Teaching-HEIGVD-SRX-2020-Laboratoire-VPN

Ce travail de laboratoire est à faire en équipes de 3 personnes

Pour ce travail de laboratoire, il est votre responsabilité de chercher vous-même sur internet, le support du cours ou toute autre source (vous avez aussi le droit de communiquer avec les autres équipes), toute information relative au sujet VPN, le logiciel eve-ng, les routeur Cisco, etc que vous ne comprenez pas !

ATTENTION : Commencez par créer un Fork de ce repo et travaillez sur votre fork.

Clonez le repo sur votre machine. Vous pouvez répondre aux questions en modifiant directement votre clone du README.md ou avec un fichier pdf que vous pourrez uploader sur votre fork.

Le rendu consiste simplement à répondre à toutes les questions clairement identifiées dans le text avec la mention "Question" et à les accompagner avec des captures. Le rendu doit se faire par une "pull request". Envoyer également le hash du dernier commit et votre username GitHub par email au professeur et à l'assistant

N'oubliez pas de spécifier les noms des membres du groupes dans la Pull Request ainsi que dans le mail de rendu !!!

Echéance

Ce travail devra être rendu le dimanche après la fin de la 2ème séance de laboratoire, soit au plus tard, **le 11 mai 2020, à 23h59.**

Introduction

Dans ce travail de laboratoire, vous allez configurer des routeurs Cisco émulés, afin de mettre en œuvre une infrastructure sécurisée utilisant des tunnels IPSec.

Les aspects abordés

- Contrôle de fonctionnement de l'infrastructure
- Contrôle du DHCP serveur hébergé sur le routeur
- Gestion des routeurs en console
- Capture Sniffer avec filtres précis sur la communication à épier
- Activation du mode « debug » pour certaines fonctions du routeur
- Observation des protocoles IPSec

Matériel

La manière la plus simple de faire ce laboratoire est dans les machines des salles de labo. Le logiciel d'émulation c'est eve-ng. Vous trouverez un guide très condensé pour l'utilisation de eve-ng ici.

Vous pouvez faire fonctionner ce labo sur vos propres machines à condition de copier la VM eve-ng. A présent, la manière la plus simple d'utiliser eve-ng est de l'installer sur Windows (mais, il est possible de le faire fonctionner sur Mac OS et sur Linux...).

Tuto d'installation de la VM eve-ng : https://www.eve-ng.net/index.php/documentation/installation/virtual-machine-install/

Récupération de la VM pré-configurée (vous ne pouvez pas utiliser la versión qui se trouve sur le site de eve-ng) : vous la trouverez sur \eistore1\cours\iict\SRX\LaboVPn

Il est conseillé de passer la VM en mode "Bridge" si vous avez des problèmes. Le mode NAT **devrait** aussi fonctionner.

Les user-password en mode terminal sont : "root" | "eve"

Les user-password en mode navigateur sont : "admin" | "eve"

Ensuite, terminez la configuration de la VM, connectez vous et récupérez l'adresse ip de la machine virtuelle.

Utilisez un navigateur internet (hors VM) et tapez l'adresse IP de la VM.

