Annexes : ACL

Les **Access Control Lists (ACLs) étendues** sur les équipements Cisco permettent de filtrer le trafic réseau en fonction de plusieurs critères tels que l'adresse source, l'adresse destination, le protocole, les ports, etc.

Contrairement aux ACLs standard (qui ne filtrent que sur l'adresse IP source), les ACLs étendues offrent un contrôle beaucoup plus granulaire et sont couramment utilisées pour gérer la sécurité réseau.

Syntaxe d'une ACL étendue

La syntaxe générale d'une ACL étendue est la suivante :

access-list [numéro_ACL] [action] [protocole] [adresse_source] [wildcard_source] [opérateur_port_source (optionnel)] [adresse_destination] [wildcard_destination] [opérateur_port_destination (optionnel)]

Explication des éléments :

1. **access-list [numéro_ACL]** :
    - Il s'agit du **numéro de l'ACL**. Pour une ACL étendue, ce numéro doit être compris entre **100 et 199** ou entre **2000 et 2699**.
2. **[action]** :
    - L'action à effectuer sur les paquets correspondants : **permit** ou **deny**.
3. **[protocole]** :
    - Spécifie le **protocole** à filtrer. Il peut s'agir de **ip** (pour tous les protocoles IP), **tcp**, **udp**, **icmp**, etc.
4. **[adresse_source] [wildcard_source]** :
    - C'est l'**adresse IP source** à partir de laquelle le paquet est envoyé.
    - Le **wildcard mask** définit la plage d'adresses. Un **wildcard mask** est l'inverse d'un masque de sous-réseau (par exemple, le masque 0.0.0.255 correspond à un sous-réseau de 255.255.255.0).
    - Exemple : 192.168.1.0 0.0.0.255 couvre toutes les adresses dans le sous-réseau **192.168.1.0/24**.

5. **[opérateur_port_source] (optionnel)** :
    - Utilisé lorsque le protocole est **TCP** ou **UDP**. Permet de filtrer les paquets en fonction du **port source**.
    - Les opérateurs possibles incluent : **eq** (égal à), **lt** (moins que), **gt** (plus que), **neq** (différent de), et **range** (pour définir une plage de ports).
6. **[adresse_destination] [wildcard_destination]** :
    - C'est l'**adresse IP destination** à laquelle le paquet est envoyé.
    - Exemple : 10.0.0.0 0.0.0.255 couvre toutes les adresses dans le sous-réseau **10.0.0.0/24**.
7. **[opérateur_port_destination] (optionnel)** :
    - Utilisé pour définir le **port de destination** si le protocole est **TCP** ou **UDP**.
    - Exemple : **eq** 443 pour spécifier que le port de destination est **443** (HTTPS).

Exemple de syntaxe et explication

Exemple 1 : Autoriser tout le trafic IP entre deux sous-réseaux

access-list 100 permit ip 192.168.1.0 0.0.0.255 10.0.0.0 0.0.0.255

- **Numéro de l'ACL** : 100
- **Action** : permit, donc ce trafic est autorisé.
- **Protocole** : ip (tous les paquets IP, quelle que soit la couche transport).
- **Adresse source** : 192.168.1.0 (sous-réseau) avec le **wildcard** 0.0.0.255 (tous les hôtes du sous-réseau 192.168.1.0/24).
- **Adresse destination** : 10.0.0.0 avec le **wildcard** 0.0.0.255 (tous les hôtes du sous-réseau 10.0.0.0/24).
- **Action** : Tout le trafic entre ces deux sous-réseaux est **autorisé**.

Exemple 2 : Bloquer le trafic HTTP (port 80) entre deux sous-réseaux

access-list 101 deny tcp 192.168.1.0 0.0.0.255 10.0.0.0 0.0.0.255 eq 80

- **Numéro de l'ACL** : 101
- **Action** : deny (bloquer le trafic correspondant à cette règle).
- **Protocole** : tcp (on cible spécifiquement le protocole TCP).
- **Adresse source** : 192.168.1.0 avec le **wildcard** 0.0.0.255 (tout le sous-réseau).
- **Adresse destination** : 10.0.0.0 avec le **wildcard** 0.0.0.255.
- **Port destination** : eq 80 (HTTP, port 80).
- **Action** : Bloque tout le trafic TCP à destination du **port 80** (HTTP) entre ces deux sous-réseaux.

