Les ACL Cisco

Par Dr RIAHLA Mohamed Amine

Les ACL

- Permettent de filtrer les accès entre les différents réseaux ou de filtrer les accès au routeur lui même.
- Les paramètres contrôlés sont:
  - Adresse source
  - Adresse destination
  - Protocole utilisé
  - Numéro de port

- Peuvent être appliquées sur le traffic **entrant** ou **sortant**. Il y a deux actions: soit le traffic est **interdit**, soit le traffic est **autorisé**.

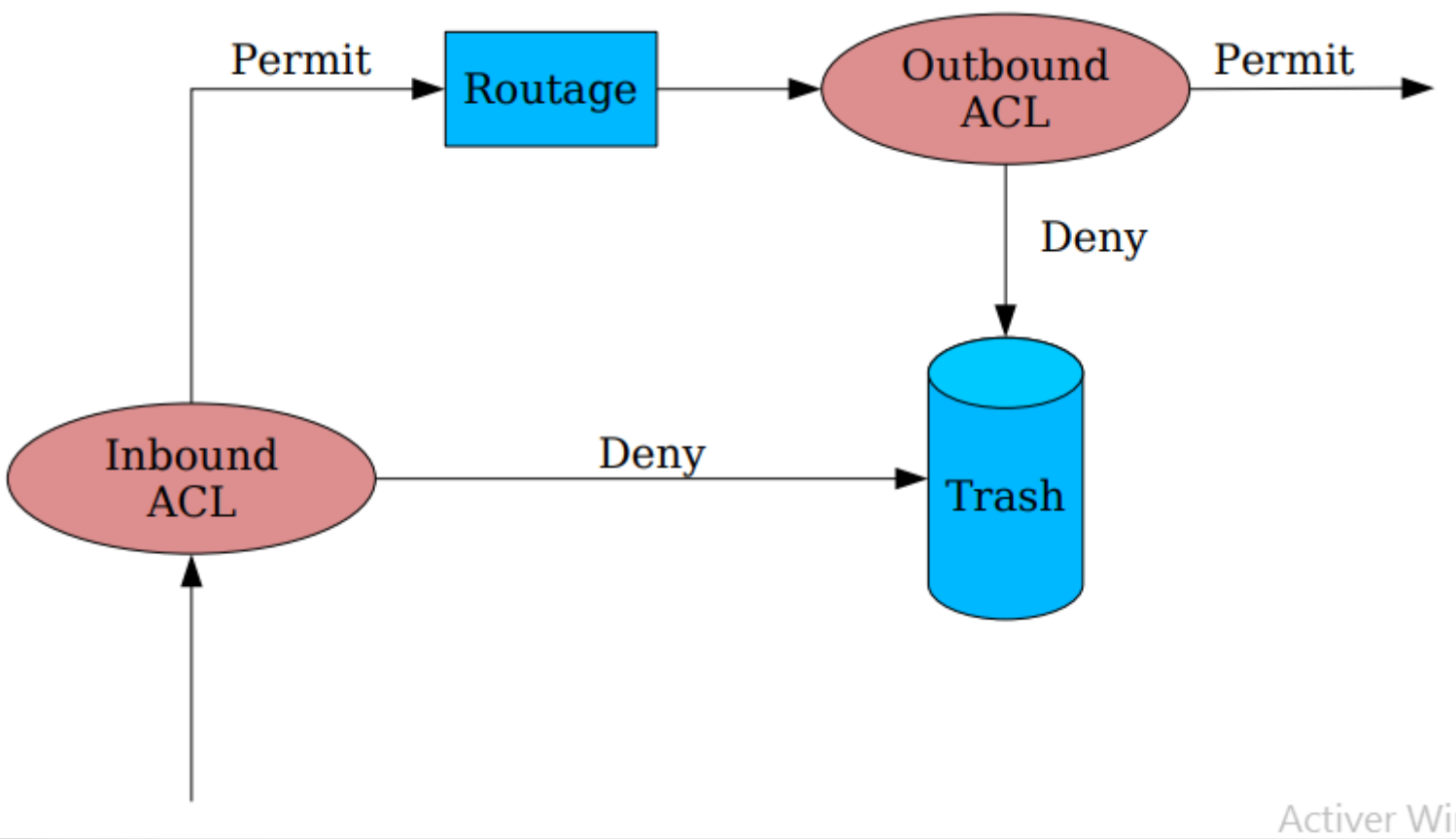
Les ACL

- Sont prises en compte de façon séquentielle.
- **Conseil**: Les instructions les plus **précises en premier** et l'instruction la plus **générique en dernier**.
- **Par défaut**, tout le trafic est **interdit**.
- Dès qu'une règle correspond au trafic, l'action définie est appliquée, le reste de l'ACL n'est pas analysé
- Il est possible d'appliquer au maximum une ACL par interface et par sens (input/output),
- Les critères sont définit sur les informations contenues dans les en-têtes IP, TCP ou UDP

Appliquer Les ACL

- Créer l'ACL puis l’appliquer à une interface en entrée ou en sortie (in ou out).
- Si l'ACL doit être modifiée, il sera nécessaire de supprimer celle ci puis de la recréer entièrement.
- Une façon pratique de faire est de conserver l'acl dans un fichier texte puis de faire un copier/coller.

