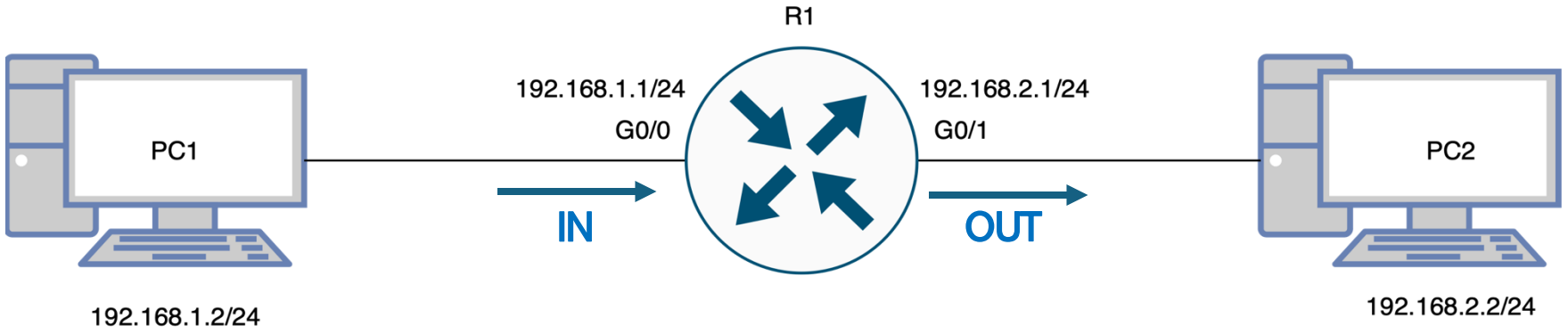
LISTE DE CONTRÔLE D'ACCES STANDARD



access-list [numéro] {permit|deny} [adresse_source] [wildcard]

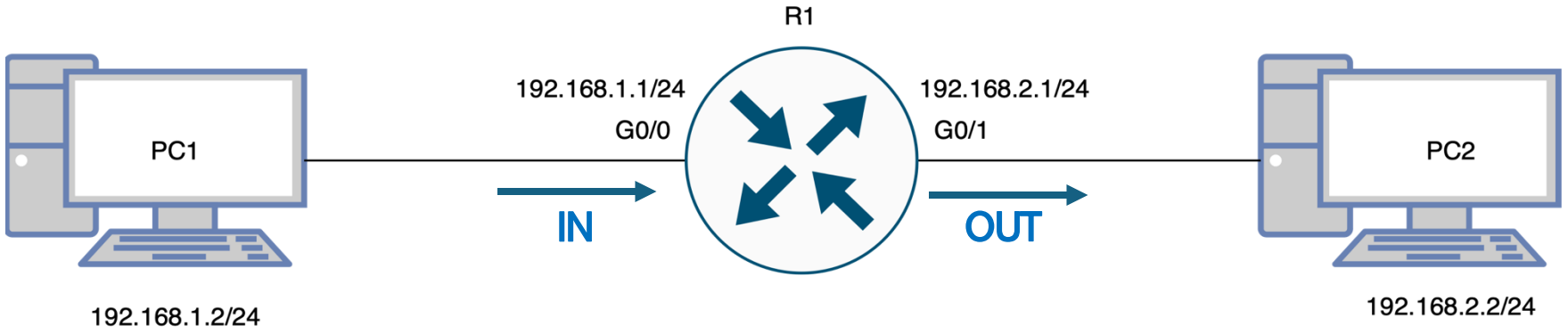
On veut interdire le flux venant de PC1 :
access-list 1 deny host 192.168.1.2
que l'on peut aussi écrire :
access-list 1 deny 192.168.1.2 0.0.0.0

On applique la liste sur une interface,
Pour les listes standard, on applique au plus proche du destinataire :

interface G0/1
ip access-group 1 out

Les **listes de contrôle d'accès (ACL) standard** sont un mécanisme de filtrage utilisé sur les routeurs pour **restreindre le trafic réseau** en fonction de **l'adresse IP source**. Elles permettent **d'autoriser ou de bloquer** des paquets **sans prendre en compte l'adresse de destination, les ports ou les protocoles**.