Exemple 3 : Autoriser le trafic HTTPS (port 443) depuis une adresse IP spécifique

access-list 102 permit tcp host 192.168.1.10 any eq 443

- **Numéro de l'ACL** : 102
- **Action** : permit (autoriser le trafic correspondant à cette règle).

- **Protocole** : tcp (trafic TCP).
- **Adresse source** : host 192.168.1.10 (on filtre seulement pour une IP source précise, l'hôte 192.168.1.10).
- **Adresse destination** : any (n'importe quelle adresse de destination).
- **Port destination** : eq 443 (autorise seulement les connexions vers le port 443, qui est HTTPS).
- **Action** : Le trafic HTTPS provenant de **192.168.1.10** est autorisé vers n'importe quelle adresse.

Exemple 4 : Bloquer le ping (ICMP) depuis tout le réseau vers un serveur spécifique

access-list 103 deny icmp any host 10.0.0.1

- **Numéro de l'ACL** : 103
- **Action** : deny (bloquer le trafic correspondant).
- **Protocole** : icmp (le protocole ICMP utilisé pour le ping).
- **Adresse source** : any (n'importe quelle source).
- **Adresse destination** : host 10.0.0.1 (on cible un serveur spécifique).
- **Action** : Bloque le trafic **ICMP** (ping) vers le serveur **10.0.0.1** depuis n'importe quelle adresse.

Application de l'ACL sur une interface

Après avoir créé une ACL, il faut l'appliquer à une interface du routeur. Cela peut être fait en entrée (pour le trafic entrant) ou en sortie (pour le trafic sortant). La syntaxe pour appliquer une ACL est la suivante :

Router(config)# interface [interface_id]

Router(config-if)# ip access-group [numéro_ACL] [in|out]

- **interface [interface_id]** : C'est l'interface sur laquelle appliquer l'ACL (par exemple GigabitEthernet0/0).
- **ip access-group [numéro_ACL]** : Spécifie le numéro de l'ACL que vous voulez appliquer (par exemple, 100).

- **[in|out]** : in pour appliquer l'ACL au trafic entrant dans l'interface, out pour le trafic sortant.

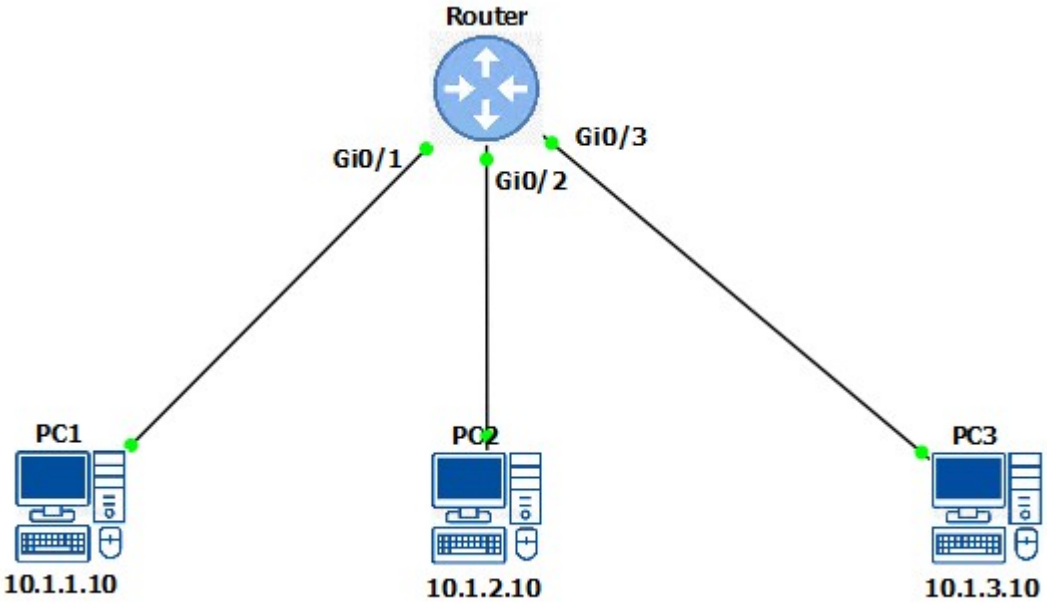
Exemple : Appliquer l'ACL 100 sur l'interface GigabitEthernet0/0 pour le trafic entrant

