Exercice SISR2-TP-ACL-1

Config ACL	Adresse IP source	Autorisé ?	Drop ?
Access-list 10 permit 192.168.122.128 0.0.0.63	192.168.122.198	Non	Oui
Access-list 11 permit 192.168.2.64 0.0.0.31	192.168.2.25	Non	Oui
Access-list 12 permit 192.168.225.32 0.0.0.31	192.168.225.32	Oui	Non
Access-list 13 permit 192.168.100.0 0.0.0.255	192.168.100.244	Oui	Non
Access-list 14 deny 192.168.10.100 0.0.0.0	192.168.10.100	Non	Oui
Access-list 15 deny any	192.168.1.1	Non	Oui
Access-list 16 remark deny 192.168.10.0 255.255.255.0	192.168.10.50	N/A	N/A
Ip access-list standard ACCES-BLOQUE deny 192.168.10.10 0.0.0.0	192.168.10.10	Non	Oui
Ip access-list standard ACCES-INTERDIT Deny 192.168.10.10 0.0.0.255	192.168.10.10	Non	Oui
Ip access-list 17 permit 192.168.10.100 0.0.0.255	192.168.10.101	Oui	Non

Explications :

- **Access-list 10** : La plage autorisée est 192.168.122.128 à 192.168.122.191. L'adresse 192.168.122.198 est en dehors de cette plage, donc elle est bloquée.
  128+63 = 191
- **Access-list 11** : La plage autorisée est 192.168.2.64 à 192.168.2.95. L'adresse 192.168.2.25 est en dehors de cette plage, donc elle est bloquée.
- **Access-list 12** : L'adresse IP 192.168.225.32 est exactement dans la plage autorisée, donc elle est permise.
- **Access-list 13** : Toute adresse dans la plage 192.168.100.0 à 192.168.100.255 est permise, donc 192.168.100.244 est autorisée.
- **Access-list 14** : L'adresse 192.168.10.100 correspond exactement à la règle "deny", donc elle est bloquée.
- **Access-list 15** : "deny any" bloque toute adresse, donc 192.168.1.1 est bloquée.
- **Access-list 16** : "remark" n'est qu'un commentaire, donc il n'a aucun effet sur l'autorisation ou le blocage du trafic.
- **ACCES-BLOQUE** : L'adresse 192.168.10.10 est explicitement bloquée.
- **ACCES-INTERDIT** : Toute adresse dans la plage 192.168.10.0 à 192.168.10.255 est bloquée, donc 192.168.10.10 est bloquée.

- **Access-list 17** : La plage autorisée est 192.168.10.100 à 192.168.10.255. L'adresse 192.168.10.101 se trouve dans cette plage, donc elle est autorisée.

Exercice SISR2-TP-ACL-2 – Isoler un sous réseau

Récapitulatif de l'exercice

Objectif : L'objectif de cet exercice est de configurer une **ACL** sur un routeur afin d'isoler le réseau **192.168.1.0/24** et de bloquer tout trafic provenant de ce sous-réseau sur une interface du routeur. Les deux PC doivent être capables de communiquer entre eux via un **ping**, mais le trafic provenant de **192.168.1.0/24** doit être filtré.

Étapes de réalisation :

Création du réseau :
- Les deux PCs ont été connectés à des interfaces distinctes sur le routeur via des commutateurs.
- Les adresses IP ont été configurées :
  - **PC0** (192.168.1.10/24) sur le réseau 192.168.1.0/24.
  - **PC1** (192.168.2.10/24) sur le réseau 192.168.2.0/24.
  - Les passerelles sont respectivement **192.168.1.1** (pour PC0) et **192.168.2.1** (pour PC1).

Configuration des interfaces du routeur :
- Les interfaces du routeur ont été configurées avec les adresses IP suivantes :
  - **gig0/0** : 192.168.1.1/24
  - **gig0/1** : 192.168.2.1/24

Création de l'ACL 10 :
- Une **ACL standard** a été créée pour bloquer tout le trafic venant du sous-réseau **192.168.1.0/24**.

```
Router(config)# access-list 10 deny 192.168.1.0 0.0.0.255
Router(config)# access-list 10 permit any
```

Application de l'ACL en entrée sur l'interface fa0/0 :
- L'ACL a été appliquée en **entrée** sur l'interface **gig0/0** afin de filtrer le trafic arrivant du réseau **192.168.1.0/24**:

```
Router(config)# interface gig0/0
Router(config-if)# ip access-group 10 in
```

Résultat :
- **PC0** (dans le réseau 192.168.1.0/24) ne peut plus pinger **PC1** (192.168.2.0/24), car le trafic entrant sur l'interface gig0/0 depuis **PC1** est bloqué par l'ACL.

