

# Configuration et Analyse des VPNs Site-à-Site et Client-à-Site avec IPsec

Réalisé par:

**WAFAA EL MAIFI**



# Table des matières

<b>1. Introduction aux VPNs.....</b>	<b>3</b>
<b>2. Objectif du TP.....</b>	<b>3</b>
<b>3. Configuration d'un VPN Site-à-Site avec IPsec sur Cisco Packet Tracer.....</b>	<b>3</b>
<b>A. Aperçu du réseau.....</b>	<b>4</b>
<b>B. Étapes de configuration .....</b>	<b>4</b>
<b>1. Installation de Cisco Packet Tracer .....</b>	<b>4</b>
<b>2. Mise en place du schéma du lab .....</b>	<b>4</b>
<b>3. Configuration des Routeurs .....</b>	<b>5</b>
<b>4. Test de connectivité avec ping .....</b>	<b>7</b>
<b>5. Activation des modules de sécurité sur R1 et R2 .....</b>	<b>7</b>
<b>6. Configuration d'IPsec sur R1.....</b>	<b>9</b>
<b>7. Configuration d'IPsec sur Routeur R2.....</b>	<b>15</b>
<b>8. Vérification de la configuration IPsec .....</b>	<b>16</b>
<b>9. Test de connectivité entre PC0 et PC1 .....</b>	<b>17</b>
<b>10. Analyse du trafic avec un sniffer .....</b>	<b>17</b>
<b>4. Configuration d'un VPN Client-à-Site avec IPsec.....</b>	<b>18</b>
<b>A. Activation des modules de sécurité sur R0 .....</b>	<b>18</b>
<b>B. Configuration des interfaces réseau .....</b>	<b>19</b>
<b>C. Création d'un modèle d'authentification.....</b>	<b>19</b>
<b>D. Configuration d'ISAKMP et clé pré-partagée .....</b>	<b>20</b>
<b>E. Création d'un pool d'adresses IP pour les clients VPN.....</b>	<b>22</b>
<b>F. Création d'une IPsec SA .....</b>	<b>22</b>
<b>G. Configuration du VPN sur le poste client .....</b>	<b>24</b>
<b>H. Vérification du fonctionnement du VPN.....</b>	<b>25</b>

# 1. Introduction aux VPNs

Qu'est-ce qu'un VPN ?

Un VPN (Virtual Private Network) est une technologie qui crée une connexion sécurisée sur un réseau non sécurisé, comme Internet. Il chiffre les données pour protéger la confidentialité et l'intégrité des informations échangées.

Pourquoi utiliser un VPN ?

- **Confidentialité des données** : Empêche les interceptions de données par des attaquants.
- **Intégrité des données** : Garantit que les données ne sont pas altérées pendant le transfert.
- **Authentification** : Vérifie l'identité des utilisateurs et des appareils.

Différence entre VPN Site-à-Site et Client-à-Site

- **Site-à-Site VPN** : Connecte des réseaux entiers entre eux (ex : réseau du siège avec un réseau de filiale).
- **Client-à-Site VPN** : Permet à un utilisateur distant de se connecter au réseau de l'entreprise (ex : un employé en déplacement qui accède aux ressources internes).

## 2. Objectif du TP

Ce TP a pour objectif principal d'**apprendre à configurer et analyser le fonctionnement des VPNs (Virtual Private Networks) en utilisant IPsec** sur des équipements Cisco dans un environnement simulé avec Cisco Packet Tracer.

## 3. Configuration d'un VPN Site-à-Site avec IPsec sur Cisco Packet Tracer

## **A. Aperçu du réseau**

Nous allons configurer un VPN site-à-site entre :

- **Site A (Routeur R1)**
- **Site B (Routeur R2)**
- **Routeur intermédiaire (ISP - Routeur R0)**

Le schéma du réseau est le suivant :

- **Site A** : 192.168.1.0/24
- **Site B** : 192.168.2.0/24
- **Lien WAN entre R1 et R0** : 10.0.0.0/24
- **Lien WAN entre R2 et R0** : 11.0.0.0/24