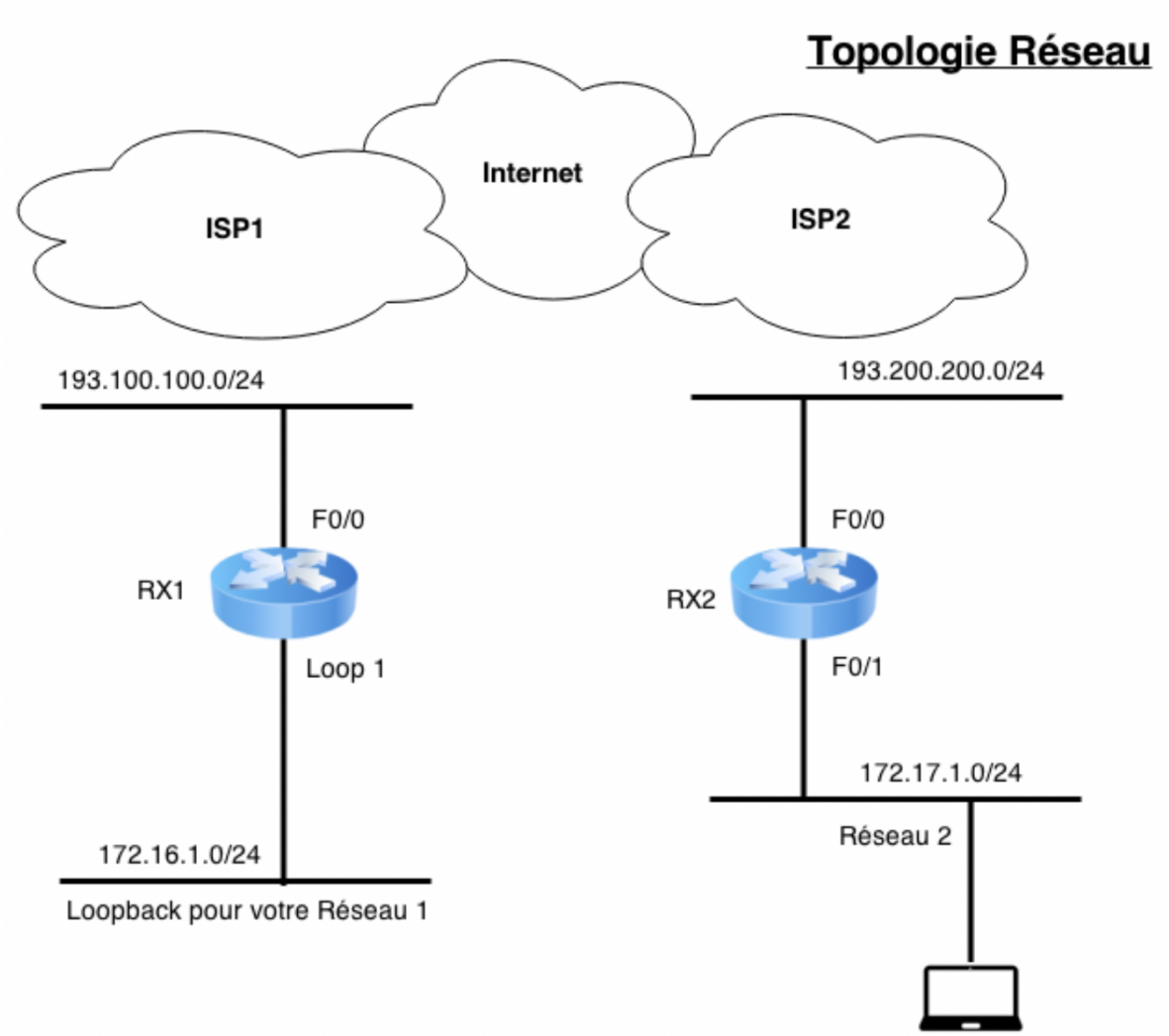
Fichiers nécessaires

Tout ce qu'il vous faut c'est un fichier de projet eve-ng, que vous pourrez importer directement dans votre environnement de travail.

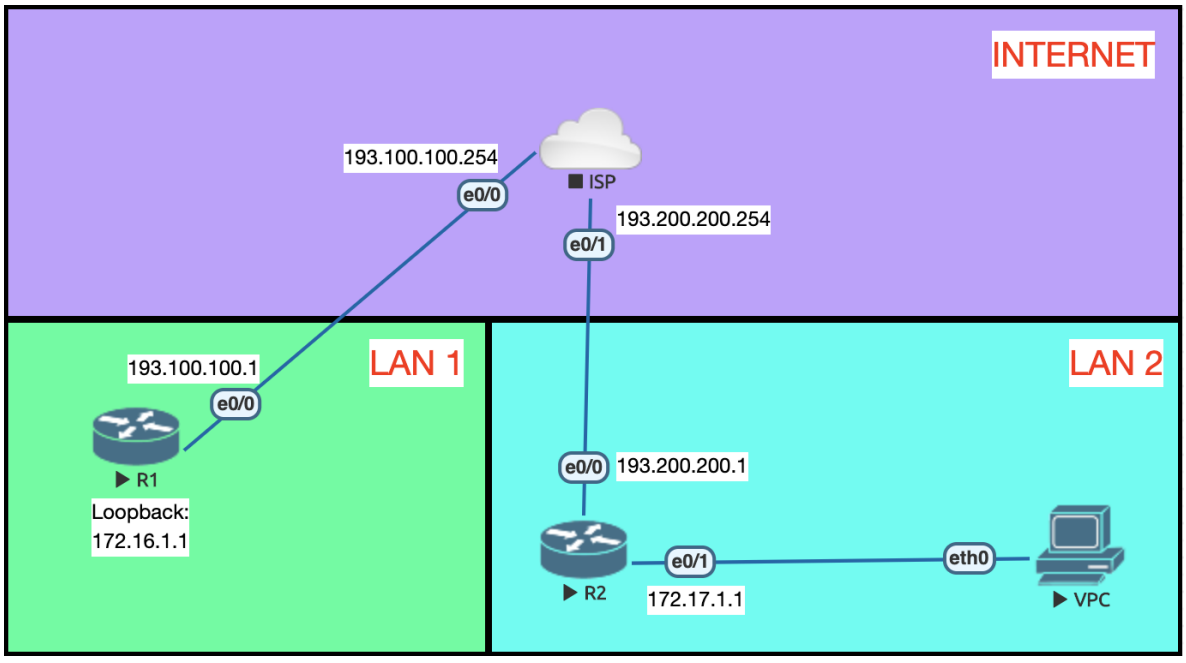
Mise en place

Voici la topologie qui sera simulée. Elle comprend deux routeurs interconnectés par l'Internet. Les deux réseaux LAN utilisent les services du tunnel IPSec établi entre les deux routeurs pour communiquer.

