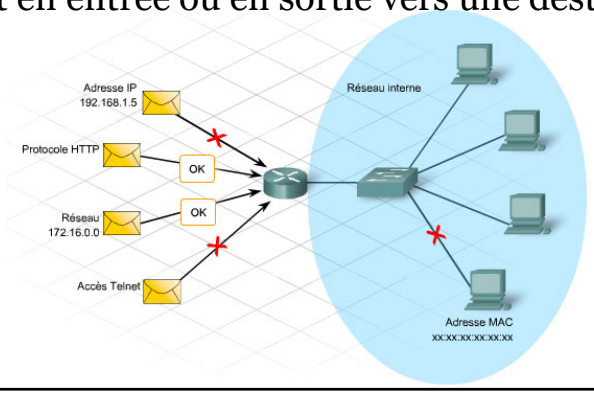
ACL

- En anglais ACL = Acces Control Lists
- En Français ACL = liste de contrôle d'accès
- ACL permettent d'établir des règles de filtrage sur les routeurs, pour régler le trafic des datagrammes en transit.
- Les ACL sont des instructions qui expriment une liste de règles, imposées par l'administrateur, donnant un contrôle sur les paquets reçus et transmis par le routeur.

ACL

- Les listes de contrôle d'accès sont capables d'autoriser ou d'interdire des paquets, que ce soit en entrée ou en sortie vers une destination.



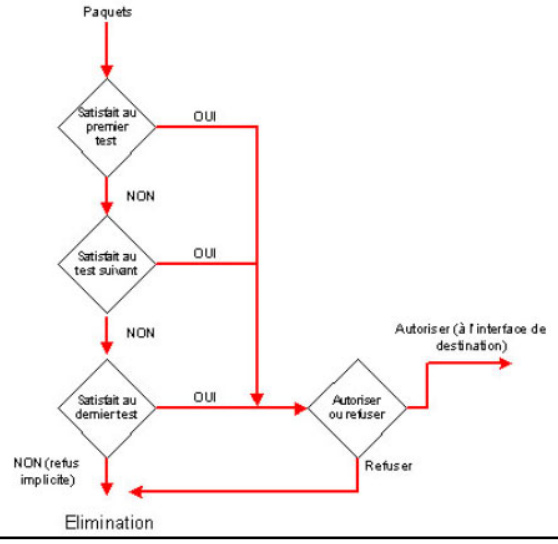
ACL

- Une liste de contrôle d'accès permet d'autoriser ou de refuser des paquets en fonction d'un certain nombre de critères, tels que :
  - Adresse d'origine
  - Adresse de destination
  - Numéro de port.
  - Protocoles de couches supérieures
  - Autres paramètres (horaires par exemple)

Vérification des paquets

- Les ACL opèrent selon un ordre séquentiel et logique, en évaluant les paquets à partir du début de la liste d'instructions.
- Si le paquet satisfait à une condition, il est autorisé ou refusé (suivant l'instruction) et les autres instructions ne sont pas vérifiées.
- Si un paquet ne correspond à aucune instruction dans l'ACL, le paquet est jeté. Ceci est le résultat de l'instruction implicite *deny any* à la fin de chaque ACL

Vérification des paquets



ACL

- Il existe deux familles d'ACL :
  - Standard : uniquement sur les IP sources
  - Etendue : sur quasiment tous les champs des en-têtes IP, TCP et UDP

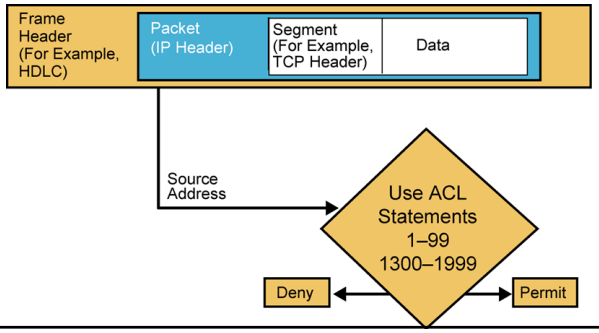
Numérotation des ACL

- Une liste de contrôle d'accès est identifiable par son numéro, attribué suivant le protocole et le type
- Type de liste / Plage de numéros

