contre le rejeu** : IPSec permet de se prémunir des attaques consistant à capturer un ou plusieurs paquets dans le but de les envoyer à nouveau pour bénéficier des mêmes avantages que l'envoyeur initial. Elle est assurée par la numérotation des paquets IP et la vérification de la séquence d'arrivée des paquets.

Puisque IPSEC est embarqué par défaut dans IPv6, c'est forcément un protocole d'avenir et déjà du présent.

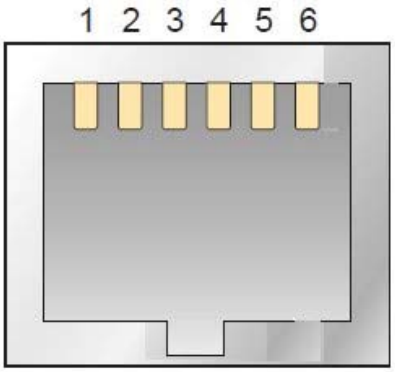
On donne page suivante un bref aperçu de ce protocole IPSEC en mode tunnel mode ESP (Encapsulated Security Payload)

La technologie SDSL

G.HDSL est une norme basée sur l'ATM avec un débit variables (jusqu'à 2.3Mbits). La technologie SDSL (symmetrical digital susbscriber line) permet des transferts de données entre un simple équipement CPE (customer premises equipment) et un bureau central CO (DSLAN). La technologie G.SHDSL est supportée par les cartes d'interfaces WAN (WIC-1SHDSL) des routeurs CISCO avec un port d'interface.
La carte WIC G.SHDSL supporte la couche d'adaptation 2 de l'ATM AAL2, la couche d'adaptation 5 de l'ATM AAL5 et les différentes classes de service ATM.



RJ-11 Pin Assignment



Pin	Signal
1	
2	
3	Line 0 tip
4	Line 0 ring
5	
6	

Tonalité ou synchro