Les "machines" du LAN1 (connecté au ISP1) sont simulées avec l'interface loopback du routeur. Les "machines" du LAN2 sont représentées par un seul ordinateur.



Voici le projet eve-ng utilisé pour implémenter la topologie. Le réseau Internet (nuage) est simulé par un routeur.



Manipulations

- Commencer par importer le projet dans eve-ng.

- Prenez un peu de temps pour vous familiariser avec la topologie présentée dans ce guide et comparez-la au projet eve-ng. Identifiez les éléments, les interconnexions et les adresses IPs.
- À tout moment, il vous est possible de sauvegarder la configuration dans la mémoire de vos routeurs :
- Au Shell privilégié (symbole #) entrer la commande suivante pour sauvegarder la configuration actuelle dans la mémoire nvram du routeur : `wr`
    - Vous **devez** faire des sauvegardes de la configuration (exporter) dans un fichier - c.f. document guide eve-ng

Vérification de la configuration de base des routeurs

Objectifs:

Vérifier que le projet a été importé correctement. Pour cela, nous allons contrôler certains paramètres :

- Etat des interfaces (`show interface`)
- Connectivité (`ping` , `show arp`)
- Contrôle du DHCP serveur hébergé sur R2

A faire...

- Contrôlez l'état de toutes vos interfaces dans les deux routeurs et le routeur qui simule l'Internet - Pour contrôler l'état de vos interfaces (dans R1, par exemple) les commandes suivantes sont utiles :

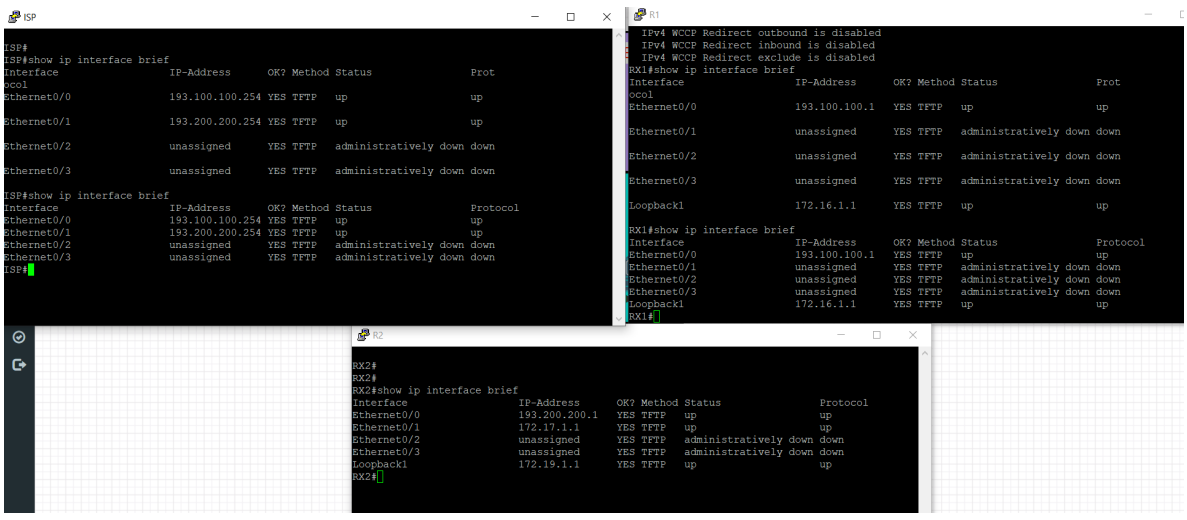
```
R1# show ip interface brief
R1# show interface <interface-name>
R1# show ip interface <interface-name>
```

Un « status » différent de `up` indique très souvent que l'interface n'est pas active.

Un « protocol » différent de `up` indique la plupart du temps que l'interface n'est pas connectée correctement (en tout cas pour Ethernet).

Question 1: Avez-vous rencontré des problèmes ? Si oui, qu'avez-vous fait pour les résoudre ?

Réponse :



Les seules interfaces "down" sont celles qu'on n'utilise pas durant ce labo. On voit bien que toutes les interfaces employés dans la topologie réseau sont actives.

On constate donc que tout va bien et qu'aucun problème n'est à signaler

---

- Contrôlez que votre serveur DHCP sur R2 est fonctionnel - Contrôlez que le serveur DHCP préconfiguré pour vous sur R2 a bien distribué une adresse IP à votre station « VPC ».