Type de liste	Plage de numéros
Listes d'accès IP standard	*1 à 99 et 1300 - 1999*
Listes d'accès IP étendues	*100 à 199 et 2000 - 2699*
Listes d'accès Appletalk	*600 à 699*
Listes d'accès IPX standard	*800 à 899*
Listes d'accès IPX étendues	*900 à 999*
Listes d'accès IPX SAP	*1000 à 1099*
…	

Listes de contrôle d'accès standard

- La liste de contrôle d'accès standard constitue le type le plus simple.
  - le filtre est basé sur l'adresse IP source d'un paquet.



Listes de contrôle d'accès standard

- Pour les listes d'accès standard autorisant ou refusant le trafic IP, le numéro d'identification est compris
  - entre <1 et 99>

  et
  - entre <1 300 et 1 999>

Listes de contrôle d'accès standard

- Une liste d'accès standard se crée par la commande suivante :
  access-list num_acl {permit | deny} source {masque_source}
  - *Numéro_de_liste_d'accès :* identifie la liste
  - *Permit | deny* : autoriser ou interdire
  - *Source* : identifie l'adresse IP source
  - *Masque_source* : bits de masque générique
- Exemple :
  - access-list 1 deny 212.217.0.0 0.0.255.255

Principe du masque générique

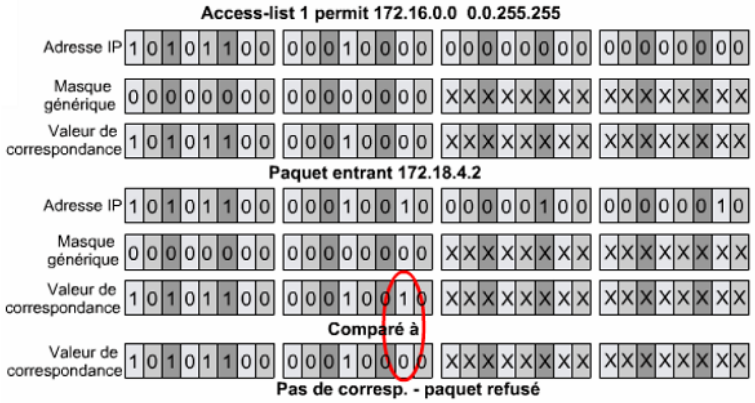
- Les listes de contrôle d'accès utilisent le masquage générique pour identifier une adresse unique ou plusieurs adresses dans le but d'effectuer des vérifications visant à accorder ou interdire l'accès.
- Un masque générique (Wildcard mask) est une suite de 32 bits divisés en quatre octets contenant chacun 8 bits.
  - 0 signifie " vérifier la valeur du bit correspondant "
  - 1 signifie " ne pas vérifier (ignorer) la valeur du bit correspondant "

Principe du masque générique

- Le terme masque générique est un surnom du procédé de correspondance masque-bit des listes de contrôle d'accès.
- Exemples:
  - 0.0.0.0 : tous les bits doivent être examinés
  - 0.0.0.255 : seuls les 3 premiers octets sont examinés
  - 255.255.255.255 : l'adresse n'a pas besoin d'être examinée : désigne tout le monde
  - 0.0.15.255 : les 20 premiers bits sont examinés