Les ACL standard utilisent des **numéros compris entre 1 et 99**, ainsi que **1300-1999** pour les versions étendues de cette plage. Elles sont généralement appliquées sur une interface en entrée **(inbound)** ou en sortie **(outbound)** afin de **limiter l'accès aux ressources réseau**.




```
access-list 1 deny 192.168.1.0 0.0.0.255
access-list 1 permit any
```

Les ACL sont évaluées **de haut en bas**, règle par règle, dans l'ordre où elles ont été configurées. Dès qu'un paquet correspond à une règle, **celle-ci est appliquée et les règles suivantes ne sont pas analysées**. Si aucune correspondance n'est trouvée, une règle implicite **"deny all"** bloque le trafic par défaut. Il est donc essentiel d'organiser les règles de manière stratégique : les plus spécifiques doivent être placées en premier pour éviter qu'une règle trop large ne bloque ou autorise du trafic indésirable.

CAS D'UTILISATION DES ACL STANDARD :

Les **ACL standard** sont les plus simples et permettent de filtrer le trafic **uniquement en fonction de l'adresse IP source**. Elles ne tiennent pas compte des adresses de destination, des ports ou des protocoles utilisés.

1️⃣ **Restreindre l'accès d'un réseau ou d'un hôte à certaines ressources**

✅ **Exemple : Bloquer l'accès d'un PC spécifique à tout le réseau**
Si un utilisateur **PC1 (192.168.1.10)** ne doit pas accéder au réseau **192.168.2.0/24**, on peut appliquer une **ACL standard** pour le bloquer :

```
access-list 10 deny 192.168.1.10 0.0.0.0
access-list 10 permit any
interface GigabitEthernet0/1
ip access-group 10 out
```

✓ **Résultat : PC1 ne peut pas envoyer de trafic vers l'extérieur**, mais les autres hôtes du VLAN peuvent circuler librement.

2️⃣ Autoriser uniquement certains hôtes à communiquer avec un réseau spécifique

✅ **Exemple : Autoriser uniquement un serveur spécifique à accéder à un réseau**
Seul **le serveur 192.168.1.100** peut communiquer avec le réseau **192.168.2.0/24**.

```
access-list 20 permit 192.168.1.100 0.0.0.0
access-list 20 deny any
interface GigabitEthernet0/1
ip access-group 20 out
```

✓ **Résultat : Seul le serveur 192.168.1.100 peut accéder au réseau 192.168.2.0/24.**

3 **Limiter le trafic pour améliorer la performance du réseau**

✅ **Exemple : Empêcher certains hôtes de saturer la bande passante**
Si **PC1 et PC2** (192.168.1.10 et 192.168.1.11) génèrent trop de trafic, on peut **les bloquer** :

```
access-list 40 deny 192.168.1.10 0.0.0.0
access-list 40 deny 192.168.1.11 0.0.0.0
access-list 40 permit any
interface GigabitEthernet0/1
ip access-group 40 in
```

✓ **Résultat : PC1 et PC2 ne peuvent plus envoyer de trafic via cette interface**, libérant ainsi de la bande passante.

4 Contrôler l'accès administratif aux équipements réseau

✅ Exemple : Autoriser uniquement certaines IP à se connecter en SSH à un routeur

Si seuls **les administrateurs (192.168.1.50 et 192.168.1.51)** peuvent administrer le routeur en **SSH**, on applique une ACL sur **l'interface de management** :

```
access-list 30 permit 192.168.1.50 0.0.0.0
access-list 30 permit 192.168.1.51 0.0.0.0
access-list 30 deny any
line vty 0 4
access-class 30 in
```

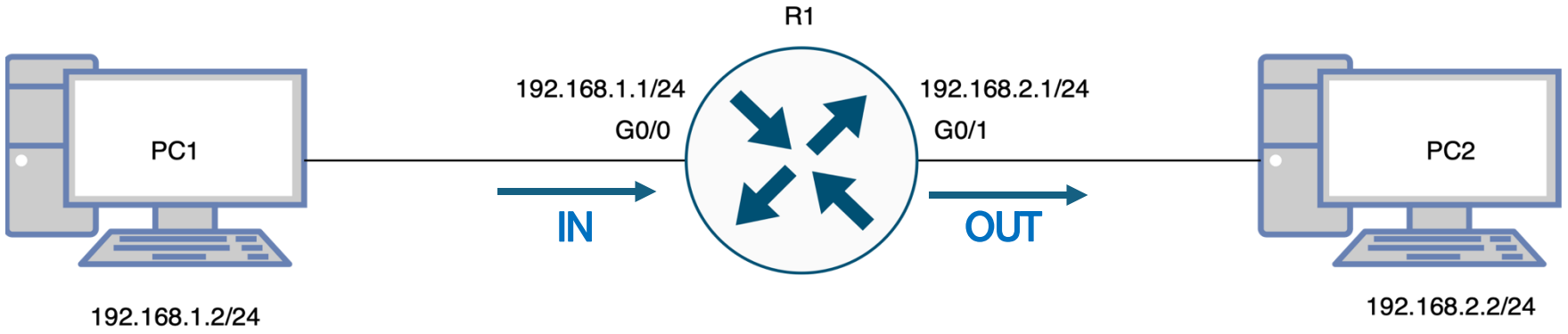
✓ **Résultat : Seuls les PC 192.168.1.50 et 192.168.1.51 peuvent accéder au routeur en SSH.**