Les commandes utiles sont les suivantes :

```
R2# show ip dhcp pool
R2# show ip dhcp binding
```

Côté station (VPC) vous pouvez valider les paramètres reçus avec la commande `show ip`. Si votre station n'a pas reçu d'adresse IP, utilisez la commande `ip dhcp`.

- Contrôlez la connectivité sur toutes les interfaces à l'aide de la commande ping.

Pour contrôler la connectivité les commandes suivantes sont utiles :

```
R2# ping ip-address
R2# show arp (utile si un firewall est actif)
```

Pour votre topologie il est utile de contrôler la connectivité entre :

- R1 vers ISP1 (193.100.100.254)
- R2 vers ISP2 (193.200.200.254)
- R2 (193.200.200.1) vers RX1 (193.100.100.1) via Internet
- R2 (172.17.1.1) et votre poste « VPC » (172.17.1.100)

Question 2: Tous vos pings ont-ils passé ? Si non, est-ce normal ? Dans ce cas, trouvez la source du problème et corrigez-la.

Réponse :

Les pings ne passaient pas avant d'avoir tapé la commande "ip dhcp" sur le VPC car ce dernier n'avait pas encore reçu d'adresse ip du DHCP.

De ISP1 à R1 et de ISP2 à R2

```
ISP#ping 193.100.100.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 193.100.100.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
ISP#ping 193.200.200.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 193.200.200.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

De R1 à ISP1 et de R1 à R2

```
RX1#ping 193.100.100.254
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 193.100.100.254, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
RX1#ping 193.200.200.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 193.200.200.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

De R2 à à ISP2, de R2 à R1 et de R2 à VPC

```
RX2#ping 172.17.1.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.1.100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
RX2#ping 193.200.200.254
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 193.200.200.254, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
RX2#ping 193.100.100.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 193.100.100.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

De VPC à R2

```
VPCS> ping 172.17.1.1

84 bytes from 172.17.1.1 icmp_seq=1 ttl=255 time=0.578 ms
84 bytes from 172.17.1.1 icmp_seq=2 ttl=255 time=0.257 ms
84 bytes from 172.17.1.1 icmp_seq=3 ttl=255 time=0.194 ms
84 bytes from 172.17.1.1 icmp_seq=4 ttl=255 time=0.193 ms
84 bytes from 172.17.1.1 icmp_seq=5 ttl=255 time=0.267 ms
```

- Activation de « debug » et analyse des messages ping.

Maintenant que vous êtes familier avec les commandes « show » nous allons travailler avec les commandes de « debug ». A titre de référence, vous allez capturer les messages envoyés lors d'un ping entre votre « poste utilisateur » et un routeur. Trouvez ci-dessous la commande de « debug » à activer.

Activer les messages relatif aux paquets ICMP émis par les routeurs (repérer dans ces messages les type de paquets ICMP émis - < ICMP: echo xxx sent ...>)

```
R2# debug ip icmp
```

Pour déclencher et pratiquer les captures vous allez « pinger » votre routeur R1 avec son IP=193.100.100.1 depuis votre « VPC ». Durant cette opération vous tenterez d'obtenir en simultané les informations suivantes :

- Une trace sniffer (Wireshark) à la sortie du routeur R2 vers Internet. Si vous ne savez pas utiliser Wireshark avec eve-ng, référez-vous au document explicatif eve-ng. Le filtre de

capture (attention, c'est un filtre de **capture** et pas un filtre d'affichage) suivant peut vous aider avec votre capture : `ip host 193.100.100.1`.

- Les messages de R1 avec `debug ip icmp`.

Question 3: Montrez vous captures

Screenshots :

Ping debug R1