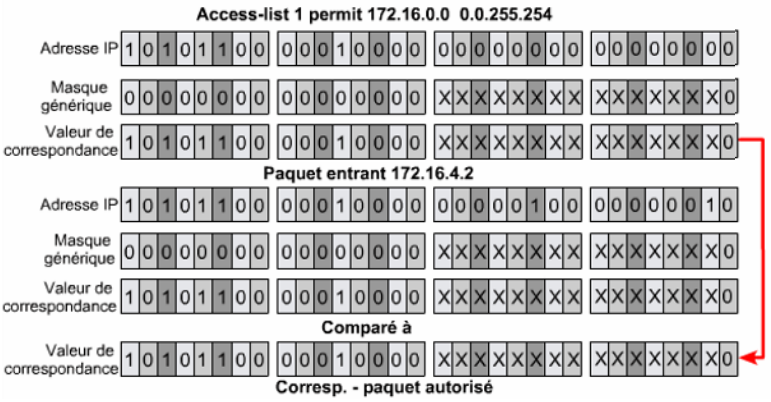
Principe du masque générique

- Exemple 1:



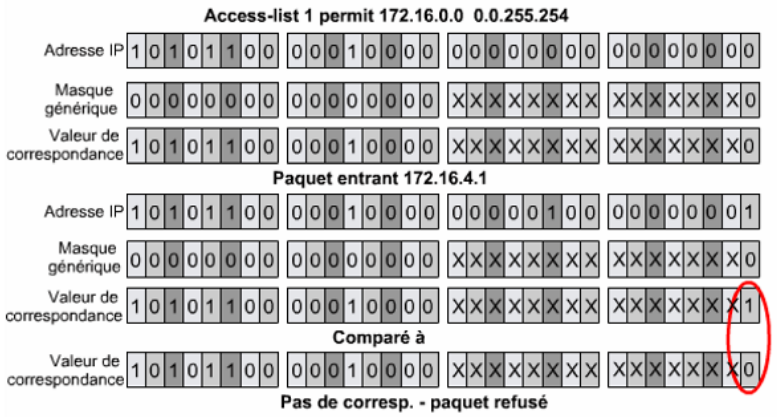
Principe du masque générique

- Exemple 2:



Principe du masque générique

- Exemple 3:



Principe du masque générique

- Commande *host* et *any* : Ces deux commandes sont des abréviations permettant de simplifier la lecture ainsi que l'écriture des listes de contrôle d'accès :
  - any : n'importe quelle adresse (équivaut à 0.0.0.0 255.255.255.255)
  - host : désigne une machine (équivaut au masque 0.0.0.0)
- Exemple :
  - host 212.217.40.5 équivaut à 212.217.40.5 0.0.0.0

Affectation d'une liste de contrôle d'accès à une interface

- Une fois la liste de contrôle d'accès crée, il faut l'assigner à une interface de la manière suivante :

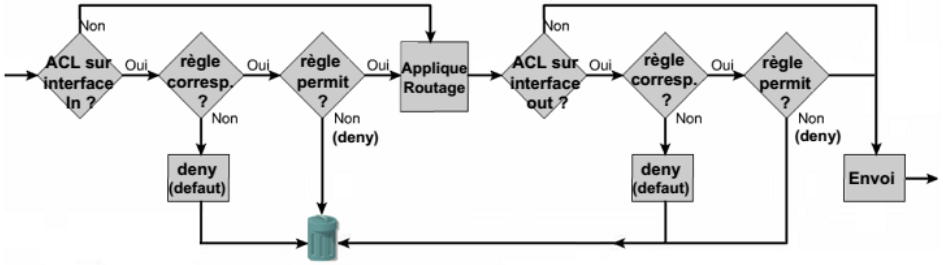
  *Routeur(config-if)#**ip access-group** numéro_liste_d'accès {in | out }*

  - *In | out* : indique si la liste doit être appliquée pour le trafic entrant ou sortant
- Exemple :