Les ACL


Permit
Routage
Outbound ACL
Permit
Deny
Inbound ACL
Deny
Trash

ACL (masques**)**

- Des masques ont été définis pour pouvoir identifier une ou plusieurs adresses IP en **une seule définition** (pour sélectionner des plages d'adresses)
  - Seuls les bits de l'IP qui correspondent à 0 du masque sont vérifiés.
  - 0.0.255.255: seuls les 2 premiers octets doivent être examinés
  - deny 10.1.3.0 avec 0.0.0.255 : refus de all les IP débutant par 10.1.3,
  - 0.0.0.0 0.0.0.0, toutes les adresses sont concernées (**any**).
  - 192.168.2.3 255.255.255.255, on vérifie uniquement l'hote ayant l'IP 192.168.2.3 (**host**)
- Il existe 2 types d'ACL
  - **Standard** : uniquement sur les IP sources
  - **Etendue** : sur tous les champs des en-têtes IP, TCP et UDP

ACL Standard

- Permettent d’analyser du trafic en fonction de l’adresse IP source,
- Sont à appliquer le plus **proche** possible de la **destination** en raison de leur faible précision.
- Elle est de la forme:
  - **access-list** numéro-de-la-liste **{permit|deny}** {host|source source-wildcard|any}
  - **access-list** number **remark** test **(commentaire)**
- Le numéro de l'acl standard est compris entre 1 et 99 ou entre 1300 et 1999.

ACL Standard

- Activation d'une ACL sur une interface
  - **ip access-group** [number | name [**in** | **out**]]

- Visualiser les ACL
  - **show access-lists** [number | name] : toutes les ACL quelque soit l'interface
  - **show ip access-lists** [number | name] : les ACL uniquement liés au protocole IP

ACL Standard

interface Ethernet0
ip address 172.16.1.1 255.255.255.0
ip access-group 1 out

access-list 1 remark stop tous les paquets d'IP source 172.16.3.10
access-list 1 deny 172.16.3.10 0.0.0.0
access-list 1 permit 0.0.0.0 255.255.255.255

- **access-list 1 deny 172.16.3.10 0.0.0.0**
  - Refuse les paquets d'IP source 172.16.3.10
  - Le masque (également appelé **wildcard** mask) signifie ici que tous les bits de l'adresse IP sont significatifs
- **access-list 1 permit 0.0.0.0 255.255.255.255**
  - Tous les paquets IP sont autorisés
  - Le masque 255.255.255.255 signifie qu'aucun bit n'est significatif

ACL Standard

interface Ethernet0
ip address 172.16.1.1 255.255.255.0
ip access-group 1 out

access-list 1 remark stop tous les paquets d'IP source 172.16.3.10
access-list 1 deny host 172.16.3.10
access-list 1 permit any

- Une notation améliorée est possible pour remplacer
  - le masque 255.255.255.255 qui désigne une machine: **host**
  - 0.0.0.0 avec le wildcard masque à 255.255.255.255 qui désigne tout le monde : **any**

ACL Standard

```
interface Ethernet0
ip address 172.16.1.1 255.255.255.0
ip access-group 1 out

interface Ethernet1
ip address 172.16.2.1 255.255.255.0
ip access-group 2 in

access-list 1 remark stop tous les paquets d'IP source 172.16.3.10
access-list 1 deny host 172.16.3.10
access-list 1 permit any

access-list 2 remark Autorise que les trames d'IP source 172.16.3.0/24
access-list 2 permit 172.16.3.0 0.0.0.255
```

ACL nommée standard

ACL « numériques »: ACL identifiées par un nombre.

1 à 99 : ACL Standard
100 à 199 : ACL Etendue
1300 à 1999 : ACL Standard
2000 à 2699 : ACL Etendue

ACL « nommées »: ACL identifiées par un nom sous la forme d'une chaîne de caractères alphanumériques.