Bonnes pratiques :

- **Placer les ACL standard aussi près que possible de la destination** pour éviter un filtrage trop large.
- Toujours **ajouter une règle "permit any"** à la fin si l'on ne veut pas bloquer tout le trafic.
- **Optimiser l'ordre des règles** en plaçant les règles spécifiques avant les règles générales.
- Ne pas oublier que **par défaut, tout ce qui n'est pas explicitement autorisé est bloqué**.

Les ACL standard sont simples mais limitées, car elles ne tiennent compte que de l'adresse source. Pour un filtrage plus précis (basé sur la destination, le protocole ou les ports), on utilisera les **ACL étendues**.

LISTE DE CONTRÔLE D'ACCES ÉTENDUE



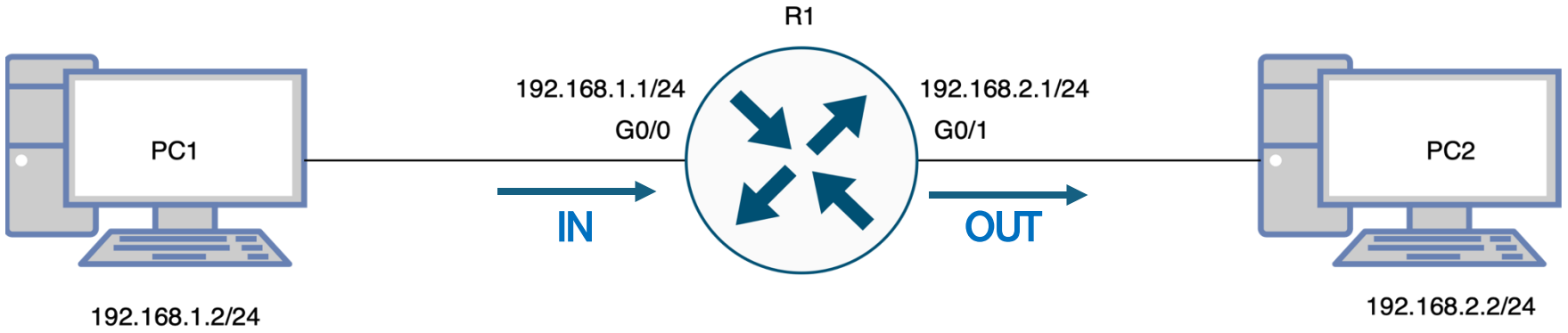
access-list [numéro] {permit|deny} [protocole] [adresse_source] [wildcard] [opérateur_port] [adresse_destination] [wildcard] [opérateur_port]

On veut interdire le flux icmp (ping) venant de PC1 vers PC2 en autorisant la réponse au ping venant de 192.168.2.0 :

access-list 100 permit icmp host 192.168.1.2 192.168.2.0 0.0.0.255 echo-reply
access-list 100 deny icmp host 192.168.1.2 192.168.2.0 0.0.0.255
access-list 100 permit ip any any

que l'on peut aussi écrire :

access-list 100 permit icmp 192.168.1.2 0.0.0.0 192.168.2.0 0.0.0.255 echo-reply
access-list 100 deny 192.168.1.2 0.0.0.0 192.168.2.0 0.0.0.255
access-list 100 permit ip any any



On applique la liste sur une interface,
Pour les listes étendues, on applique au plus proche de la source pour **optimiser les performances du réseau et éviter un trafic inutile :**

```
interface G0/1
ip access-group 1 in
```

Les listes de contrôle d’accès (ACL) étendues offrent un filtrage plus précis du trafic réseau en permettant de spécifier plusieurs critères : l’adresse IP source et destination, le protocole (TCP, UDP, ICMP, etc.), ainsi que les ports concernés. Elles permettent ainsi de contrôler finement le trafic et de mettre en place des règles de sécurité avancées.