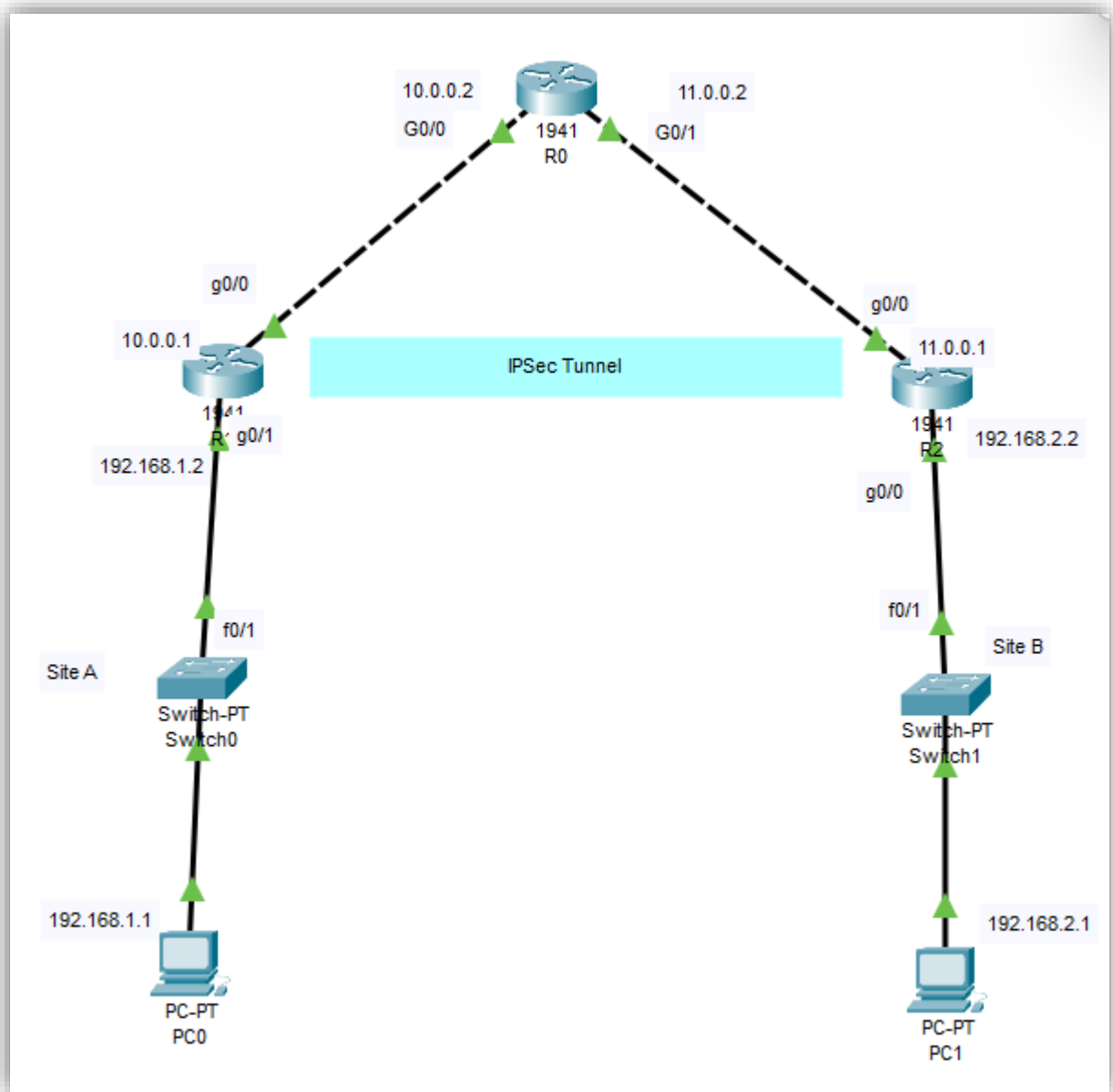
**Objectif** : Sécuriser le trafic entre les réseaux de Site A et Site B avec IPsec.

## **B. Étapes de configuration**

### **1. Installation de Cisco Packet Tracer**

Télécharge et installe Cisco Packet Tracer sur ton ordinateur pour simuler le réseau.

### **2. Mise en place du schéma du lab**



### 3. Configuration des Routeurs

#### a) Configuration du Routeur R1 (Site A)

Accède au terminal du routeur R1 et entre les commandes suivantes :

```
Router> enable
Router# configure terminal
Router(config)# hostname R1
R1(config)# int Gig0/1
R1(config-if)# ip address 192.168.1.2 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# int Gig0/0
R1(config-if)# ip address 10.0.0.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# ip route 0.0.0.0 0.0.0.0 10.0.0.2
```

*b) Configuration du Routeur R0 (ISP)*

```
Router> enable
Router# configure terminal
Router(config)# hostname R0
R0(config)# int Gig0/0
R0(config-if)# ip address 10.0.0.2 255.255.255.0
R0(config-if)# no shutdown
R0(config-if)# int Gig0/1
R0(config-if)# ip address 11.0.0.2 255.255.255.0
R0(config-if)# no shutdown
```


### c) Configuration du Routeur R2 (Site B)

```
Router> enable
Router# configure terminal
Router(config)# hostname R2
R2(config)# int Gig0/1
R2(config-if)# ip address 192.168.2.2 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# int Gig0/0
R2(config-if)# ip address 11.0.0.1 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2(config)# ip route 0.0.0.0 0.0.0.0 11.0.0.2
```

## 4. Test de connectivité avec ping

- **Vérifie la connectivité** : Utilise ping pour tester la connexion entre :
  - **R1 ↔ R2** (devrait fonctionner)
  - **PC0 ↔ PC1** (ne devrait pas fonctionner, car aucun VPN n'est configuré)

## 5. Activation des modules de sécurité sur R1 et R2

-  **securityk9** : Active les fonctionnalités de sécurité nécessaires pour IPsec.

```
R1# show license feature
R1# configure terminal
R1(config)# license boot module c1900 technology-package securityk9
R1(config)# exit
R1# reload
```

Explication des Commandes :

1. **R1# show license feature**

- **Objectif** : Cette commande permet d'afficher les fonctionnalités de licence disponibles sur le routeur.
- **Explication** :
  - Le routeur Cisco utilise des licences pour activer ou désactiver certaines fonctionnalités. Par exemple, la fonctionnalité **securityk9** est nécessaire pour activer les fonctionnalités de sécurité avancées comme **IPsec**.
  - Cette commande montre quelles fonctionnalités sont actuellement activées ou désactivées sur le routeur.
  - **Exemple de sortie** :

Feature name	Enforcement	Evaluation	Subscription	Enabled
ipbasek9	no	no	no	yes
securityk9	yes	yes	no	no
datak9	yes	no	no	no


Ici, **securityk9** est disponible mais pas encore activée (Enabled: no).

