

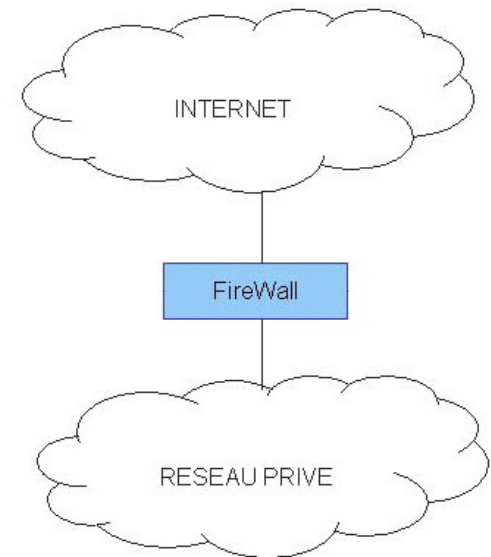


Sécurité des réseaux

Filtrage ACL

Que sont les ACL ?

- **ACL** (*Access Control Lists*) ou
 - Listes de contrôle d'accès ou
 - Listes d'accès (*Access Lists*)...
- Liste ordonnée d'ACEs (*Access Control Entries*)
 - Règles de filtrage
 - D'acceptation ou de refus
 - Sur les paquets entrants ou sortants
- Permettent aux routeurs de contrôler le trafic en le filtrant



Que contrôlent les ACL ?

- Règles
 - **D'acceptation** ou de **refus**
 - Sur les paquets **entrants** ou **sortants**

- Le contrôle peut se faire sur
 - L'adresse d'origine (source)
 - L'adresse de destination
 - Le numéro de port source
 - Le numéro de port destination
 - Le protocole transporté par la trame Ethernet (IP, ICMP, ARP...)
 - Les protocoles de couches supérieures (TCP, UDP...)
 - D'autres paramètres (horaires par exemple...)



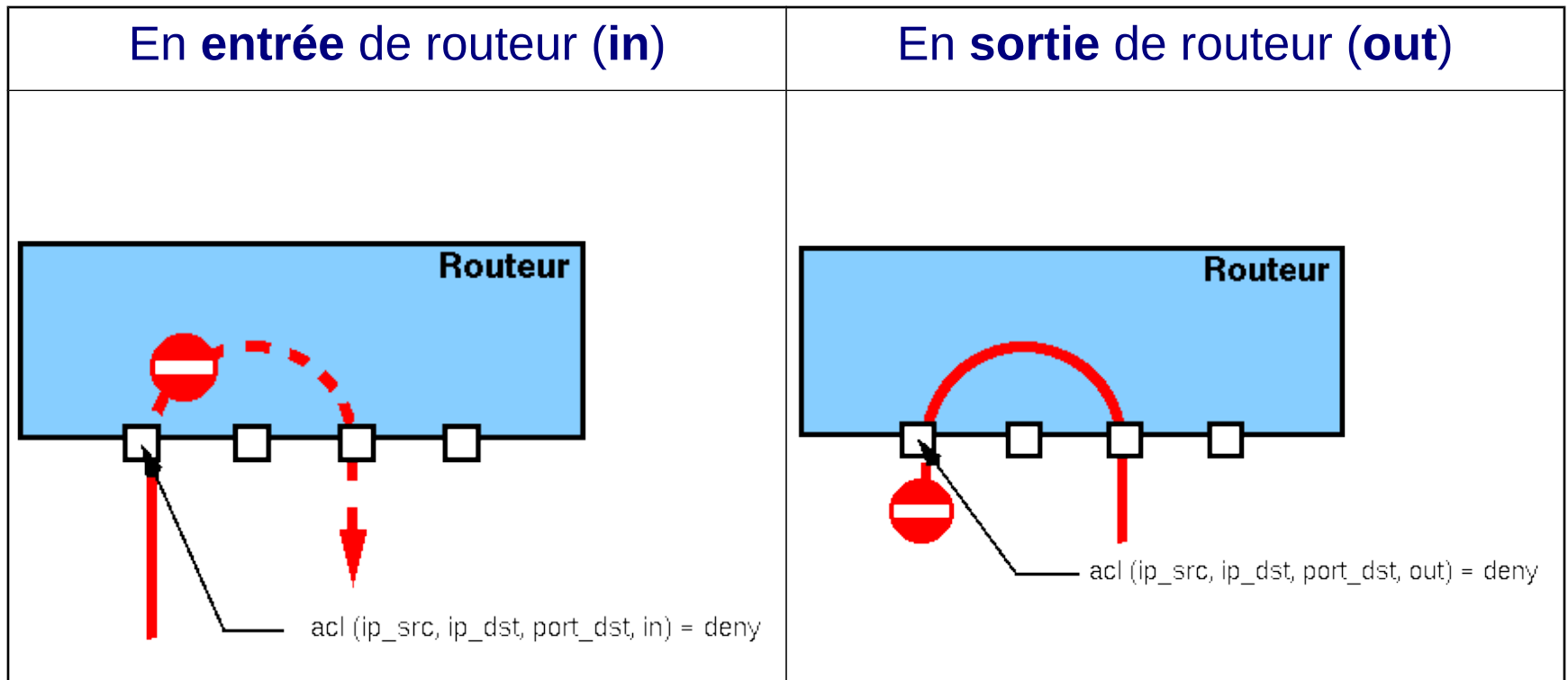
- ▶ Les types d'ACL proposés sont les suivants :
 - ▶ les **ACL standards** qui filtrent sur l'adresse source ;
 - ▶ les **ACL étendues** qui filtrent sur l'adresse source, l'adresse destination ainsi que les ports sources et destination ;
 - ▶ les **ACL lock and Key** se mettent en place après authentification de l'utilisateur (en telnet) ;
 - ▶ les **named ACL** sont des ACL étendues qui reçoivent un nom au lieu d'un numéro ;
 - ▶ les **ACL reflexives** utilisent les informations de session pour laisser entrer les paquets de retour correspondant aux paquets envoyés ;
 - ▶ les **time-based ACL** sont actives sur une plage de temps donnée ;
 - ▶ les **ACL Context-based access control** utilisent les informations de session pour autoriser à la demande et en fonction du sens d'initialisation le passage du trafic.