Router(config)# interface GigabitEthernet0/0

Router(config-if)# ip access-group 100 in

Points sur les ACLs

1. ACL STANDARD (ID = **1 à 99** et de **1300 à 1999)**.



Dans l'exemple ci-dessus, les trois ordinateurs, situés sur des segments réseau différents, communiquent entre eux. Le but est d'interdire au réseau « **10.1.1.0/24 »** de communiquer avec le réseau « **10.1.2.0/24 »** tout en ayant la possibilité de communiquer avec le réseau « **10.1.3.0/24 »**

? L'interface Gi0/1, Gi0/2 ou Gi0/3

? La règle

Etape 1 :

Pour ce faire, nous allons, sur l'interface **Gi0/2**, interdire les paquets provenant du réseau « **10.1.1.0/24 »**

Router(config)#access-list 1 deny 10.1.1.0 0.0.0.255

On précise :

- "access-list 1" on attribue **un ID à notre ACL**,
- Le blocage avec "deny",
- et enfin l'adresse IP de source (10.1.1.0) et le masque au format inversé appelé *wildcards mask* (0.0.0.225).

Etape 2 :

il faut autoriser explicitement les réseaux que l'on veut laisser passer, dans notre exemple c'est « **10.1.3.0/24 »**

Router(config)#access-list 1 permit 10.1.3.0 0.0.0.255

- "access-list 1" on attribue **un ID à notre ACL**,

- L’autorisation avec "permit",
- et enfin l'adresse IP de source (10.1.3.0) et le masque au format inversé appelé *wildcards mask* (0.0.0.225).

Etape 3 :

Il faut appliquer la règle en sortie de l’interface **Gi0/2**

Router(config)#interface gigabitEthernet 0/2

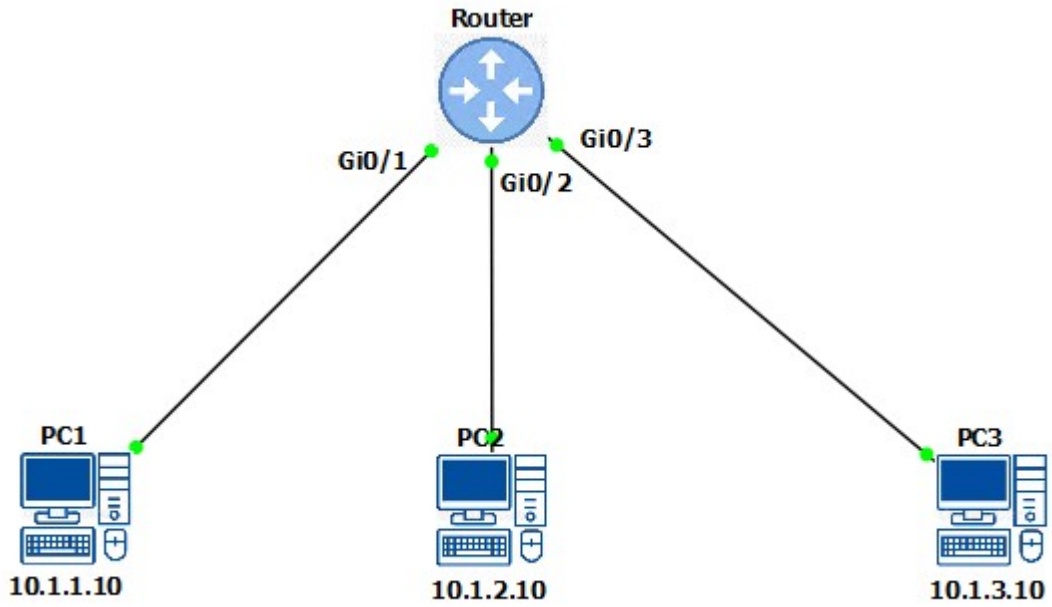
Router(config-if)# ip access-group 1 out

? Supprimer une ACL

Le « no access-list » + le Numéro de l'ACL, supprime tout le contenu de l’ACL.