```
RX1#
*May 10 14:47:41.612: ICMP: echo reply sent, src 193.100.100.1, dst 172.17.1.100, topology BASE
, dscp 0 topoid 0
*May 10 14:47:42.614: ICMP: echo reply sent, src 193.100.100.1, dst 172.17.1.100, topology BASE
, dscp 0 topoid 0
RX1#
*May 10 14:47:43.617: ICMP: echo reply sent, src 193.100.100.1, dst 172.17.1.100, topology BASE
, dscp 0 topoid 0
*May 10 14:47:44.621: ICMP: echo reply sent, src 193.100.100.1, dst 172.17.1.100, topology BASE
, dscp 0 topoid 0
RX1#
*May 10 14:47:45.623: ICMP: echo reply sent, src 193.100.100.1, dst 172.17.1.100, topology BASE
, dscp 0 topoid 0
```

Wireshark capture de R2 e0/0

1	0.000000	aa:bb:cc:00:20:00	aa:bb:cc:00:20:00	LOOP	60	Reply
2	0.538403	aa:bb:cc:00:30:10	aa:bb:cc:00:30:10	LOOP	60	Reply
3	0.614921	193.200.200.1	224.0.0.9	RIPv2	66	Response
4	10.007578	aa:bb:cc:00:20:00	aa:bb:cc:00:20:00	LOOP	60	Reply
5	10.548415	aa:bb:cc:00:30:10	aa:bb:cc:00:30:10	LOOP	60	Reply
6	11.768858	172.17.1.100	193.100.100.1	ICMP	98	Echo (ping) request id=0x0d14, seq=1/256, ttl=63 (reply in 7)
7	11.769593	193.100.100.1	172.17.1.100	ICMP	98	Echo (ping) reply id=0x0d14, seq=1/256, ttl=254 (request in 6)
8	12.770876	172.17.1.100	193.100.100.1	ICMP	98	Echo (ping) request id=0x0e14, seq=2/512, ttl=63 (reply in 9)
9	12.771600	193.100.100.1	172.17.1.100	ICMP	98	Echo (ping) reply id=0x0e14, seq=2/512, ttl=254 (request in 8)
10	13.774122	172.17.1.100	193.100.100.1	ICMP	98	Echo (ping) request id=0x0f14, seq=3/768, ttl=63 (reply in 11)
11	13.774611	193.100.100.1	172.17.1.100	ICMP	98	Echo (ping) reply id=0x0f14, seq=3/768, ttl=254 (request in 10)
12	14.777821	172.17.1.100	193.100.100.1	ICMP	98	Echo (ping) request id=0x1014, seq=4/1024, ttl=63 (reply in 13)
13	14.778250	193.100.100.1	172.17.1.100	ICMP	98	Echo (ping) reply id=0x1014, seq=4/1024, ttl=254 (request in 12)
14	15.780324	172.17.1.100	193.100.100.1	ICMP	98	Echo (ping) request id=0x1114, seq=5/1280, ttl=63 (reply in 15)
15	15.780836	193.100.100.1	172.17.1.100	ICMP	98	Echo (ping) reply id=0x1114, seq=5/1280, ttl=254 (request in 14)
16	16.829477	193.200.200.254	224.0.0.9	RIPv2	86	Response
17	20.013921	aa:bb:cc:00:20:00	aa:bb:cc:00:20:00	LOOP	60	Reply
18	20.553484	aa:bb:cc:00:30:10	aa:bb:cc:00:30:10	LOOP	60	Reply
19	25.323446	aa:bb:cc:00:30:10	CDP/VTP/DTP/PAgP/UD…	CDP	379	Device ID: ISP.lab.local Port ID: Ethernet0/1
20	26.194732	193.200.200.1	224.0.0.9	RIPv2	66	Response
21	26.705066	aa:bb:cc:00:20:00	CDP/VTP/DTP/PAgP/UD…	CDP	379	Device ID: RX2.lab.local Port ID: Ethernet0/0
22	27.508340	aa:bb:cc:00:20:00	DEC-MOP-Remote-Cons…	0x6002	77	DEC DNA Remote Console
23	30.022408	aa:bb:cc:00:20:00	aa:bb:cc:00:20:00	LOOP	60	Reply
24	30.553346	aa:bb:cc:00:30:10	aa:bb:cc:00:30:10	LOOP	60	Reply
25	32.989658	aa:bb:cc:00:30:10	DEC-MOP-Remote-Cons…	0x6002	77	DEC DNA Remote Console

---

Configuration VPN LAN 2 LAN

Il est votre responsabilité de chercher vous-même sur internet toute information relative à la configuration que vous ne comprenez pas ! La documentation CISCO en ligne est extrêmement complète et le temps pour rendre le labo est plus que suffisant !

Nous allons établir un VPN IKE/IPsec entre le réseau de votre « loopback 1 » sur R1 (172.16.1.0/24) et le réseau de votre « VPC » R2 (172.17.1.0/24). La terminologie Cisco est assez « particulière » ; elle est listée ici, avec les étapes de configuration, qui seront les suivantes :

- Configuration des « proposals » IKE sur les deux routeurs (policy)
- Configuration des clefs « preshared » pour l'authentification IKE (key)
- Activation des « keepalive » IKE
- Configuration du mode de chiffrement IPsec
- Configuration du trafic à chiffrer (access list)
- Activation du chiffrement (crypto map)

Configuration IKE

Sur le routeur R1 nous activons un « proposal » IKE. Il s'agit de la configuration utilisée pour la phase 1 du protocole IKE. Le « proposal » utilise les éléments suivants :

Element	Value
Encryption	AES 256 bits
Signature	Basée sur SHA-1
Authentification	Preshared Key
Diffie-Hellman	avec des nombres premiers sur 1536 bits
Renouvellement	des SA de la Phase I toutes les 30 minutes
Keepalive	toutes les 30 secondes avec 3 « retry »
Preshared-Key	pour l'IP du distant avec le texte « cisco-1 », Notez que dans la réalité nous utiliserions un texte plus compliqué.

Les commandes de configurations sur R1 ressembleront à ce qui suit :