- Le trafic de **PC1** vers **PC0** est autorisé, car l'ACL bloque uniquement le trafic en provenance du réseau 192.168.1.0/24. Cependant PC1 ne peut plus recevoir de réponse.

Conclusion : L'exercice a permis d'appliquer une ACL standard pour isoler le sous-réseau 192.168.1.0/24 et bloquer tout trafic entrant de ce réseau sur l'interface du routeur. Le test de ping a confirmé que l'ACL fonctionne correctement en bloquant le trafic en fonction des règles spécifiées.

Exercice SISR2-TP-ACL-3 – Refuser un hôte

Récapitulatif

Objectif : L'objectif de cet exercice est de configurer une **ACL étendue** sur un routeur afin de bloquer le trafic entre le sous-réseau **192.168.2.0/24** et l'hôte **192.168.1.11**, tout en permettant à ce sous-réseau de communiquer avec les autres hôtes.

Étapes de réalisation :

Configuration des réseaux :

- **PC0** dans le réseau **192.168.1.0/24** avec l'adresse **192.168.1.10**.
- **PC1** dans le réseau **192.168.1.0/24** avec l'adresse **192.168.1.11**.
- **PC2** dans le réseau **192.168.2.0/24** avec l'adresse **192.168.2.10**.
- Les passerelles sont respectivement **192.168.1.1** (pour PC0 et PC1) et **192.168.2.1** (pour PC2).

Création de l'ACL étendue : Une **ACL étendue** a été créée pour refuser le trafic entre le réseau **192.168.2.0/24** et l'hôte **192.168.1.11**, tout en permettant tout autre type de communication :

```
Router(config)# access-list 100 deny ip 192.168.2.0 0.0.0.255 host 192.168.1.11
Router(config)# access-list 100 permit ip any any
```

Application de l'ACL aux interfaces du routeur :

- L'ACL a été appliquée en **entrée** sur les deux interfaces du routeur pour s'assurer que tout le trafic en provenance de **192.168.2.0/24** vers **192.168.1.11** est bloqué, tandis que le reste du trafic est autorisé.

```
Router(config)# interface gig0/0
Router(config-if)# ip access-group 100 in
Router(config)# interface gig0/1
Router(config-if)# ip access-group 100 in
```

Résultats :

- Les tests effectués avec des **pings** montrent que le **PC1** et le **PC2** ne peuvent plus communiquer, conformément aux règles ACL.
- **PC0** et **PC2** peuvent toujours communiquer ensemble, ainsi que **PC0** et **PC1.**

Conclusion :

L'exercice a permis de configurer avec succès une **ACL étendue** pour bloquer sélectivement le trafic concernant un hôte spécifique du réseau. L'ACL étendue a été appliquée sur les deux interfaces du routeur, et le filtrage du trafic fonctionne comme prévu.

Exercice SISR2-TP-ACL-4 – Sécuriser l'accès au terminal virtuel du routeur par ACL

Récapitulatif

Objectif : L'objectif de cet exercice est de sécuriser l'accès au terminal virtuel du routeur via Telnet en limitant l'accès à une seule machine dans le sous-réseau **192.168.1.0/24**. Seule la machine ayant l'adresse IP **192.168.1.254** doit pouvoir se connecter au routeur via Telnet.

Étapes de réalisation :

Création du réseau :

- Le routeur est connecté à un commutateur sans adresse IP, et 4 hôtes sont configurés dans le sous-réseau **192.168.1.0/24**. La passerelle du sous-réseau est **192.168.1.1**.

Configuration de l'ACL : Une **ACL standard** est utilisée pour restreindre l'accès Telnet à la seule machine autorisée **192.168.1.254**. Voici les commandes pour créer et appliquer cette ACL :

```
Router(config)# access-list 10 permit 192.168.1.254
Router(config)# access-list 10 deny any
```

.

Configuration de la ligne d'entrée du terminal virtuel (VTY) comme indiqué dans l'exercice.

Test de la configuration :

- La connexion **réussit** avec le PC3 dont l'ip est 192.168.1.254
- Avec les trois autres la connexion échoue avec comme message **"connexion refused by remote host"**

Résultats :

- Seule la machine **192.168.1.254** peut se connecter au routeur via Telnet, conformément aux règles définies dans l'ACL.
- Toutes les autres machines du sous-réseau **192.168.1.0/24** se voient refuser l'accès Telnet au routeur.