Les ACL sont de plusieurs types (CISCO) :

- ACL **standard** (« *access list* » simple)
 - Numérotée de 1 à 99
 - Ne prend en charge que de l'IP
 - La vérification ne porte que sur l'adresse IP source

- ACL **étendue** - EACL (*Extend ACL*, ACLE « *access list* » étendue...)
 - Numérotée de 100 à 199
 - On peut spécifier le protocole (IP, TCP, UDP, ICMP)
 - La vérification porte sur les adresses source et/ou destination, ports...

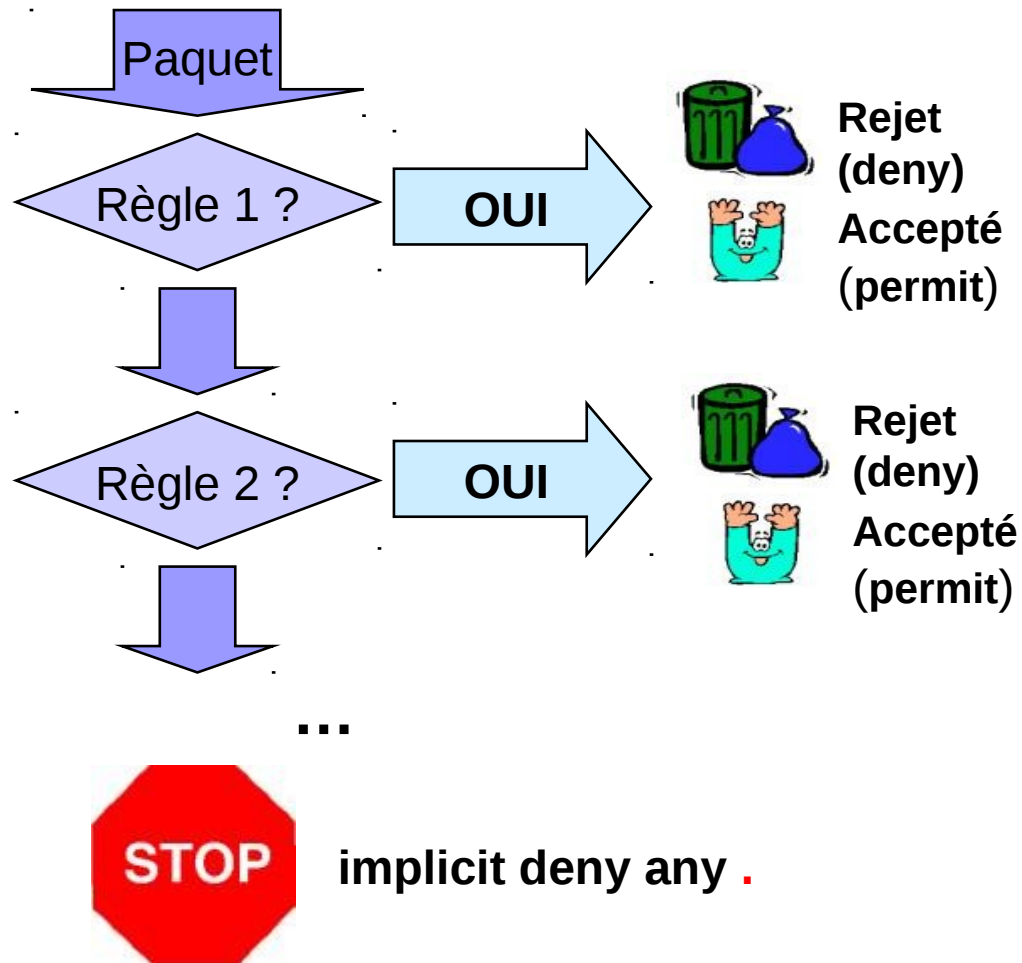
Où se fait le filtrage avec des ACL ?