2. `R1(config)# license boot module c1900 technology-package securityk9`


- **Objectif** : Cette commande active le module de sécurité **securityk9** sur le routeur.
- **Explication** :
  - **license boot module** est utilisé pour activer un module de licence spécifique sur le routeur.
  - **c1900** fait référence au modèle du routeur (dans ce cas, un routeur de la série 1900).
  - **technology-package securityk9** spécifie que vous souhaitez activer le package de sécurité (**securityk9**), qui inclut des fonctionnalités comme IPsec, VPN, etc.
  - **Pourquoi est-ce nécessaire ?**
    - Sans cette licence, le routeur ne peut pas utiliser les fonctionnalités de sécurité avancées nécessaires pour configurer un VPN IPsec.




## 6. Configuration d'IPsec sur R1

 **ISAKMP Policy** : Définit les paramètres de négociation de la sécurité.

```
R1(config)# crypto isakmp policy 10
R1(config-isakmp)# encryption aes 256
R1(config-isakmp)# hash sha
R1(config-isakmp)# authentication pre-share
R1(config-isakmp)# group 2
R1(config-isakmp)# lifetime 86400
```

 **Clé pré-partagée** pour l'authentification :

```
R1(config)# crypto isakmp key vpnuser address 11.0.0.1
```

 **IPsec Transform Set** pour le chiffrement des données :

```
R1(config)# crypto ipsec transform-set myset esp-aes esp-sha-hmac
```

 **ACL pour le trafic chiffré** :

```
R1(config)# access-list 100 permit ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255
```

 **Crypto Map** :

```
R1(config)# crypto map mymap 10 ipsec-isakmp
R1(config-crypto-map)# set peer 11.0.0.1
R1(config-crypto-map)# set transform-set myset
R1(config-crypto-map)# match address 100
```

 **Application à l'interface externe** :

```
R1(config)# int Gig0/0
R1(config-if)# crypto map mymap
```

**Explication des Commandes:**

### 1. `R1(config)# crypto isakmp policy 10`

- **Objectif** : Créer une politique ISAKMP avec un numéro de priorité (dans ce cas, 10).
- **Explication** :
  - **crypto isakmp policy** est utilisé pour définir une politique ISAKMP.
  - Le numéro 10 est un identifiant de priorité. Plus le numéro est bas, plus la politique a une priorité élevée. Si plusieurs politiques sont configurées, celle avec le numéro le plus bas sera utilisée en premier.

### 2. `R1(config-isakmp)# encryption aes 256`

- **Objectif** : Définir l'algorithme de chiffrement utilisé pour sécuriser les communications.
- **Explication** :
  - **encryption** spécifie l'algorithme de chiffrement à utiliser.
  - **aes 256** indique que l'algorithme AES (Advanced Encryption Standard) avec une clé de 256 bits sera utilisé.
  - **Pourquoi AES 256 ?**
    - AES est un algorithme de chiffrement symétrique très sécurisé et largement utilisé. Une clé de 256 bits offre un niveau de sécurité élevé.

### 3. `R1(config-isakmp)# hash sha`

- **Objectif** : Définir l'algorithme de hachage utilisé pour l'intégrité des données.
- **Explication** :
  - **hash** spécifie l'algorithme de hachage à utiliser pour vérifier l'intégrité des données.
  - **sha** indique que l'algorithme SHA (Secure Hash Algorithm) sera utilisé. Par défaut, il s'agit de SHA-1, mais des versions plus récentes comme SHA-256 peuvent être utilisées.
  - **Pourquoi SHA ?**
    - SHA est un algorithme de hachage sécurisé qui garantit que les données n'ont pas été altérées pendant la transmission.

#### 4. `R1(config-isakmp)# authentication pre-share`

- **Objectif** : Définir la méthode d'authentification utilisée pour établir la communication sécurisée.
- **Explication** :
  - authentication spécifie la méthode d'authentification.
  - pre-share indique que l'authentification se fera à l'aide d'une **clé pré-partagée** (pre-shared key, PSK).
  - **Pourquoi une clé pré-partagée ?**
    - Une clé pré-partagée est une méthode simple et couramment utilisée pour l'authentification dans les VPN. Les deux pairs (routeurs) doivent partager la même clé pour s'authentifier.

#### 5. `R1(config-isakmp)# group 2`

- **Objectif** : Définir le groupe Diffie-Hellman (DH) à utiliser pour l'échange de clés.
- **Explication** :
  - **group** spécifie le groupe Diffie-Hellman à utiliser.
  - 2 indique que le groupe Diffie-Hellman de taille 1024 bits (DH Group 2) sera utilisé.
  - **Pourquoi Diffie-Hellman ?**
    - L'algorithme Diffie-Hellman permet à deux parties de générer une clé secrète partagée sur un canal non sécurisé. Cela est essentiel pour établir une communication sécurisée.
  - **Groupes disponibles** :
    - **Group 1** : 768 bits (moins sécurisé).
    - **Group 2** : 1024 bits (sécurité moyenne).
    - **Group 5** : 1536 bits (plus sécurisé).