Exemple:

```
R1(config)#ip access-list standard monACL
R1(config-std-nacl)#permit 192.168.0.0 0.0.0.255
R1(config-std-nacl)#permit 192.168.1.0 0.0.0.255
R1(config-std-nacl)#deny 192.168.0.0 0.0.3.255
R1(config-std-nacl)#permit any
R1(config-std-nacl)#exit
```

ACL nommée standard

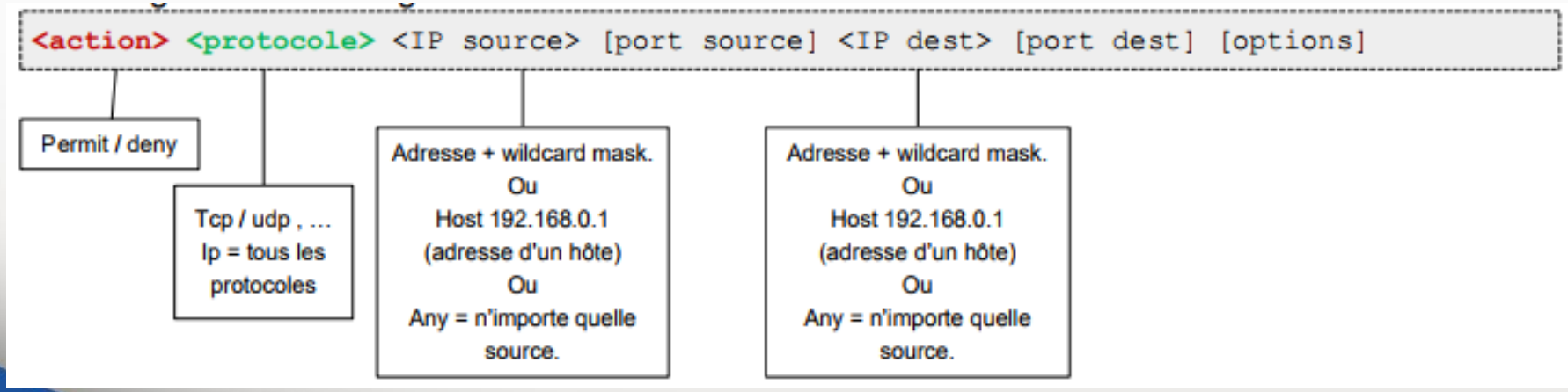
R1#show access-lists
Standard IP access list 1
10 permit 192.168.0.0, wildcard bits 0.0.0.255
20 permit 192.168.1.0, wildcard bits 0.0.0.255
30 deny 192.168.0.0, wildcard bits 0.0.3.255
40 permit any

Standard IP access list monACL
10 permit 192.168.0.0, wildcard bits 0.0.0.255
20 permit 192.168.1.0, wildcard bits 0.0.0.255
30 deny 192.168.0.0, wildcard bits 0.0.3.255
40 permit any

- Les ACLs sont identiques. Tout le trafic provenant du réseau 192.168.0.0/22 est bloqué à l'exception des deux subnets 192.168.0.0/24 et 192.168.1.0/24.

ACL étendue

- Permettent filtrer des paquets en fonction de
  - l'adresse de destination IP
  - Du type de protocole (TCP, UDP, ICMP, IGRP, IGMP, ...)
  - Port source
  - Port destination
  - ...
- A appliquer le plus proche possible de la source



ACL étendue ‘détail’

access-list numéro de la liste {deny|permit} protocole source masque-source [operateur [port]] destination masque-destination [operateur [port]][established][log]

- Le numéro de l'acl étendue est compris entre 100 et 199 ou entre 2000 et 2699.
- Quelques opérateurs:
  - **eq** : égal
  - **neq** : différent
  - **gt** : plus grand que
  - **lt** : moins grand que
- Comme dans l’ACL standard, Il est possible de nommer les acls: il suffit de préciser dans la commande,

ACL étendue ‘Exemples’

- **access-list 101 deny ip any host 10.1.1.1**

Refus des paquets IP à destination de la machine 10.1.1.1 et provenant de n'importe quelle source

- **access-list 101 deny tcp any gt 1023 host 10.1.1.1 eq 23**

Refus de paquet TCP provenant d'un port > 1023 et à destination du port 23 de la machine d'IP 10.1.1.1

- **access-list 101 deny tcp any host 10.1.1.1 eq http**

Refus des paquets TCP à destination du port 80 de la machine d’IP 10.1.1.1

ACL ‘Exemples’

R1(config)#ip access-list extended monACLextended
R1(config-ext-nacl)#permit tcp any host 192.168.1.100 eq 80
R1(config-ext-nacl)#permit icmp 192.168.0.0 0.0.0.255 host 192.168.1.100
R1(config-ext-nacl)#exit

- Tout trafic HTTP à destination de 192.168.1.100 est autorisé.
- Tout le traffic ICMP provenant de 192.168.0.0/24 à destination de 192.168.1.100 est autorisé.
- Tout autre trafic est rejeté,