Les ACL étendues utilisent des numéros compris entre 100 et 199, ainsi que 2000-2699 pour les versions numérotées étendues. Contrairement aux ACL standard, qui filtrent uniquement en fonction de l’adresse source, les ACL étendues permettent de restreindre l’accès à des services spécifiques ou d’empêcher certains types de connexions entre différentes parties du réseau.

Comme les ACL standard, elles sont évaluées de haut en bas, dans l’ordre où elles ont été définies. Dès qu’une règle correspond au paquet, elle est appliquée et les règles suivantes ne sont plus prises en compte. Si aucune règle ne correspond, le paquet est bloqué par défaut à cause de la règle implicite "deny all". Il est donc crucial de bien organiser les règles, en plaçant les plus spécifiques en premier pour éviter des filtrages involontaires.

CAS D'UTILISATION DES ACL ÉTENDUES :

Les **ACL étendues** permettent un **filtrage avancé** basé sur :

- **L'adresse IP source**
- **L'adresse IP destination**
- **Le protocole (TCP, UDP, ICMP, etc.)**
- **Les ports (ex : HTTP, SSH, FTP, etc.)**

Elles sont **placées près de la source** pour éviter le trafic inutile sur le réseau.

1 Bloquer l'accès à un service spécifique (ex : HTTP)

✅ **Cas d'usage :** On veut empêcher le réseau **192.168.1.0/24** d'accéder à **tous les serveurs web (HTTP, port 80)** sur Internet.

```
access-list 101 deny tcp 192.168.1.0 0.0.0.255 any eq 80
access-list 101 permit ip any any
interface GigabitEthernet0/1
ip access-group 101 in
```

✓ **Effet : Les PC du réseau 192.168.1.0/24 ne peuvent pas naviguer sur Internet**, mais tout le reste du trafic est autorisé.

2️⃣ **Autoriser uniquement un serveur spécifique à communiquer avec une base de données**

✅ **Cas d'usage** : Seul le serveur **192.168.1.100** peut se connecter à la base de données **MySQL (port 3306)** du serveur **192.168.3.200** dans la DMZ.

```
access-list 102 permit tcp 192.168.1.100 0.0.0.0 192.168.3.200 0.0.0.0 eq 3306
access-list 102 deny tcp any any eq 3306
access-list 102 permit ip any any
interface GigabitEthernet0/1
ip access-group 102 in
```

✓ **Effet : Seul le serveur 192.168.1.100 peut interroger la base de données**, et toutes les autres connexions MySQL sont bloquées.

3 Empêcher un réseau de pinguer un autre réseau

✅ **Cas d'usage** : Le **réseau extérieur (192.168.2.0/24)** ne doit pas pouvoir **pinguer** le VLAN Agents (192.168.1.0/24).

```
access-list 103 deny icmp 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255 echo
access-list 103 permit ip any any
interface GigabitEthernet0/0
ip access-group 103 in
```

✓ **Effet : Les pings depuis l'extérieur vers les PC du VLAN Agents sont bloqués.**

4 Restreindre l'accès SSH à un routeur

✅ **Cas d'usage :** Seuls les administrateurs **192.168.1.50 et 192.168.1.51** peuvent se connecter en SSH au routeur (192.168.1.1).

```
access-list 104 permit tcp 192.168.1.50 0.0.0.0 192.168.1.1 0.0.0.0 eq 22
access-list 104 permit tcp 192.168.1.51 0.0.0.0 192.168.1.1 0.0.0.0 eq 22
access-list 104 deny ip any any
line vty 0 4
access-class 104 in
```

✓ **Effet : Seuls les administrateurs définis peuvent accéder au routeur en SSH.**

5 Autoriser l'accès aux serveurs web de la DMZ depuis l'extérieur

✅ **Cas d'usage** : Seuls les serveurs web de la **DMZ (192.168.3.0/24)** doivent être accessibles depuis l'extérieur en **HTTP et HTTPS**.

```
access-list 105 permit tcp any 192.168.3.0 0.0.0.255 eq 80
access-list 105 permit tcp any 192.168.3.0 0.0.0.255 eq 443
access-list 105 deny ip any 192.168.3.0 0.0.0.255
access-list 105 permit ip any any
interface GigabitEthernet0/0
ip access-group 105 in
```

✓ **Effet : Seuls les ports 80 (HTTP) et 443 (HTTPS) des serveurs de la DMZ sont accessibles depuis l'extérieur.**