1. ACL ETENDUES (ID = **100 à 199** et de **2000 à 2699)**.

N allons créer une règle qui aura pour but d’**interdire le Ping de PC2 vers le PC3, tout en l'autorisant vers le PC1,** en posant les règles sur les sous-réseaux (tous en /24). Mettons ça en place en reprenant la même topologie :



? L’interface Gi0/1, Gi0/2 ou Gi0/3

? règle

Router(config)# access-list 100 deny icmp 10.1.3.0 0.0.0.255 10.1.2.0 0.0.0.255

Router(config)# access-list 100 permit icmp 10.1.1.0 0.0.0.255 10.1.2.0 0.0.0.255

Router(config)#interface gig 0/2

Router(config-if)#ip access-group 100 in

? Expliquez les règles suivantes :

Router(config)# access-list 100 permit tcp 10.1.1.0 0.0.0.255 10.1.2.0 0.0.0.255 eq 21

access-list 101 deny tcp host 172.16.3.10 172.16.1.0 0.0.0.255 eq ftp

access-list 102 deny tcp host 172.16.2.10 host 172.16.1.100 eq www

access-list 102 permit ip any any

TD ACL

1. Questions Théoriques

1. Qu'est-ce qu'une ACL étendue et en quoi se distingue-t-elle d'une ACL standard ?

2. Quel est le rôle d'une ACL dans la gestion de la sécurité réseau ?

3. Quels sont les différents protocoles que vous pouvez spécifier dans une ACL étendue ? Donnez quelques exemples.

4. Expliquez ce qu'est un wildcard mask et comment il est utilisé dans une ACL étendue.

5. Quelle est la différence entre une ACL appliquée en entrée (inbound) et une ACL appliquée en sortie (outbound) sur une interface ?

6. Quels opérateurs peuvent être utilisés pour filtrer les ports dans une ACL étendue ?

7. Quelles sont les bonnes pratiques à suivre lors de la création d'une ACL étendue ?

8. Que se passe-t-il si une ACL ne comporte aucune règle permettant le trafic ?

2. Questions Pratiques

1. Créez une ACL étendue (numéro 110) qui bloque tout le trafic HTTP (port 80) entre le réseau 192.168.10.0/24 et le réseau 10.1.1.0/24, tout en autorisant le reste du trafic.

2. Rédigez la commande permettant de créer une ACL qui autorise uniquement le trafic ICMP (ping) provenant du réseau 172.16.1.0/24 vers l'hôte 192.168.0.10.

3. Comment appliquer une ACL étendue (numéro 105) en entrée sur l'interface GigabitEthernet0/1 ? Écrivez la commande exacte.

4. Modifiez l'ACL 115 pour qu'elle autorise tout le trafic TCP sortant depuis l'adresse IP 192.168.20.5 vers n'importe quelle adresse, uniquement pour les ports de destination compris entre 1000 et 2000.

5. Diagnostiquez le problème suivant : L'utilisateur situé dans le réseau 192.168.1.0/24 n'arrive pas à pinguer un serveur situé dans le réseau 10.0.0.0/24. Vous remarquez que la règle ACL suivante est configurée sur le routeur : access-list 101 deny ip any 10.0.0.0 0.0.0.255. Expliquez pourquoi le ping échoue et proposez une solution.

6. Créez une ACL qui autorise seulement le trafic HTTPS (port 443) provenant de l'hôte 192.168.50.100 vers n'importe quelle destination, tout en bloquant tout autre trafic provenant de cet hôte.

7. Expliquez comment vous testeriez une ACL que vous venez de configurer pour vous assurer qu'elle fonctionne correctement. Quels outils et commandes utiliseriez-vous ?

8. Écrivez une ACL qui bloque le trafic ICMP entre le réseau 192.168.5.0/24 et le réseau 172.16.0.0/16, tout en autorisant tout autre type de trafic entre ces deux réseaux.

3. Scénarios d'Étude de Cas

1. Vous gérez un réseau dans lequel vous souhaitez bloquer l'accès SSH (port 22) à un serveur spécifique (adresse IP 192.168.100.10) depuis le réseau externe 203.0.113.0/24. Comment configureriez-vous une ACL pour réaliser cela tout en permettant le reste du trafic ?

2. Un utilisateur sur le réseau 10.0.0.0/24 se plaint de ne pas pouvoir accéder à un site web hébergé sur 192.168.50.5 via HTTP. L'ACL suivante est configurée sur le routeur : access-list 120 deny tcp 10.0.0.0 0.0.0.255 192.168.50.0 0.0.0.255 eq 80. Diagnostiquez et résolvez le problème.

3. Comment pouvez-vous utiliser une ACL pour limiter l'accès à certains services réseau à des utilisateurs internes tout en permettant aux administrateurs réseau d'y accéder à distance depuis une adresse IP fixe (par exemple 198.51.100.2) ?