#### 6. `R1(config-isakmp)# lifetime 86400`

- **Objectif** : Définir la durée de vie de l'association de sécurité (SA).
- **Explication** :
  - **lifetime** spécifie la durée de vie de l'association de sécurité en secondes.
  - 86400 indique que la SA expirera après 86 400 secondes (soit 24 heures).
  - **Pourquoi définir une durée de vie ?**
    - La durée de vie d'une SA garantit que les clés de chiffrement sont régulièrement renouvelées, ce qui améliore la sécurité. Après expiration, une nouvelle SA est négociée.

7. `crypto isakmp key vpnuser address 11.0.0.1`

- **Objectif** : Configurer une clé pré-partagée pour l'authentification ISAKMP.
- **Explication** :
  - **crypto isakmp key** est la commande utilisée pour définir une clé pré-partagée.
  - Cette clé est utilisée pour authentifier les deux routeurs (pairs) qui établissent une connexion sécurisée via IPsec.
  - **vpnuser** est la clé pré-partagée (mot de passe) qui sera utilisée pour l'authentification.
  - Cette clé doit être identique sur les deux routeurs (pairs) qui établissent la connexion VPN.
  - **address 11.0.0.1** indique que la clé pré-partagée vpnuser sera utilisée pour authentifier le routeur dont l'adresse IP est 11.0.0.1.

8. `crypto ipsec transform-set myset esp-aes esp-sha-hmac`

- **Objectif** : Créer un ensemble de transformations (transform-set) pour IPsec.
- **Explication** :
  - **crypto ipsec transform-set** est la commande utilisée pour définir un ensemble de transformations.
  - Un transform-set regroupe les algorithmes de chiffrement et d'authentification qui seront utilisés pour sécuriser les données dans le tunnel VPN.

- **myset** est le nom donné à cet ensemble de transformations. Vous pouvez choisir n'importe quel nom, mais il doit être unique pour chaque transform-set sur le routeur.
- **esp-aes** spécifie que l'algorithme de chiffrement AES (Advanced Encryption Standard) sera utilisé dans le cadre du protocole ESP (Encapsulating Security Payload).
- Options disponibles :
  - **esp-aes** : Utilise AES avec une clé de 128 bits par défaut.
  - **esp-aes 192** : Utilise AES avec une clé de 192 bits.
  - **esp-aes 256** : Utilise AES avec une clé de 256 bits (le plus sécurisé).
- Dans cette commande, esp-aes utilise AES avec une clé de 128 bits par défaut.
- **esp-sha-hmac** spécifie que l'algorithme de hachage SHA (Secure Hash Algorithm) sera utilisé pour l'authentification HMAC (Hash-based Message Authentication Code) dans le cadre du protocole ESP.
- Options disponibles :
  - **esp-md5-hmac** : Utilise MD5 pour l'authentification HMAC (moins sécurisé).
  - **esp-sha-hmac** : Utilise SHA-1 pour l'authentification HMAC.
  - **esp-sha256-hmac** : Utilise SHA-256 pour l'authentification HMAC (plus sécurisé).

9. `access-list 100 permit ip 192.168.1.0 0.0.0.255 192.168.2.0`

- **Objectif** : La commande est utilisée pour créer une liste de contrôle d'accès (ACL) sur un routeur . Cette ACL définit le trafic réseau qui sera autorisé à passer à travers le VPN IPsec. Dans ce cas, elle permet au trafic entre les réseaux 192.168.1.0/24 et 192.168.2.0/24 d'être chiffré et transmis via le tunnel VPN.
- **Explication** :
  - **access-list** est la commande utilisée pour créer ou modifier une ACL.

- Le numéro **100** est l'identifiant de l'ACL. Les ACLs standard utilisent des numéros entre 1 et 99, tandis que les ACLs étendues utilisent des numéros entre 100 et 199.
- **permit** est utilisé pour autoriser le trafic qui correspond aux conditions définies dans l'ACL.

#### 10. `R1(config)# crypto map mymap 10 ipsec-isakmp`

- **Objectif** : Créer une crypto map avec un nom et un numéro de séquence.
- **Explication** :
  - **crypto map** est la commande utilisée pour créer ou modifier une crypto map.
  - **mymap** est le nom de la crypto map. Vous pouvez choisir n'importe quel nom, mais il doit être unique pour chaque crypto map sur le routeur.
  - **10** est le numéro de séquence de la crypto map. Si vous avez plusieurs entrées dans la même crypto map, vous pouvez utiliser des numéros de séquence pour les organiser (par exemple, 10, 20, 30, etc.).
  - **ipsec-isakmp** indique que cette crypto map utilisera IPsec avec ISAKMP pour établir le tunnel VPN.
  - **Pourquoi une crypto map ?**
    - Une crypto map regroupe tous les éléments nécessaires pour configurer un VPN IPsec, tels que les transform-sets, les pairs, et les ACLs. Elle est ensuite appliquée à une interface pour activer le VPN.

#### 11. `R1(config-crypto-map)# set peer 11.0.0.1`

- **Objectif** : Spécifier l'adresse IP du pair (routeur distant) avec lequel le tunnel VPN sera établi.
- **Explication** :
  - **set peer** est utilisé pour définir l'adresse IP du pair (routeur distant) avec lequel le tunnel VPN sera établi.
  - **11.0.0.1** est l'adresse IP du pair (dans ce cas, R2).