Comment fonctionnent les ACL ?

- Les paquets sont évalués en ordre **séquentiel** et **logique**
- Si un en-tête de paquet **correspond** à une instruction, les suivantes sont ignorées, et le paquet est **accepté** ou **refusé** selon l'instruction
- Si l'en-tête **ne correspond pas** à une instruction, on passe à l'instruction suivante...
- Si on atteint la **fin de la liste**, le paquet est en principe implicitement **rejeté** (*implicit deny any*)
- **L'ordre** des instructions de toute liste d'accès est donc **significatif**

Comment fonctionnent les ACL ?



Comment rédiger les ACL ?



Rejet
(deny)
Accepté
(permit)

- **Organiser** ses listes d'accès !
- Préparer **AVANT** d'installer (avec CISCO « difficile » à corriger – impossible de supprimer une seule ligne - toute l'ACL est supprimée !),
- Placer les conditions les **plus spécifiques avant les plus générales**,
- Placer les conditions qui se présentent le **plus fréquemment, avant les autres**,
- **Au moins une** instruction **permit** dans la liste (sinon... comme on termine par un **implicit deny any**... on risque de refouler les paquets qui ne sont traités par aucune règle !)

Comment créer une ACL ?

- La commande de création est **access-list**

- ACL standard

`access-list numéro_acl { deny | permit } ip-source source-masque`

Ex : **access-list 1 deny 172.16.16.0 0.0.0.255**

- ACL étendue

`access-list numéro_acl { deny | permit } protocole ip-source source-masque ip-destination destination-masque condition port-source port-destination`

Ex : **access-list 101 deny tcp 172.16.4.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 20**

Qu'est ce que le masque ?

- Le masque (pas confondre avec *masque réseau*) repère les bits de l'adresse qui nécessitent une vérification **impérative** et ceux qui peuvent être **ignorés**
- un masque à **0** signifie « **Tester** la valeur du bit correspondant »
- un masque à **1** signifie « **Ignorer** la valeur du bit correspondant »
- Ex : interdire une @ **IP précise** (121.32.45.128). Il faut **vérifier tous les bits** de l'adresse et donc appliquer le **masque 0.0.0.0**
- Ex : interdire **tous** les hosts d'un réseau (121.0.0.0). Il faut vérifier que le réseau est 121 et donc appliquer le **masque 0.255.255.255**

Comment définir un masque ?

- Exemple : on veut vérifier l'appartenance ou non du paquet à tester à une plage d'adresses 172.30.16.0 à 172.30.31.0
- Les deux premiers octets de l'adresse IP sont **identiques** (172.30)
- Les bits correspondants seront à **vérifier** et donc mis à **0** dans le masque
- En ce qui concerne le troisième octet de l'adresse IP
 - 16 en binaire = 0001 0000
 - 31 en binaire = 0001 1111
- Les bits sont donc différents à partir du 5ème bit du 3ème octet.
- Les 4 premiers bits seront à **vérifier** et donc mis à **0** dans le masque
- Les bits restants pourront être **ignorés** et donc mis à **1** dans le masque
- Le masque générique de filtrage est donc 0.0.15.255

Comment activer une ACL ?

- Avant d'activer une ACL il faut choisir l'interface sur laquelle elle s'applique

Ex : **interface fastethernet 0/0**

- L'activation de l'ACL sur une interface se fait par la commande

`ip access-group numéro_acl { in | out }`

Ex : **ip access-group 1 out**

Comment enlever les ACL ?