```
crypto isakmp policy 20
  encr aes 256
  authentication pre-share
  hash sha
  group 5
  lifetime 1800
crypto isakmp key cisco-1 address 193.200.200.1 no-xauth
crypto isakmp keepalive 30 3
```

Sur le routeur R2 nous activons un « proposal » IKE supplémentaire comme suit :

```
crypto isakmp policy 10
  encr 3des
  authentication pre-share
  hash md5
  group 2
  lifetime 1800
crypto isakmp policy 20
  encr aes 256
  authentication pre-share
  hash sha
  group 5
  lifetime 1800
crypto isakmp key cisco-1 address 193.100.100.1 no-xauth
crypto isakmp keepalive 30 3
```

Vous pouvez consulter l'état de votre configuration IKE avec les commandes suivantes. Faites part de vos remarques :

Question 4: Utilisez la commande `show crypto isakmp policy` et faites part de vos remarques :

Réponse :

R1 policy :

```
Global IKE policy
Protection suite of priority 20
        encryption algorithm:   AES - Advanced Encryption Standard (256 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #5 (1536 bit)
        lifetime:               1800 seconds, no volume limit
```

R2 policy :

```
Global IKE policy
Protection suite of priority 10
        encryption algorithm:   Three key triple DES
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #2 (1024 bit)
        lifetime:               1800 seconds, no volume limit
Protection suite of priority 20
        encryption algorithm:   AES - Advanced Encryption Standard (256 bit keys).
        hash algorithm:         Secure Hash Standard
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #5 (1536 bit)
        lifetime:               1800 seconds, no volume limit
```

Question 5: Utilisez la commande `show crypto isakmp key` et faites part de vos remarques :

Réponse :

R1 key :

```
Keyring      Hostname/Address                                Preshared Key

default      193.200.200.1                                   cisco-1
```

R2 key :

```
Keyring      Hostname/Address                                Preshared Key

default      193.100.100.1                                   cisco-1
```

La clé "cisco-1" n'est pas très sécurisé.

Configuration IPsec

Nous allons maintenant configurer IPsec de manière identique sur les deux routeurs. Pour IPsec nous allons utiliser les paramètres suivants :

Paramètre	Valeur
IPsec avec IKE	IPsec utilisera IKE pour générer ses SA
Encryption	AES 192 bits
Signature	Basée sur SHA-1
Proxy ID R1	172.16.1.0/24
Proxy ID R2	172.17.1.0/24

Changement de SA toutes les 5 minutes ou tous les 2.6MB

Si inactifs les SA devront être effacés après 15 minutes

Les commandes de configurations sur R1 ressembleront à ce qui suit :

```
crypto ipsec security-association lifetime kilobytes 2560
crypto ipsec security-association lifetime seconds 300
crypto ipsec transform-set STRONG esp-aes 192 esp-sha-hmac
  ip access-list extended TO-CRYPT
  permit ip 172.16.1.0 0.0.0.255 172.17.1.0 0.0.0.255
crypto map MY-CRYPTO 10 ipsec-isakmp
  set peer 193.200.200.1
  set security-association idle-time 900
  set transform-set STRONG
  match address TO-CRYPT
```

Les commandes de configurations sur R2 ressembleront à ce qui suit :

```
crypto ipsec security-association lifetime kilobytes 2560
crypto ipsec security-association lifetime seconds 300
crypto ipsec transform-set STRONG esp-aes 192 esp-sha-hmac
  mode tunnel
  ip access-list extended TO-CRYPT
  permit ip 172.17.1.0 0.0.0.255 172.16.1.0 0.0.0.255
crypto map MY-CRYPTO 10 ipsec-isakmp
  set peer 193.100.100.1
  set security-association idle-time 900
  set transform-set STRONG
  match address TO-CRYPT
```

Vous pouvez contrôler votre configuration IPsec avec les commandes suivantes :

```
show crypto ipsec security-association
show crypto ipsec transform-set
show access-list TO-CRYPT
show crypto map
```

Activation IPsec & test

Pour activer cette configuration IKE & IPsec il faut appliquer le « crypto map » sur l'interface de sortie du trafic où vous voulez que l'encryption prenne place.

Sur R1 il s'agit, selon le schéma, de l'interface « Ethernet0/0 » et la configuration sera :

```
interface Ethernet0/0
  crypto map MY-CRYPTO
```

Sur R2 il s'agit, selon le schéma, de l'interface « Ethernet0/0 » et la configuration sera :

```
interface Ethernet0/0
  crypto map MY-CRYPTO
```

Après avoir entré cette commande, normalement le routeur vous indique que IKE (ISAKMP) est activé. Vous pouvez contrôler que votre « crypto map » est bien appliquée sur une interface avec la commande `show crypto map`.

Pour tester si votre VPN est correctement configuré vous pouvez maintenant lancer un « ping » sur la « loopback 1 » de votre routeur RX1 (172.16.1.1) depuis votre poste utilisateur (172.17.1.100). De manière à recevoir toutes les notifications possibles pour des paquets ICMP envoyés à un routeur comme RX1 vous pouvez activer un « debug » pour cela. La commande serait :