12. `R1(config-crypto-map)# set transform-set myset`

- **Objectif** : Associer un transform-set à la crypto map.
- **Explication** :
  - **set transform-set** est utilisé pour spécifier l'ensemble de transformations (transform-set) qui sera utilisé pour chiffrer et authentifier le trafic VPN.
  - **myset** est le nom du transform-set que vous avez précédemment configuré.

13. `R1(config-crypto-map)# match address 100`

- **Objectif** : Associer une ACL à la crypto map pour identifier le trafic à chiffrer.
- **Explication** :
  - **match address** est utilisé pour spécifier l'ACL qui définit le trafic à chiffrer et à transmettre via le tunnel VPN.
  - **100** est le numéro de l'ACL que vous avez précédemment configurée.
  - Cette commande associe l'ACL 100 à la crypto map, ce qui signifie que le trafic correspondant à cette ACL sera chiffré et transmis via le tunnel VPN.

14. `R1(config-if)# crypto map mymap`

- **Objectif** : Appliquer la crypto map **mymap** à l'interface **GigabitEthernet0/0**.
- **Explication** :
  - **crypto map mymap** est la commande utilisée pour appliquer une crypto map à une interface.
  - **mymap** est le nom de la crypto map que vous avez précédemment configurée.

## 7. Configuration d'IPsec sur Routeur R2

La configuration sur R2 est très similaire à celle de R1, mais avec quelques différences dans les adresses IP et les ACL. Voici les étapes à suivre :

-  **ISAKMP Policy** : Définir les paramètres de négociation de sécurité pour IPsec.

```
R2(config)# crypto isakmp policy 11
R2(config-isakmp)# encryption aes 256
R2(config-isakmp)# hash sha
R2(config-isakmp)# authentication pre-share
R2(config-isakmp)# group 2
R2(config-isakmp)# lifetime 86400
```

✚ **Clé pré-partagée** : Doit être la même que celle de R1 pour établir la connexion sécurisée.

```
R2(config)# crypto isakmp key vpnuser address 10.0.0.1
```

✚ **IPsec Transform Set** : Spécifie le chiffrement et l'authentification des données.

```
R2(config)# crypto ipsec transform-set myset esp-aes esp-sha-hmac
```

✚ **ACL pour définir le trafic à chiffrer** :

```
R2(config)# access-list 110 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
```

✚ **Crypto Map** : Pour regrouper toutes les configurations précédentes.

```
R2(config)# crypto map mymap 11 ipsec-isakmp
R2(config-crypto-map)# set peer 10.0.0.1
R2(config-crypto-map)# set transform-set myset
R2(config-crypto-map)# match address 110
```

✚ **Application de la Crypto Map à l'interface externe**

```
R2(config)# int Gig0/0
R2(config-if)# crypto map mymap
```

## 8. Vérification de la configuration IPsec

Après avoir configuré IPsec sur R1 et R2, il est important de vérifier que le tunnel VPN fonctionne correctement. Utilisez les commandes suivantes :



### Vérifier l'état de l'association ISAKMP :

```
R1# show crypto isakmp sa
R2# show crypto isakmp sa
```

### Vérifier l'état de l'association IPsec :

```
R1# show crypto ipsec sa
R2# show crypto ipsec sa
```

Si tout fonctionne correctement, tu devrais voir l'état **QM\_IDLE** avec le statut **ACTIVE**, indiquant que le tunnel VPN est établi et opérationnel.

## 9. Test de connectivité entre PC0 et PC1

Après avoir configuré le VPN site-à-site, fais un ping entre les PC dans les réseaux des deux sites :

- Depuis **PC0 (Site A)** → **PC1 (Site B)**
- Depuis **PC1 (Site B)** → **PC0 (Site A)**

Si le VPN fonctionne correctement, le ping devrait réussir.

Si le VPN ne fonctionne pas, active les **logs de débogage** pour voir où ça bloque :

```
R1# debug crypto isakmp
R1# debug crypto ipsec
```

Fais un ping entre les réseaux protégés et observe les erreurs qui s'affichent.

## 10. Analyse du trafic avec un sniffer

Pour vérifier que le trafic est bien chiffré :