ACL étendue ‘Modifier une ACL’

```
R1#show access-list 1
Standard IP access list 1
    10 permit 192.168.0.0, wildcard bits 0.0.0.255
    20 permit 192.168.1.0, wildcard bits 0.0.0.255
    30 deny   192.168.0.0, wildcard bits 0.0.3.255
    40 permit any
R1#configure terminal
R1(config)#ip access-list standard 1
R1(config-std-nacl)#no 20
R1(config-std-nacl)#15 permit 192.168.1.0 0.0.0.127
R1(config-std-nacl)#^Z
R1#show access-list 1
Standard IP access list 1
    10 permit 192.168.0.0, wildcard bits 0.0.0.255
    15 permit 192.168.1.0, wildcard bits 0.0.0.127
    30 deny   192.168.0.0, wildcard bits 0.0.3.255
    40 permit any
R1#
```

Entre en mode de configuration d'ACL

Supprime la règle portant le n° de séquence 20

Ajoute une régle avec le n° de séquence 15

ACL étendue ‘Supprimer une ACL’

```
R1#show access-lists
Standard IP access list 1
    10 permit 192.168.0.0, wildcard bits 0.0.0.255
    15 permit 192.168.1.0, wildcard bits 0.0.0.127
    30 deny   192.168.0.0, wildcard bits 0.0.3.255
    40 permit any
Standard IP access list monACL
    10 permit 192.168.0.0, wildcard bits 0.0.0.255
    20 permit 192.168.1.0, wildcard bits 0.0.0.255
    30 deny   192.168.0.0, wildcard bits 0.0.3.255
    40 permit any
Extended IP access list 100
    10 permit tcp any host 192.168.1.100 eq www
    20 permit icmp 192.168.0.0 0.0.0.255 host 192.168.1.100
Extended IP access list monACLextended
    10 permit tcp any host 192.168.1.100 eq www
    20 permit icmp 192.168.0.0 0.0.0.255 host 192.168.1.100
Extended IP access list test
R1#configure terminal t
R1(config)#no access-list 100
R1(config)#no ip access-list standard monACL
R1(config)#^Z
R1#show access-lists
Standard IP access list 1
    10 permit 192.168.0.0, wildcard bits 0.0.0.255
    15 permit 192.168.1.0, wildcard bits 0.0.0.127
    30 deny   192.168.0.0, wildcard bits 0.0.3.255
    40 permit any
Extended IP access list monACLextended
    10 permit tcp any host 192.168.1.100 eq www
    20 permit icmp 192.168.0.0 0.0.0.255 host 192.168.1.100
Extended IP access list test
R1#
```

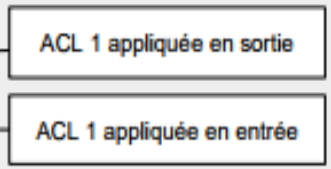
Suppression d'une ACL numérotée

Suppression d'une ACL nommée

ACL étendue ‘autres commandes’

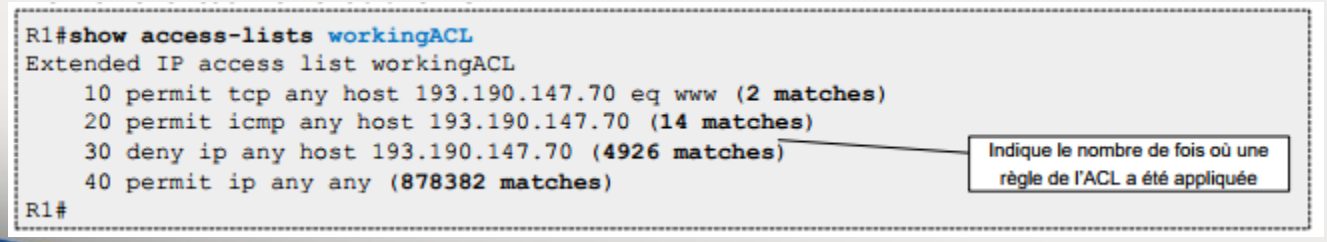
- Appliquer une ACL sur une interface
  - **R1(config)#interface fastethernet 0/0**
  - **R1(config-if)#ip access-group 1 in** (trafic entrant sur l’interface)
  - OU
  - **R1(config-if)#ip access-group 1 out** (trafic sortant de l’interface)
- Vérification des ACLs appliquées sur une interface

```
R1#show ip interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  Internet address is 192.168.0.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is 1
  Inbound  access list is 1
  Proxy ARP is enabled
  Local Proxy ARP is disabled
  Security level is default
  < … suite de l’affichage omis … >
R1#
```



ACL étendue ‘autres commandes’