```
debug ip icmp
```

Pensez à démarrer votre sniffer sur la sortie du routeur R2 vers internet avant de démarrer votre ping, collectez aussi les éventuels messages à la console des différents routeurs.

Question 6: Ensuite faites part de vos remarques dans votre rapport. :

Réponse :

R1 isakmp UP

```
RX1(config)#interface Ethernet0/0
RX1(config-if)#crypto map MY-CRYPTO
RX1(config-if)#
*May 10 16:15:33.526: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
```

R1 crypto map

```
RX1(config)#do show crypto map
Crypto Map IPv4 "MY-CRYPTO" 10 ipsec-isakmp
        Peer = 193.200.200.1
        Extended IP access list TO-CRYPT
            access-list TO-CRYPT permit ip 172.16.1.0 0.0.0.255 172.17.1.0 0.0.0.255
        Current peer: 193.200.200.1
        Security association lifetime: 2560 kilobytes/300 seconds
        Security association idletime: 900 seconds
        Responder-Only (Y/N): N
        PFS (Y/N): N
        Mixed-mode : Disabled
        Transform sets={
                STRONG:  { esp-192-aes esp-sha-hmac  } ,
        }
        Interfaces using crypto map MY-CRYPTO:
                Ethernet0/0

        Interfaces using crypto map NiStTeSt1:
```

R2 isakmp UP

```
RX2(config)#interface Ethernet0/0
RX2(config-if)#crypto map MY-CRYPTO
RX2(config-if)#
*May 10 16:16:00.387: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
```

R2 crypto map

```
RX2(config)#do show crypto map
Crypto Map IPv4 "MY-CRYPTO" 10 ipsec-isakmp
        Peer = 193.100.100.1
        Extended IP access list TO-CRYPT
            access-list TO-CRYPT permit ip 172.17.1.0 0.0.0.255 172.16.1.0 0.0.0.255
        Current peer: 193.100.100.1
        Security association lifetime: 2560 kilobytes/300 seconds
        Security association idletime: 900 seconds
        Responder-Only (Y/N): N
        PFS (Y/N): N
        Mixed-mode : Disabled
        Transform sets={
                STRONG:  { esp-192-aes esp-sha-hmac  } ,
        }
        Interfaces using crypto map MY-CRYPTO:
                Ethernet0/0