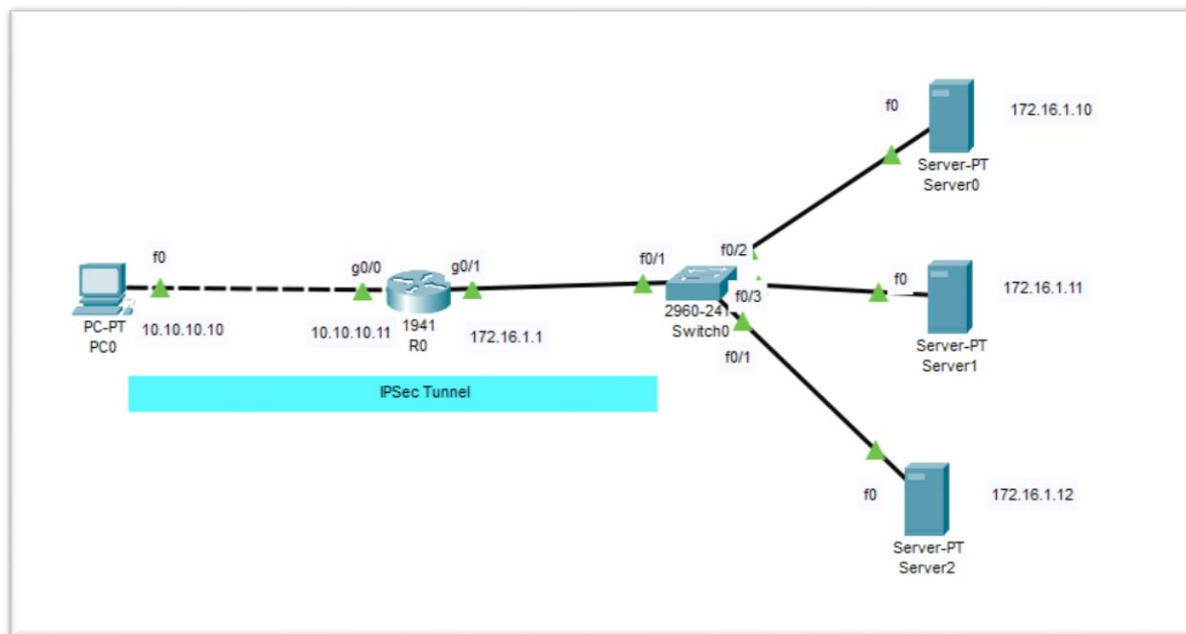
- Place un **sniffer** entre le Routeur R1 et le Routeur ISP (R0).

- Active le **mode simulation** dans Cisco Packet Tracer.
- Fais un ping entre PC0 et PC1 et observe les paquets qui transitent.
- Les paquets doivent être chiffrés (pas de données en clair visibles).

## 4. Configuration d'un VPN Client-à-Site avec IPsec

Un VPN Client-à-Site permet à un utilisateur distant de se connecter au réseau interne de l'entreprise via un tunnel sécurisé.

Dans ce TP, le client se connecte au **Routeur R0** qui agit comme un serveur VPN.



### A. Activation des modules de sécurité sur R0

Comme pour R1 et R2, il faut activer les modules de sécurité :

```
R0# show license feature
R0# configure terminal
R0(config)# license boot module c1900 technology-package securityk9
R0(config)# exit
R0# reload
```

## B. Configuration des interfaces réseau

Configure les interfaces réseau sur R0 pour gérer le trafic VPN :

```
R0(config)# int Gig0/0
R0(config-if)# ip address 10.10.10.11 255.255.255.0
R0(config-if)# no shutdown
R0(config-if)# int Gig0/1
R0(config-if)# ip address 172.16.1.1 255.255.255.0
R0(config-if)# no shutdown
```

## C. Création d'un modèle d'authentification

Pour authentifier les utilisateurs distants qui se connectent au VPN :

```
R0(config)# aaa new-model
R0(config)# aaa authentication login list1 local
R0(config)# aaa authorization network list2 local
R0(config)# username admin password admin
```

Explication des Commandes :

### 1. `aaa new-model`

- **Objectif** : Activer le modèle AAA sur le routeur.
- **Explication** :
  - **aaa new-model** est la commande utilisée pour activer le framework AAA sur le routeur.
  - **Pourquoi activer AAA ?**
    - AAA permet de centraliser la gestion de l'authentification, de l'autorisation et de la comptabilité (Accounting) des utilisateurs. Cela améliore la sécurité et la gestion des accès au routeur.

## 2. `R0(config)# aaa authentication login list1 local`

- **Objectif** : Configurer une méthode d'authentification pour les connexions au routeur.
- **Explication** :
  - **aaa authentication login** est la commande utilisée pour configurer l'authentification des utilisateurs qui se connectent au routeur (par exemple, via SSH, Telnet, ou la console).
  - **list1** est le nom de la liste d'authentification. Vous pouvez créer plusieurs listes avec des noms différents pour appliquer des méthodes d'authentification différentes à différents types de connexions.
  - **local** indique que l'authentification sera effectuée localement, c'est-à-dire en utilisant la base de données d'utilisateurs locale du routeur.

## 3. `R0(config)# aaa authorization network list2 local`

- **Objectif** : Configurer une méthode d'autorisation pour les connexions réseau (par exemple, les connexions VPN).
- **Explication** :
  - **aaa authorization network** est la commande utilisée pour configurer l'autorisation des utilisateurs qui accèdent à des services réseau, tels que les connexions VPN.
  - **list2** est le nom de la liste d'autorisation. Vous pouvez créer plusieurs listes avec des noms différents pour appliquer des méthodes d'autorisation différentes à différents services.

## 4. `R0(config)# username admin password admin`

- **Objectif** : Créer un utilisateur local avec un nom d'utilisateur et un mot de passe.

## D. Configuration d'ISAKMP et clé pré-partagée

-  **ISAKMP Policy** : Définit les paramètres de négociation de sécurité pour IPsec.

```
R0(config)# crypto isakmp policy 10
R0(config-isakmp)# encryption aes
R0(config-isakmp)# hash sha
R0(config-isakmp)# authentication pre-share
R0(config-isakmp)# group 2
```

🚦 **Clé pré-partagée** pour l'authentification :

```
R0(config)# crypto isakmp client configuration group ccn
R0(config-isakmp-group)# key ccn123
R0(config-isakmp-group)# pool VPNPOOL
```

Explication des Commandes :

1. **R0(config)# crypto isakmp client configuration group ccn**

- **Objectif** : Créer un groupe de clients VPN avec un nom spécifique.
- **Explication** :
  - **crypto isakmp client configuration group** est la commande utilisée pour créer un groupe de clients VPN.
  - **ccn** est le nom du groupe. Vous pouvez choisir n'importe quel nom, mais il doit être unique pour chaque groupe sur le routeur.