- Désactiver une ACL sur une interface
  - **R1(config)#interface fastethernet 0/0**
  - **R1(config-if)#no access-group 1 in** (trafic entrant sur l’interface)
  - OU
  - **R1(config-if)#no access-group 1 out** (trafic sortant de l’interface)
- Appliquer une ACL sur les lignes VTY
  - **R1(config)#line vty 0 4**
  - **R1(config-if)#access-class 1 in**
- Désactiver une ACL sur les lignes VTY
  - **R1(config)#line vty 0 4**
  - **R1(config-if)#no access-class 1 in**
- Vérifier le fonctionnement d’une ACL



L'accès au Telnet avec une ACL

- Pour utiliser une ACL dans le but de contrôler l'accès au telnet (donc au vty)
  - **access-class number { in | out }**

```
line vty 0 4
login
password Cisco
access-class 3 in
!
!
access-list 3 permit 10.1.1.0 0.0.0.255
```

ACL ‘Exercices’

- Créer access-list nommée reseau-telecom
- Autoriser la machine 192.168.2.12 à se connecter via ssh à toutes les machines du réseau 192.168.3.0/24,
- Autoriser les réponses DNS en provenance de la machine 192.168.2.30, au réseau 192.168.3.0/24,
- Autoriser les paquets entrants pour les connexions tcp établies,
- Supprimer le reste du traffic qui va apparaitre dans les **logs**.
- Application de la liste d'accès à l’interface fa1/1,
- Affichage de la configuration de l'interface
- Affichage de la liste de contrôle
- Suppression de l’acl
- Suppression de l'association de la liste de contrôle à fa1/1

ACL ‘Exercices’

- Créer access-list nommée reseau-telecom
  - **R2(config)#ip access-list extended reseau-telecom**
- Autoriser la machine 192.168.2.12 à se connecter via ssh à toutes les machines du réseau 192.168.3.0/24:
  - **R2(config-ext-nacl)#permit tcp host 192.168.2.12 gt 1023 192.168.3.0 0.0.0.255 eq 22**
- Autoriser les réponses DNS en provenance de la machine 192.168.2.30:
  - **R2(config-ext-nacl)#permit udp host 192.168.2.30 eq 53 192.168.3.0 0.0.0.255 gt 1023**
- Autoriser les paquets entrants pour les connexions tcp établies:
  - **R2(config-ext-nacl)#permit tcp any any established**
- Supprimer le reste du traffic qui va apparaitre dans les **logs:**
  - **R2(config-ext-nacl)#deny ip any any log**

ACL ‘Exercices’

- Application de la liste d'accès à une interface
  - **R2(config)#int fa1/1**
    R2(config-if)#ip access-group reseau-telecom out
- Affichage de la configuration de l'interface
  - **R2#sh run int fa1/1**
    Building configuration...

    Current configuration : 136 bytes
    !
    interface FastEthernet1/1
    ip address 192.168.3.2 255.255.255.0
    ip access-group reseau-telecom out
    duplex auto
    speed auto
    end

ACL ‘Exercices’

- Affichage de la liste de contrôle
- **R2#show access-lists reseau-telecom**
  Extended IP access list reseau-telecom
  10 permit tcp host 192.168.2.1 gt 1023 192.168.3.0 0.0.0.255 eq 22
  20 permit tcp any any established
  30 deny ip any any log

- Suppression d'une acl

  R2(config)#no ip access-list extended reseau-telecom

- Suppression de l'association de la liste de contrôle à fa1/1

  R2(config)#int fa1/1

  R2(config-if)#no ip access-group reseau-telecom out

Conseils

- Créer les ACL à l'aide d'un éditeur de texte et de faire un copier/coller dans la configuration du routeur,
- Placer les extended ACL 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
  - Rappel : les ACL standard ne regardent que l'IP source
- 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)

Les ACL Turbo

- ACL de base présentent des limitations
- Dans ACL standard / étendue:
  - Recherche séquentielle d'un match → temps de recherche augmente avec la taille de l'ACL
- Fonction Turbo
  - Compile les ACLs dans des tables de recherche « lookup tables
  - Temps de recherche fixe
  - 5 itérations quelque soit la taille de l'ACL

Les ACL Turbo :Configuration

- Sur une ACL standard ou étendue :

#access-list 101 deny tcp 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255 eq telnet
#access-list 101 permit tcp 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255 eq http
#access-list 101 deny tcp 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255 eq http
#access-list 101 deny icmp 192.168.1.0 0.0.0.255 200.200.200.0 0.0.0.255

- Ajouter la commande: **access-list compiled**
- **LIMITATIONS**
  - Nécessite de la mémoire : entre 2 et 4Mb de plus
  - Si l'ACL est très grande: plus de temps pour compiler
  - access-list compiled : n'a pas d'argument
- **Fonctionnement !!!?**

Les ACL réflexives

- Les ACL réflexives, aussi appelées filtre de session IP, elles sont basées sur les sessions TCP et UDP: Permet de filtrer le trafic en fonction des informations de session des couches sup.
- Lorsqu'un trafic autorisé passe par un routeur celui-ci va créer une instruction permit pour autoriser le trafic retour.
- Elle nécessite l'utilisation d'ACL étendues nommées et des mots-clés : **reflect** et **evaluate**.