- Se fait en deux étapes :
- Désaffecter l'ACL de l'interface par la commande

```
Router(config-if)# no ip access-group numéro_acl { in | out }
```

Ex : **no ip access-group 1 out**

- Supprimer l'ACL par la commande

```
Router(config)# no access-list numéro_acl { permit | deny } { any | adresse }
```

Ex : **no access-list 1 deny any**

Attention : il est impossible de ne supprimer qu'une ligne de l'ACL

Pourquoi ACL « étendue » ?

- **ACL étendue** - EACL (*Extend ACL*, ACLE « *access list* » étendue...)
 - Permettent de contrôler plus finement le trafic
 - La vérification porte sur les adresses source et/ou destination, ports...
 - Peut spécifier un protocole plus précis (IP, TCP, UDP, ICMP)
 - Numérotée de 100 à 199 .

ACL ou ACLE ?

- ACL **standard** - filtrage simple à mettre en oeuvre
- ACL **étendue** - Permettent de contrôler plus finement le trafic .

Standard	Étendues
Filtrent uniquement sur la base de l'adresse source	Filtrent sur la base des adresses source et destination, et des numéros de port adresse et destination
Acceptent ou refusent toute la « pile » de protocoles TCP/IP	Spécifient un protocole « IP » (TCP, UDP, IP, ICMP) et un numéro de port
Plage de 1 à 99	Plage de 100 à 199

Pourquoi filtrer par port ?

- Le filtrage du port permet le filtrage de certains protocoles ! .

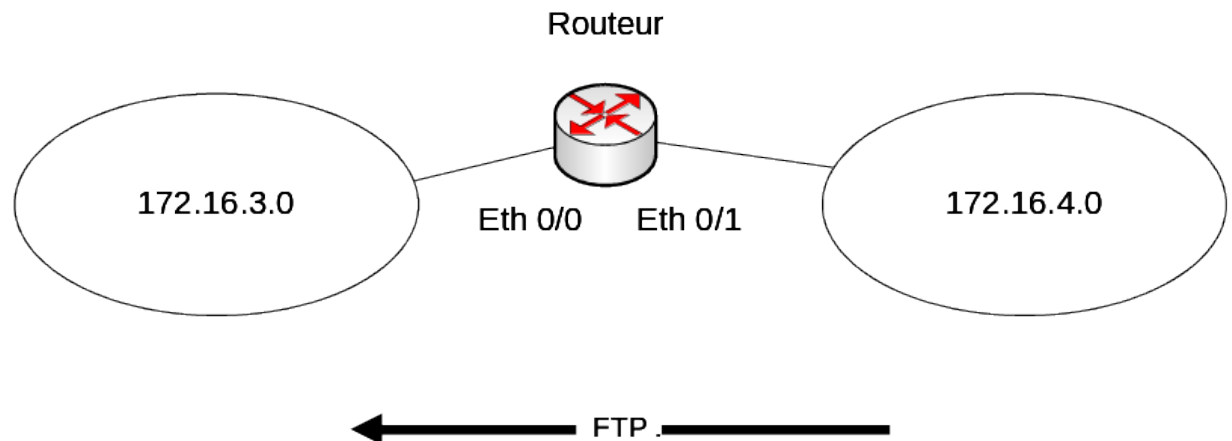
Numéro de port réservé	Protocole IP
20	Données FTP (<i>File Transfer Protocol</i>)
21	Programme FTP
23	Telnet
25	SMTP (<i>Simple Mail Transfer Protocol</i>)
69	TFTP (<i>Trivial FTP</i>)
53	DNS (<i>Domain Name System</i>)

Quelle syntaxe pour l'ACLe ?

- **Router(config)#** access-list numéro-de-liste {permit | deny } protocole adresse-source masque-source [opérateur] adresse-destination masque-destination [opérateur [port]] [established] [log]
- **opérateur :**
 - lt (less than - inférieur à),
 - gt (greater than - supérieur à),
 - eq (equal to - égal à),
 - neq (non equal to - non égal à)
- **established**
 - sert pour un flux TCP entrant, et permet au trafic TCP de passer si le paquet utilise une **connexion établie** (bits ACK positionnés)

Exemple d'ACL étendue

- Router(config)# access-list 101 deny tcp 172.16.4.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 21
- Router(config)# access-list 101 deny tcp 172.16.4.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 20
- Router(config)# access-list 101 permit ip any any
- Router(config)# interface fastethernet 0/0
- Router(config-if)# ip access-group 101 out



ACL nommée


- Il est possible de supprimer qu'une seule ligne
- Notation
 - R1(config)#ip access list extended toto
 - R1(config-ext-nacl)#permit tcp ...
- Suppression d'une ligne
 - R1(config)#ip access list extended toto
 - R1(config-ext-nacl)#no permit tcp...