2. **R0(config-isakmp-group)# key ccn123**

- **Objectif** : Définir une clé pré-partagée pour l'authentification des clients VPN.

3. **R0(config-isakmp-group)# pool VPNPOOL**

- **Objectif** : Spécifier un pool d'adresses IP qui sera attribué aux clients VPN.

## E. Création d'un pool d'adresses IP pour les clients VPN

Définit une plage d'adresses IP à attribuer dynamiquement aux clients VPN :

```
R0(config)# ip local pool VPNPOOL 192.168.1.1 192.168.1.50
```

## F. Création d'une IPsec SA

🚦 **IPsec Transform Set** : Définit le chiffrement et l'authentification des données.

```
R0(config)# crypto ipsec transform-set set1 esp-aes esp-sha-hmac
```

🚦 **Crypto Map** :

```
R0(config)# crypto dynamic-map map1 10
R0(config-crypto-map)# set transform-set set1
R0(config)# crypto map map2 client configuration address respond
R0(config)# crypto map map2 client authentication list list1
R0(config)# crypto map map2 isakmp authorization list list2
R0(config)# crypto map map2 10 ipsec-isakmp dynamic map1
```

Explication des commandes :

1. `R0(config)# crypto dynamic-map map1 10`

- **Objectif** : Créer une crypto map dynamique avec un nom et un numéro de séquence.
- **Explication** :
  - **crypto dynamic-map** est la commande utilisée pour créer une crypto map dynamique.
  - **map1** est le nom de la crypto map dynamique. Vous pouvez choisir n'importe quel nom, mais il doit être unique pour chaque crypto map sur le routeur.

- **10** est le numéro de séquence de la crypto map dynamique.
- **Pourquoi une crypto map dynamique ?**
  - Une crypto map dynamique est utilisée pour gérer les connexions VPN avec des clients dont les adresses IP ne sont pas connues à l'avance (par exemple, les clients VPN distants). Elle permet de créer des associations de sécurité (SA) dynamiques.

## 2. `R0(config)# crypto map map2 client configuration address respond`

- **Objectif :** Configurer la crypto map pour répondre aux demandes de configuration d'adresse IP des clients VPN.
- **Explication :**
  - `crypto map map2 client configuration address respond` est la commande utilisée pour indiquer que le routeur doit répondre aux demandes de configuration d'adresse IP des clients VPN.
  - **Pourquoi cette commande ?**
    - Lorsqu'un client VPN se connecte, il peut demander une adresse IP au routeur. Cette commande permet au routeur de répondre à cette demande en attribuant une adresse IP à partir du pool configuré.
  - Cette commande configure la crypto map map2 pour répondre aux demandes de configuration d'adresse IP des clients VPN.

## 3. `R0(config)# crypto map map2 client authentication list list1`

- **Objectif :** Associer une liste d'authentification à la crypto map pour authentifier les clients VPN.
- **Explication :**
  - `crypto map map2 client authentication list list1` est la commande utilisée pour spécifier la liste d'authentification qui sera utilisée pour authentifier les clients VPN.

#### 4. `R0(config)# crypto map map2 isakmp authorization list list2`

- **Objectif** : Associer une liste d'autorisation à la crypto map pour autoriser les clients VPN.
- **Explication** :
  - **crypto map map2 isakmp** authorization list list2 la commande utilisée pour spécifier la liste list2 d'autorisation qui sera utilisée pour autoriser les clients VPN.

#### 5. `R0(config)# crypto map map2 10 ipsec-isakmp dynamic map1`

- **Objectif** : la commande utilisée pour associer une crypto map dynamique (map1) à une crypto map statique (map2).