- ACL étendues + option **established**, mais uniquement pour TCP
- ACL réflexives permettent de faire ce type de filtrage avec TCP, mais aussi UDP et ICMP
-

Les ACL réflexives Fonctionnement

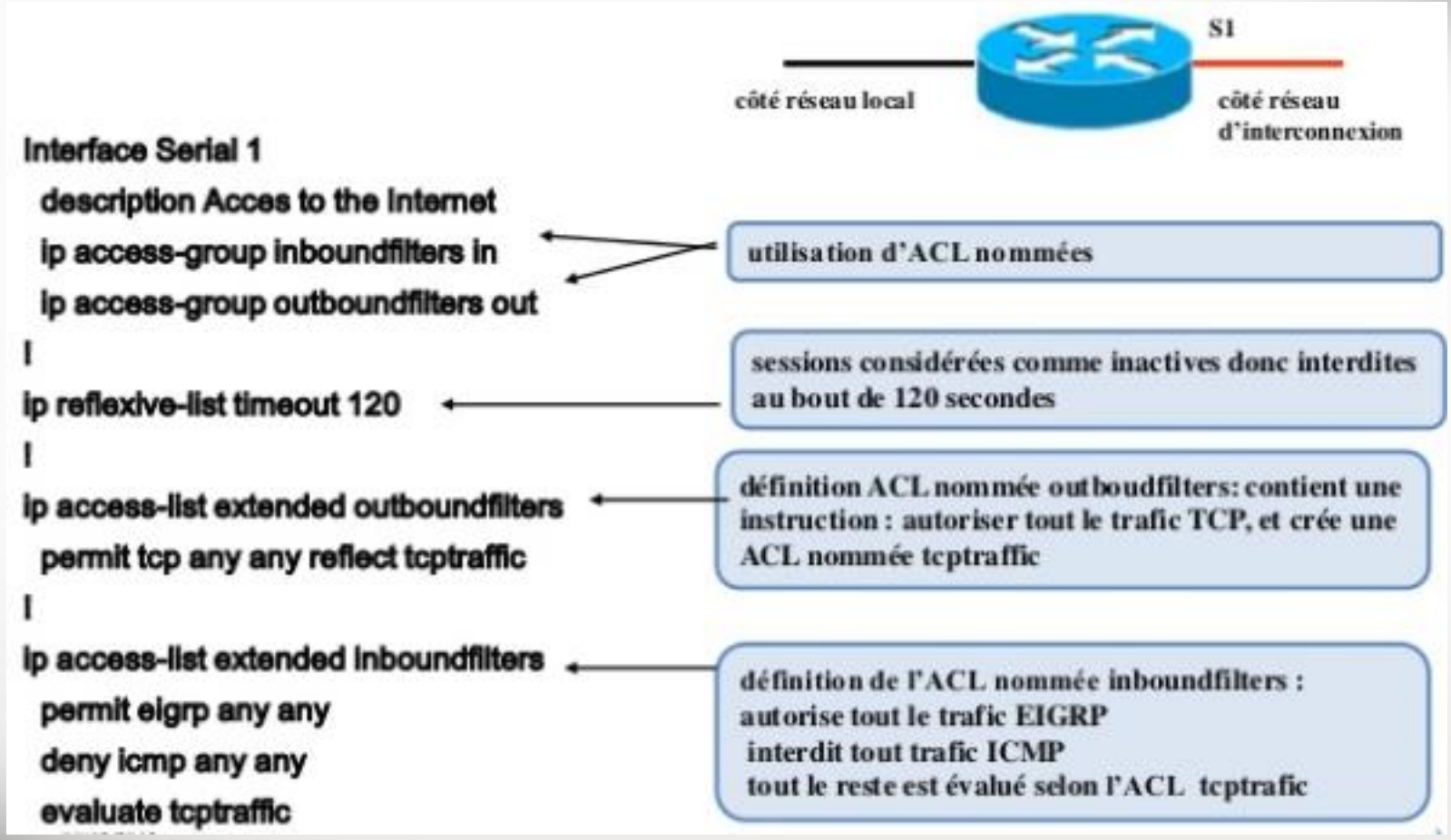
- Création dynamique d'une entrée temporaire :
  - entrée toujours: permit
  - Même Protocole que le paquet original
  - @IP source/dest inversées
  - N°port source/dest inversés

- Suppression de l'entrée temporaire une fois la session terminée

Les ACL réflexives Fonctionnement

- Session TCP:
  - Bit FIN = 1: session va se terminer,
  - Attente 5s afin que le hôte et le serveur terminent la session, puis blocage du trafic,
  - Bit RST=1: interruption brutale de session, Blocage immédiat du trafic ,
  - Par défaut : blocage de trafic après un temps d'inactivité de session ,
- Session UDP:
  - @IP source/dest ,
  - N° port source/dest ,
  - Fin de session: par défaut après un temps d'inactivité,

Les ACL réflexives: Configuration



- **faites #Show access-list** avant et après une connexion TCP

Les ACL réflexives: Limites

- Utilisée seulement avec ACL étendue nommée
- Ne peut être utilisée avec une application qui change de numéro de port !