Conclusion :

Cet exercice a permis de mettre en place une **ACL standard** pour sécuriser l'accès au terminal virtuel du routeur. En appliquant l'ACL sur les lignes **vty**, l'accès via Telnet a été limité à une seule machine spécifique. Le test a confirmé que seule la machine autorisée peut se connecter avec succès au routeur

Exercice SISR2-TP-ACL-5 - Interdire un port TCP

Récapitulatif

Objectif : L'objectif est de configurer une **ACL étendue** pour interdire tout trafic **FTP** (port TCP 21) provenant du sous-réseau **192.168.2.0/24** à destination d'un serveur FTP sur le réseau **192.168.3.0/24**. Cependant, les autres types de trafic, comme **HTTP**, doivent rester autorisés.

Étapes de réalisation :

Création du réseau :

- Deux PC dans deux sous-réseaux

- Un serveur FTP et WEB dans le sous-réseau **192.168.3.0/24** avec l'adresse IP **192.168.3.10**.

Test connexion FTP : OK sur les deux PC

Création de l'ACL étendue pour bloquer FTP :

- Nous allons créer une **ACL étendue** qui bloque le trafic **FTP** (port **TCP 21**) provenant du sous-réseau **192.168.2.0/24** vers le réseau **192.168.3.0/24**, tout en autorisant les autres types de trafic, notamment **HTTP**.

Commandes à utilisées pour créer l'ACL :

Router(config)# access-list 100 deny tcp 192.168.2.0 0.0.0.255 192.168.3.0 0.0.0.255 eq 21
Router(config)# access-list 100 permit ip any any

- J'applique l'ACL en **entrée** sur l'interface du routeur connectée au réseau **192.168.2.0/24** pour empêcher ce sous-réseau d'accéder au FTP du serveur.

Router(config)# interface GigabitEthernet 0/1
Router(config-if)# ip access-group 100 in

Vérification de la configuration :

- **Test FTP : le pc n'arrive plus à joindre le serveur avec le protocol FTP : OK.**
- **Connexion avec le navigateur web : OK.**
- **Connexion FTP avec le PC0 : ip 192.168.1.10 : OK.**

Résultats :

- Le **trafic FTP** est bloqué pour tout hôte du réseau **192.168.2.0/24** vers le serveur **192.168.3.10**.
- Le **trafic HTTP** reste autorisé et fonctionne correctement.

Conclusion :

Cet exercice a démontré l'utilisation d'une **ACL étendue** pour interdire un trafic spécifique basé sur le **port TCP** (FTP) tout en permettant les autres communications comme **HTTP**. L'ACL a été appliquée sur la bonne interface du routeur, et les tests ont confirmé que le blocage du port FTP est fonctionnel.

Exercice SISR2-TP-ACL-6 – Trafic HTTP

Récapitulatif de l'exercice SISR2-TP-ACL-6 – Trafic HTTP

Objectif : L'objectif de cet exercice est de configurer une **ACL étendue** pour permettre uniquement le **trafic HTTP (port 80)** depuis le sous-réseau **192.168.1.0/24** vers les autres réseaux, tout en bloquant tous les autres types de trafic provenant de ce sous-réseau.

Étapes de réalisation à partir de la même configuration que pour l'exercice 5:

Voici les commandes à utiliser pour configurer cette ACL :

access-list 102 deny tcp 192.168.1.0 0.0.0.255 any ne 80

Appliquer l'ACL sur l'interface du routeur :

- J'applique l'ACL en **entrée** sur l'interface du routeur connectée au réseau **192.168.2.0/24** pour empêcher ce sous-réseau d'accéder au FTP du serveur

Router(config)# interface GigabitEthernet 0/0
Router(config-if)# ip access-group 102 in

(Optionnel) : je l'ai aussi appliqué en sortie pour une meilleure cohérence

Vérification de la configuration :

- **Les PING se bloquent au niveau du routeur, protocol ICMP différent de HTTP : OK**
- **Connexion avec le navigateur web : OK.**

Résultats :

- Seul le **trafic HTTP** (port 80) est autorisé pour le sous-réseau **192.168.1.0/24** vers d'autres réseaux. (+ en provenance d'autres réseaux)

Conclusion :

L'ACL étendue a été configurée avec succès pour permettre uniquement le **trafic HTTP** provenant du sous-réseau **192.168.1.0/24** et bloquer tout autre type de trafic. Les tests ont confirmé que la connexion **HTTP** est possible, tandis que le **trafic FTP** ou ICMP sont bloqués