        Interfaces using crypto map NiStTeSt1:
```

R1 icmp debug

```
*May 10 16:34:33.635: ICMP: echo reply sent, src 172.16.1.1, dst 172.17.1.100, topology BASE, d
scp 0 topoid 0
*May 10 16:34:34.638: ICMP: echo reply sent, src 172.16.1.1, dst 172.17.1.100, topology BASE, d
scp 0 topoid 0
RX1(config)#
*May 10 16:34:35.641: ICMP: echo reply sent, src 172.16.1.1, dst 172.17.1.100, topology BASE, d
scp 0 topoid 0
*May 10 16:34:36.644: ICMP: echo reply sent, src 172.16.1.1, dst 172.17.1.100, topology BASE, d
scp 0 topoid 0
RX1(config)#
*May 10 16:34:37.646: ICMP: echo reply sent, src 172.16.1.1, dst 172.17.1.100, topology BASE, d
scp 0 topoid 0
```

R2 wireshark

1	0.000000	aa:bb:cc:00:20:00	aa:bb:cc:00:20:00	LOOP	60	Reply
2	0.561636	aa:bb:cc:00:30:10	aa:bb:cc:00:30:10	LOOP	60	Reply
3	2.721347	193.200.200.1	224.0.0.9	RIPv2	66	Response
4	10.008040	aa:bb:cc:00:20:00	aa:bb:cc:00:20:00	LOOP	60	Reply
5	10.409808	193.200.200.1	193.100.100.1	ESP	166	ESP (SPI=0x2c95474f)
6	10.410813	193.100.100.1	193.200.200.1	ESP	166	ESP (SPI=0x1976ac8d)
7	10.568492	aa:bb:cc:00:30:10	aa:bb:cc:00:30:10	LOOP	60	Reply
8	11.413129	193.200.200.1	193.100.100.1	ESP	166	ESP (SPI=0x2c95474f)
9	11.414248	193.100.100.1	193.200.200.1	ESP	166	ESP (SPI=0x1976ac8d)
10	11.889551	aa:bb:cc:00:30:10	CDP/VTP/DTP/PAgP/UD…	CDP	379	Device ID: ISP.lab.local Port ID: Ethernet0/1
11	12.416012	193.200.200.1	193.100.100.1	ESP	166	ESP (SPI=0x2c95474f)
12	12.417013	193.100.100.1	193.200.200.1	ESP	166	ESP (SPI=0x1976ac8d)
13	13.419113	193.200.200.1	193.100.100.1	ESP	166	ESP (SPI=0x2c95474f)
14	13.419722	193.100.100.1	193.200.200.1	ESP	166	ESP (SPI=0x1976ac8d)
15	14.421226	193.200.200.1	193.100.100.1	ESP	166	ESP (SPI=0x2c95474f)
16	14.421810	193.100.100.1	193.200.200.1	ESP	166	ESP (SPI=0x1976ac8d)
17	20.009274	aa:bb:cc:00:20:00	aa:bb:cc:00:20:00	LOOP	60	Reply
18	20.428462	193.200.200.254	224.0.0.9	RIPv2	86	Response
19	20.575691	aa:bb:cc:00:30:10	aa:bb:cc:00:30:10	LOOP	60	Reply

---

Question 7: Reportez dans votre rapport une petite explication concernant les différents « timers » utilisés par IKE et IPsec dans cet exercice (recherche Web). :

Réponse :

IKE : Lifetime, cela représente le temps de vie des SA et KeepAlive qui représente quand les paquets doivent être envoyés.

IPSec : Lifetime, cela représente le temps de vie des SA et Idle-time qui représente le temps d'inactivité.

Synthèse d'IPsec

En vous appuyant sur les notions vues en cours et vos observations en laboratoire, essayez de répondre aux questions. À chaque fois, expliquez comment vous avez fait pour déterminer la réponse exacte (capture, config, théorie, ou autre).

Question 8: Déterminez quel(s) type(s) de protocole VPN a (ont) été mis en œuvre (IKE, ESP, AH, ou autre).

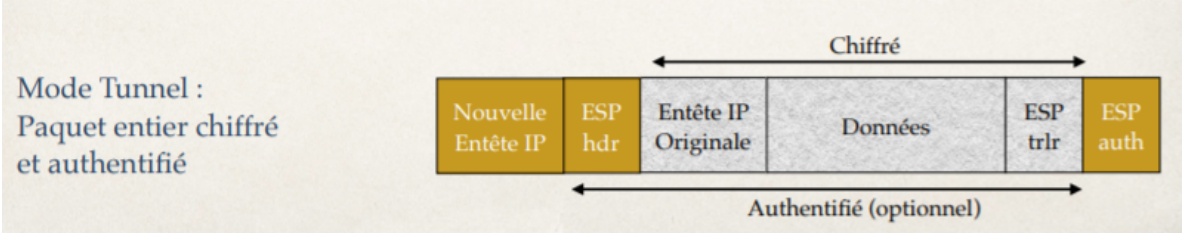
Réponse : IKE et ESP

Question 9: Expliquez si c'est un mode tunnel ou transport.

Réponse : Mode tunnel, car dans les commandes de configuration de R2 on tape "mode tunnel". On ne peut également que mettre un VPN en mode tunnel si on veut relier 2 réseaux privés.

Question 10: Expliquez quelles sont les parties du paquet qui sont chiffrées. Donnez l'algorithme cryptographique correspondant.

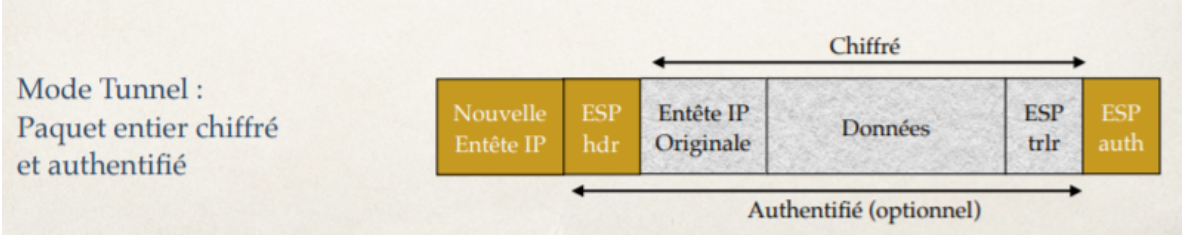
Réponse : On utilise l'algorithme cryptographique "AES", on le décrit souvent sur plusieurs de nos commandes.



Question 11: Expliquez quelles sont les parties du paquet qui sont authentifiées. Donnez l'algorithme cryptographique correspondant.

Réponse :

On utilise l'algorithme cryptographique HMAC accompagné de SHA1.



Question 12: Expliquez quelles sont les parties du paquet qui sont protégées en intégrité. Donnez l'algorithme cryptographique correspondant.

Réponse : Un paquet se doit d'être intègre pour être authentifié, dès lors, toutes les parties authentifiées seront protégées en intégrité.

On utilise l'algorithme cryptographique HMAC accompagné de SHA1 également.