Les ACL Contextuelles

- **CBAC**: Context Based Access Control
- fait partie de la fonctionnalité Pare-feu de l'IOS Cisco
- Plus performant que réflexive : tient compte des informations de la couche application.
- Supporte les protocoles utilisant plusieurs numéros de port,

Les ACL Contextuelles
Fonctionnement

- Paquets arrivant sur une interface **inspectés** par ACL de cette interface
- Seuls paquets qui passent ce barrage: inspectés par le CBAC
- Des tables d'état mises à jour grâce aux informations de session, pour chaque connexion active,
- CBAC interdit ou autorise uniquement le trafic TCP ou UDP spécifié
- Le filtrage se fait par l'ajout dynamique d'entées temporaires d'ACL

Les ACL Contextuelles Configuration

- **Étapes**:
  - Choisir l'interface
  - Configurer l'ACL sur cette interface
  - fixer les temporisations et les seuils
  - définir les règles d'inspection: spécifie quel trafic sera inspecté (application)
  - appliquer les règles d'inspection aux interfaces

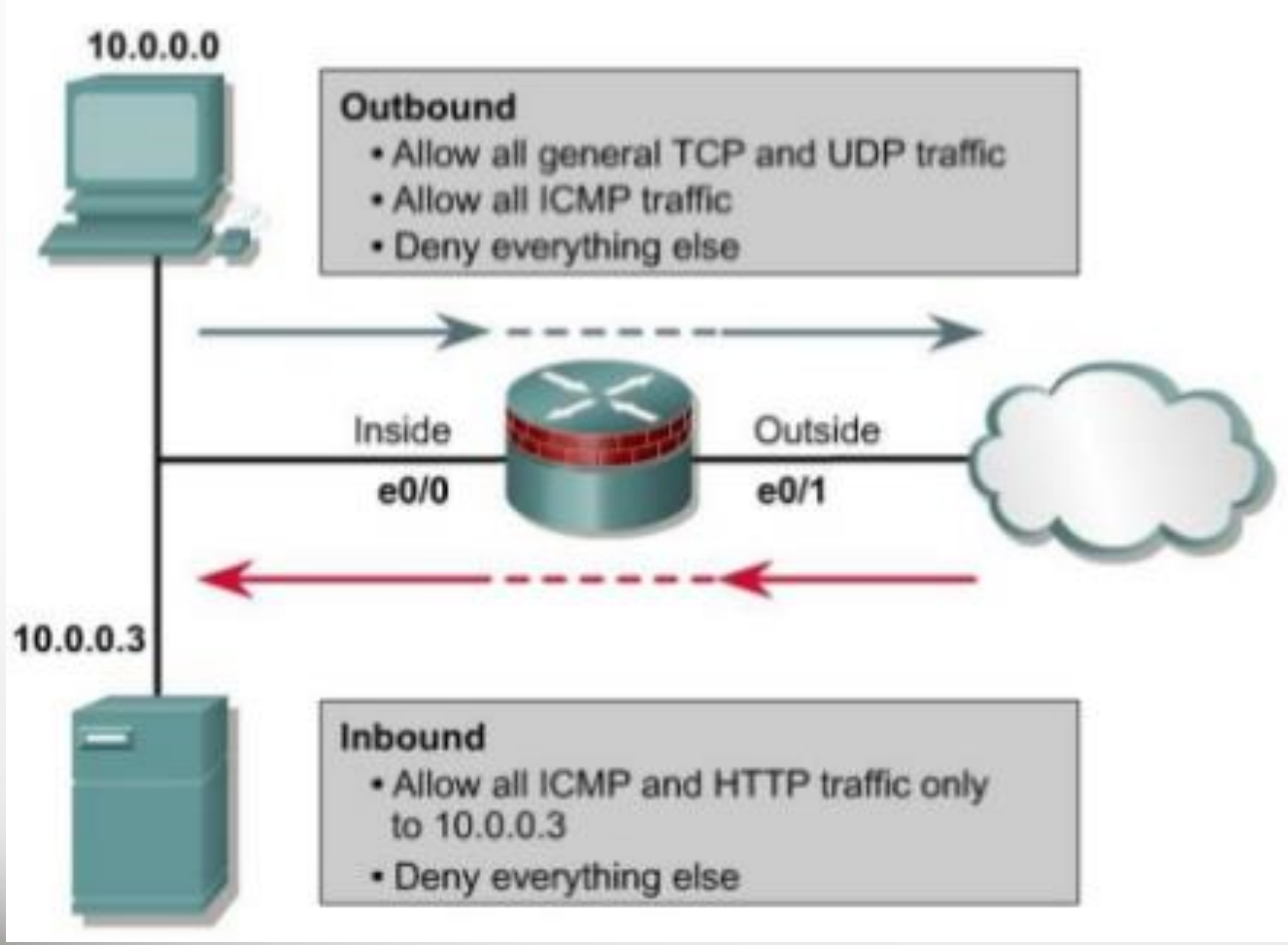
Les ACL Contextuelles
Configuration

- **Time out et seuils:**
  - Détermine le temps pendant lequel il gère les informations relatives aux sessions et pour déterminer quand une session se termine,
  - contrôle le nombre total de sessions ouvertes ainsi que celles nouvellement établies sur une certaine durée
  - Gère des compteurs de demi-sessions.
    - TCP = session na pas atteint l'Etat établi
    - UDP = routeur na pas détecté de trafic de retour

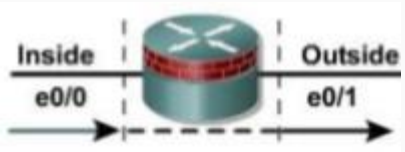
Les ACL Contextuelles Configuration

- **Les protocoles de niveau application supportés:**
  - FTP
  - TFTP
  - UNIX R-commands (rlogin, rexec, rsh, ...)
  - SMTP
  - HTTP Java
  - SQL*Net
  - RTSP (RealNetworks)
  - Autres multimédia :
    - Microsoft NetShow
    - StreamWorks
    - VDOLive

Les ACL Contextuelles Configuration



Les ACL Contextuelles Configuration




```
Router(config)# ip inspect name OUTBOUND tcp
Router(config)# ip inspect name OUTBOUND udp
```

- Configure CBAC pour l'inspection du trafic TCP et UDP

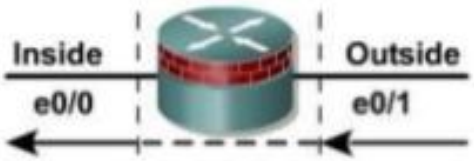