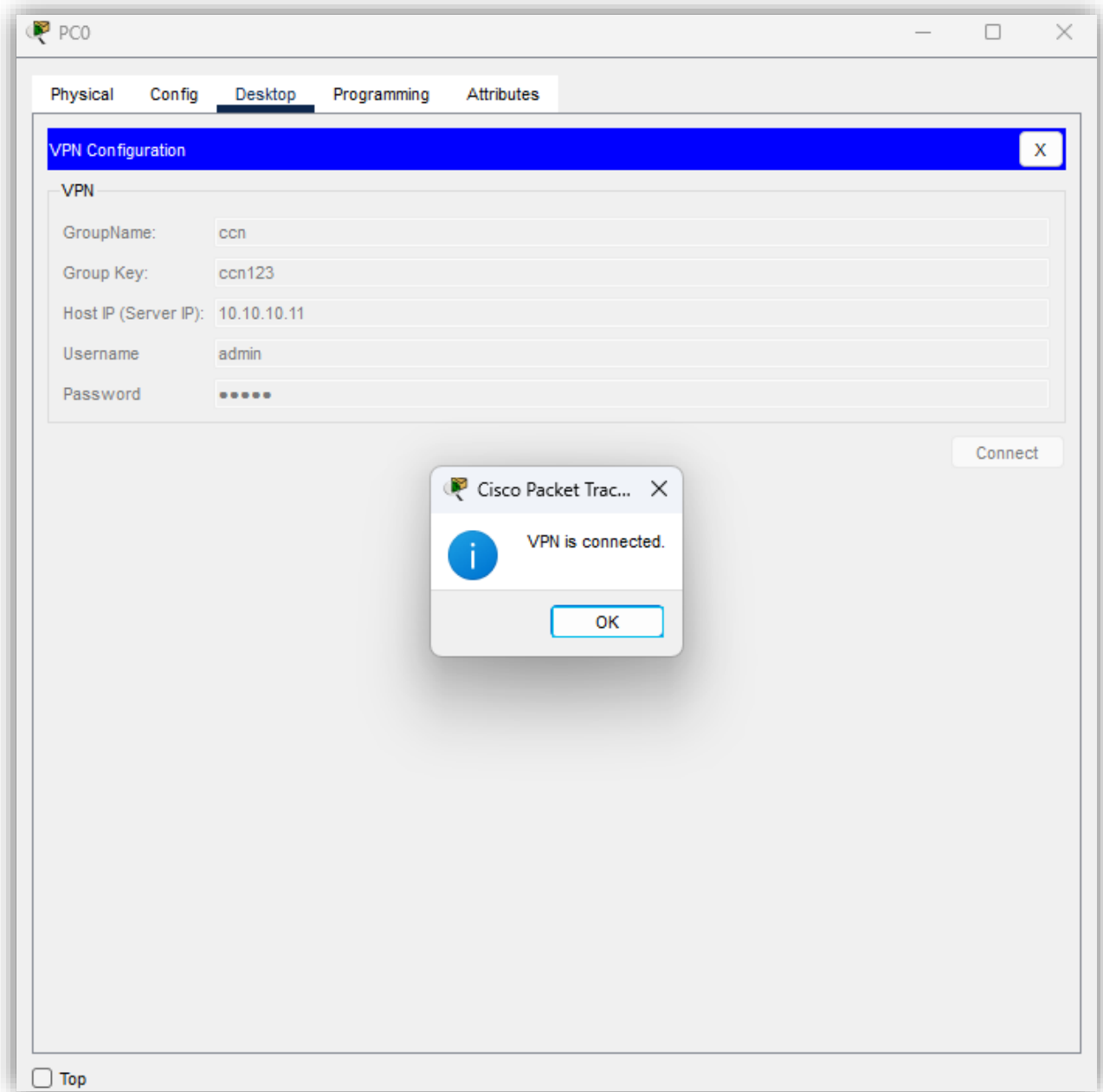
 **Application à l'interface externe :**

```
R0(config)# int Gig0/0
R0(config-if)# crypto map map2
```

## G. Configuration du VPN sur le poste client

- Utilise un **PC Client** dans Cisco Packet Tracer.
- Configure-le pour utiliser les informations suivantes :
  - **Adresse du serveur VPN** : 10.10.10.11
  - **Nom du groupe** : ccn
  - **Clé pré-partagée** : ccn123
  - **Nom d'utilisateur** : admin
  - **Mot de passe** : admin





## H. Vérification du fonctionnement du VPN

- Depuis le client, fais un ping vers le réseau interne (ex : 192.168.1.0).
- Sur le **Routeur R0**, vérifie l'état du tunnel :

```
R0# show crypto isakmp sa
```

```
R0# show crypto ipsec sa
```

```
R0# show crypto isakmp sa
```

```
IPv4 Crypto ISAKMP SA
dst          src          state          conn-id slot status
10.10.10.10  10.10.10.11  QM_IDLE       1097      0  ACTIVE
```

```
R0#show crypto ipsec sa
```

```
interface: GigabitEthernet0/0
  Crypto map tag: map2, local addr 10.10.10.11

protected vrf: (none)
local ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
remote ident (addr/mask/prot/port): (192.168.1.1/255.255.255.255/0/0)
current_peer 10.10.10.10 port 500
  PERMIT, flags={origin_is_acl,}
#pkts encaps: 8, #pkts encrypt: 8, #pkts digest: 0
#pkts decaps: 8, #pkts decrypt: 8, #pkts verify: 0
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 0, #recv errors 0

local crypto endpt.: 10.10.10.11, remote crypto endpt.:10.10.10.10
path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/0
current outbound spi: 0x25716F8B(628191115)

inbound esp sas:
  spi: 0x64F272E0(1693610720)
    transform: esp-aes esp-sha-hmac ,
    in use settings ={Tunnel, }
    conn id: 2000, flow_id: FPGA:1, crypto map: map2
    sa timing: remaining key lifetime (k/sec): (4525504/1537)
    IV size: 16 bytes
    replay detection support: N
    Status: ACTIVE

inbound ah sas:

inbound pcsp sas:

outbound esp sas:
  spi: 0x25716F8B(628191115)
    transform: esp-aes esp-sha-hmac ,
    in use settings ={Tunnel, }
    conn id: 2001, flow_id: FPGA:1, crypto map: map2
    sa timing: remaining key lifetime (k/sec): (4525504/1537)
    IV size: 16 bytes
    replay detection support: N
    Status: ACTIVE

outbound ah sas:

outbound pcsp sas:
```