  R1(config)#interface fastethernet 0/0
  R1(config-if)#ip access-group 5 in

Affectation d'une liste de contrôle d'accès à une interface

- In / Out



Désactivation d'une ACL

- Pour supprimer une liste de contrôle d'accès d'une interface, on utilise la commande

no ip access-group <numéro de liste d'accès> {in | out }

Commenter une ACL

- Pour commenter une ACL on utilise remark
  - Un commentaire d'ACL sera visible dans la configuration, mais il sera aussi affiché par la commande *show access-lists*
  - Une remarque d'ACL est donc différente d'un commentaire introduit par *!*
- Exemple*:*
  - R1(config)# access-list 109 **remark** Autoriser le traffic vers le serveur
  - R1(config)# access-list 109 permit ip any host 212.217.8.7

Vérification des ACL

- Pour vérifier les listes de contrôle d'accès, on utilise les commandes :
  - show ip interface : affiche les informations relatives à l'interface IP et indique si des listes de contrôle d'accès sont configurées.
  - show access-lists : affiche le contenu de toutes les listes de contrôle d'accès.
    - La saisie du nom ou du numéro d'une liste de contrôle d'accès en tant qu'option de cette commande vous permet de consulter une liste spécifique

Vérification des ACL

- Exemple :
  - show access-lists [number | name]
    - toutes les règles de l'ACL quelque soit l'interface
  - show ip access-lists [number | name]
    - les règles de l'ACL liées au protocole IP

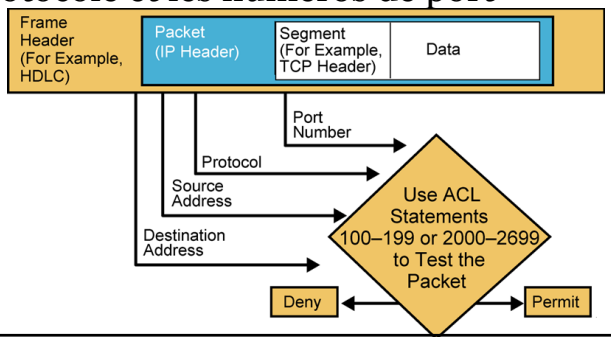
Vérification des ACL

- Exemple :

```
Routeur 1#show access-lists
Standard IP access list 1
Deny 204.59.144.0, wildcard bits 0.0.0.255
Permit any
Routeur 1#
```

Listes de contrôle d'accès étendues

- Les listes de contrôle d'accès étendues filtrent non seulement sur l'adresse IP source, mais également sur l'adresse IP de destination, le protocole et les numéros de port



Listes de contrôle d'accès étendues

- Pour les listes d'accès étendues autorisant ou refusant le trafic IP, le numéro d'identification est compris
  - entre <100 et 199>

  et
  - entre <2000 et 2699>

Listes de contrôle d'accès étendues

- Une liste de contrôle d'accès étendue se crée par la commande suivante :

  access-list numéro_de_liste_d'accès {permit | deny} protocole source {masque_source} destination {masque_destination} {opérateur opérande} [established] [log]

  - Numéro_de_liste_d'accès : identifie la liste
  - Permit | deny : autoriser ou interdire
  - Protocol : indique le type de protocole IP, TCP, UDP, ICMP, …
  - Source et destination : identifient l'adresse IP source et destination
  - Masque_source et masque_destination : bits de masque générique

Listes de contrôle d'accès étendues

- opérateur : opérateurs de comparaison sur les ports
  - Lt : (less than) plus petit
  - Gt : (greater than) plus grand
  - Eq : (equal) égal
  - neq : (not equal) non égal
  - …
- opérande : n° de port
  - Les numéros de ports peuvent être exprimé de manière numérique ou bien par une équivalence alphanumérique
- established : autorise le trafic TCP si les paquets utilisent une connexion établie (bit de ACK)
- log : active le processus de la journalisation

Listes de contrôle d'accès étendues

- Ex 1 : 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
  - On doit préciser le protocole (IP)
- Ex 2 : access-list 101 deny tcp any gt 1023 host 10.1.1.1 eq 23
  - Refus de paquet TCP d'un port source > 1023 et à destination du port 23 de la machine d'IP 10.1.1.1
- Ex 3 : 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
  - Utilisation de mots clés pour les ports « well-known » http ou www (80) , ftp (21), pop (110), smtp (25)

Listes de contrôle d'accès nommées

- Les listes de contrôle d'accès nommées (NACL) permettent d'identifier les listes de contrôle d'accès IP standards et étendues par des chaînes alphanumériques plutôt que par la représentation numérique actuelle.