```
Router(config)# access-list 101 permit ip 10.0.0.0
  0.0.0.255 any
Router(config)# access-list 101 deny ip any any
```

- Autorise le trafic initié par les hôtes du réseau 10.0.0.0/24

```
Router(config)# interface e0/0
Router(config-if)# ip inspect OUTBOUND in
Router(config-if)# ip access-group 101 in
```

- Applique les règles d'inspection et l'ACL à l'interface e0/0 en entrée

Les ACL Contextuelles
Configuration




```
Router(config)# access-list 102 permit icmp any
  host 10.0.0.3
Router(config)# access-list 102 permit tcp any
  host 10.0.0.3 eq www
Router(config)# access-list 102 deny ip any any
```

- Autorise seulement le trafic ICMP et HTTP vers 10.0.0.3, initié depuis l'extérieur

```
Router(config)# interface e0/1
Router(config-if)# ip access-group 102 in
```

- Applique l'ACL à l'interface e0/1 en entrée

Les ACL Contextuelles
Limites

- Inspecte que le trafic spécifié: contrôle plus fin, mais beaucoup d'entrées « ip inspect » pour couvrir tous les types de connexions,
- demande une connaissance des protocoles et des applications utilisés
- trafic généré par le routeur lui-même n'est pas inspecté
- trafic envoyé au routeur lui-même n'est pas inspecté
- Seul le mode passif de FTP est compatible avec le CBAC

Les ACL Comparaisons

- ACL réflexives plus performantes que les ACL étendues : tiennent compte de l'information de session

- CBAC plus performant : tient compte en plus d'informations protocolaires de niveau application,
  - Permet ainsi de renforcer la sécurité d'un site

ACL basées sur l'heure,

- Utilisent des plages temporelles

ACL basées sur l'heure distribuée

Par Dr RIAHLA Mohamed Amine

Proxy d'authentification

- Utilisent des plages temporelles

Les ACL dynamiques

- Les ACL dynamiques obligent l'utilisateur à établir une connexion Telnet sur le routeur en fournissant à ce dernier une combinaison nom d'utilisateur/ mot de passe, pour autoriser le trafic de cette utilisateur.
- Si l'authentification Telnet a réussi, le routeur modifie dynamiquement l'ACL associée, autorisant le trafic provenant de l'IP de l'utilisateur.
-