ACL réflexives

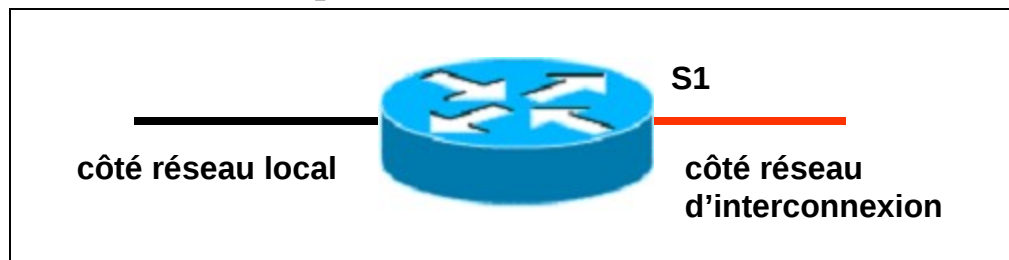
- Permet de filtrer les paquets IP en fonction des informations de session (qui a commencé ?) des couches supérieures
- On peut ainsi autoriser un certain trafic, seulement si il a été initié depuis l'intérieur du réseau
- On pouvait déjà obtenir ce fonctionnement avec des ACL étendues, en utilisant l'option *established*, mais cette option ne vaut que pour TCP (UDP est en effet un protocole non connecté)
- Les ACL réflexives permettent de faire ce type de filtrage avec TCP, mais aussi UDP et ICMP

- Les sessions TCP sont suivies grâce aux bits ACK, RST et FIN des en-têtes TCP
- La fin de la session TCP est repérée de la façon suivante :
 - quand le bit FIN de l'en-tête TCP est placé à 1, le routeur devine que la session va se terminer, il attend 5 secondes pour laisser le temps à l'hôte et au serveur de terminer leur session, puis il bloque le trafic
 - quand le bit RST est mis à 1, le routeur détecte une interruption abrupte de session et bloque immédiatement le trafic
 - par défaut au bout d'un certain temps (paramétrable) d'inactivité pour cette session

- Les sessions UDP sont suivies par les couples
 - @IP source/destination
 - n° port source/destination
- La fin de la session ne peut être détectée que par défaut au bout d'un certain temps d'inactivité.

- 
- Il y a deux restrictions à l'utilisation des ACL réflexives :
 - elles doivent être utilisées uniquement avec les ACLs étendues (cela ne marche pas avec les standards qui ne portent pas mention des n° de port)
 - cette technique ne fonctionne pas avec les applications qui changent de numéro de port en cours de session (par exemple FTP en mode actif)

ACL réflexive : exemple



```
interface Serial 1
  description Acces à Internet
  ip access-group inboundfilters in
  ip access-group outboundfilters out
```

utilisation d'ACL nommées

```
!
ip reflexive-list timeout 120
```

sessions considérées comme inactives et par conséquent interdites au bout de 120 secondes

```
!
ip access-list extended
  outboundfilters
  permit tcp any any reflect tcptraffic
```

définition de l'ACL nommée outboundfilters, elle ne contient qu'une instruction : autoriser tout le trafic IP, mais en pistant au passage les sessions sous le nom tcptraffic

```
!
ip access-list extended
  inboundfilters
  permit bgp any any
  permit eigrp any any
  deny icmp any any
  evaluate tcptraffic
```

définition de l'ACL nommée inboundfilters :

- on autorise tout le trafic BGP
- on autorise tout le trafic EIGRP
- on interdit tout trafic ICMP
- tout le reste est évalué selon la règle tcptraffic

Conseils

- La création, la mise à jour, le débogage nécessitent beaucoup de temps et de rigueur dans la syntaxe.

■ Il est donc conseillé

- De créer les ACL à l'aide d'un éditeur de texte et de faire un copier/coller dans la configuration du routeur
- Placer les **ACLe** au plus près de la source du paquet que possible pour le détruire le plus vite possible
- Placer les **ACL standard** au plus près de la destination sinon, vous risquez de détruire un paquet trop top
- Placer la règle la plus spécifique en premier
- Avant de faire le moindre changement sur une ACL, désactiver sur l'interface concerné celle-ci (no ip access-group)