Listes de contrôle d'accès nommées

- Les ACLs nommées sont utiles dans les situations suivantes :
  - Identifier plus intuitivement une ACL
  - Configurer plusieurs ACL standard et plusieurs ACL étendues dans un routeur pour un protocole donné

Listes de contrôle d'accès nommées

- Définition d'une ACL nommée :
  - Pour configurer les listes de contrôle d'accès nommées, la syntaxe est la suivante :

ip access-list {standard | extended} nom

Listes de contrôle d'accès nommées

- Exemple :

  Router(config)# ip access-list extended bloque
  Router(config-ext-nacl)# deny tcp host 10.1.1.2 eq www any
  Router(config-ext-nacl)# deny ip 10.1.1.0 0.0.0.255 any
  Router(config-ext-nacl)# permit ip any any
- Pour supprimer une des règles, il suffit de faire
  - *ip access-list {standard | extended} name*
  - *no règle-à-supprimer*
    - Exemple :
      - Router(config)# ip access-list extended bloque
      - Router(config-ext-nacl)# no permit ip any any

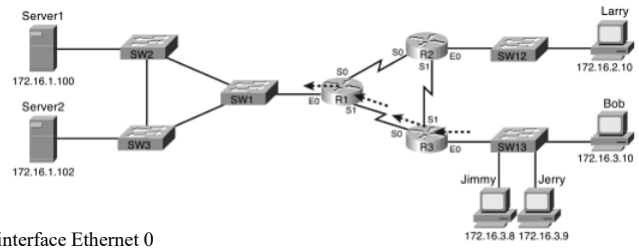
Accès au Telnet avec une ACL

- Pour restreindre les connexions entrantes ou sortantes entre un vty particulier et les adresses d'une ACL, on utilise la commande
  - access-class number { in | out }
- Exemple : utiliser une ACL dans le but de contrôler l'accès au Telnet (donc au vty)

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

Emplacement des ACL standard

- Refuser à Bob l'accès à tous les serveurs ftp de sous réseau 172.16.1.0/24. le reste est autorisé

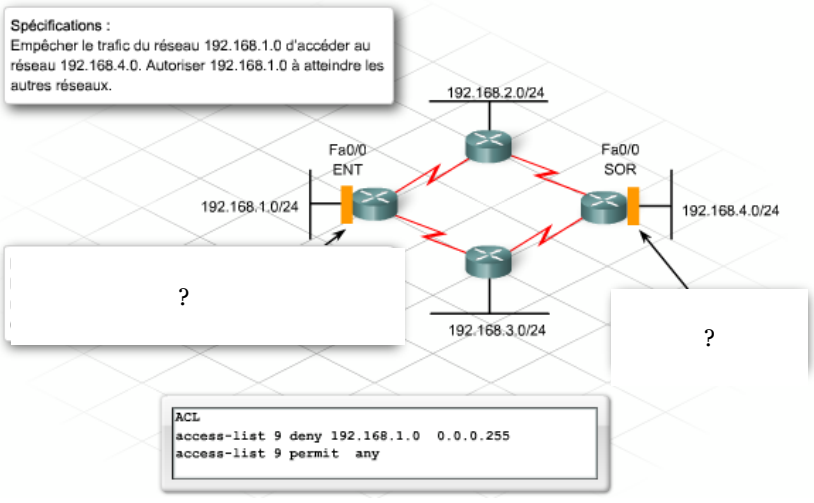


Routeur R1

R1(config)#interface Ethernet 0
Ip address 172.16.1.1 255.255.255.0
ip access-group 1 out

access-list 1 deny host 172.16.3.10
access-list 1 permit any

Emplacement des ACL standard

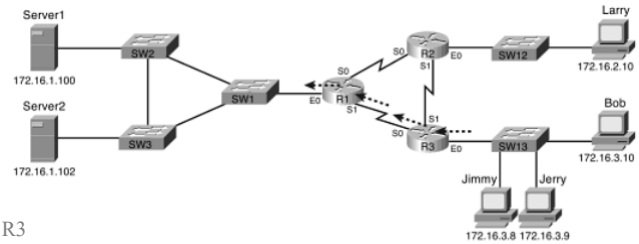


Emplacement des ACL standard

- Important : Une liste d'accès standard se place sur le trafic sortant de l'interface de routeur la plus proche de la destination

Emplacement des ACL étendues

- Ex : Refuser à Bob l'accès à tous les serveurs ftp de sous réseau 172.16.1.0/24. le reste est autorisé

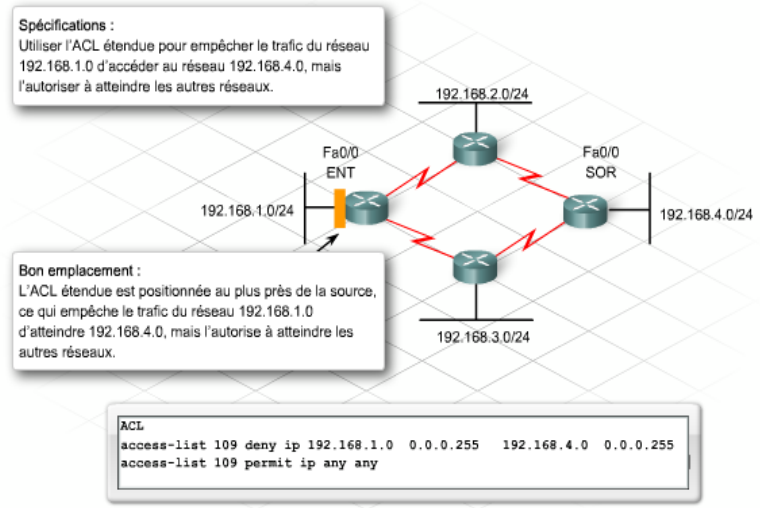


Routeur R3

```
interface Ethernet0
 ip address 172.16.3.1 255.255.255.0
 ip access-group 103 in

access-list 103 remark deny Bob to FTP servers in subnet 172.16.1.0/24
access-list 103 deny tcp host 172.16.3.10 172.16.1.0 0.0.0.255 eq ftp
access-list 103 permit ip any any
```

Emplacement des ACL étendues



Emplacement des ACL étendues

- Important : Une liste d'accès étendue devrait se place sur le flux entrant de l'interface la plus proche de la source du trafic à contrôler.

ACL avancée

ACL avancée

- Il existe d'autres type d'ACL
  - ACL à caractère temporel
  - ACL dynamique
  - ACL réflexive
  - ACL basée sur le contexte
  - ACL ZBF

ACL à caractère temporel

- Ce genre d'ACL permet d'interdire certains trafics pendant certains périodes
- La référence de temps utilisée est l'horloge interne du routeur, il est dans ce cas intéressant d'utiliser le protocole NTP (Network Time Protocol) pour bien synchroniser tous les équipements

ACL à caractère temporel

- Exemple :

```
access-list 101 permit tcp 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255 eq
telnet time-range EVERYOTHERDAY

time range EVERYOTHERDAY

periodic Monday Wednesday Friday 8:00 to 17:00

interface Ethernet0/0

ip address 10.1.1.1 255.255.255.0

ip access-group 101 in
```

les connexions Telnet sont autorisées les lundis, mercredis et vendredis de 8h à 17h

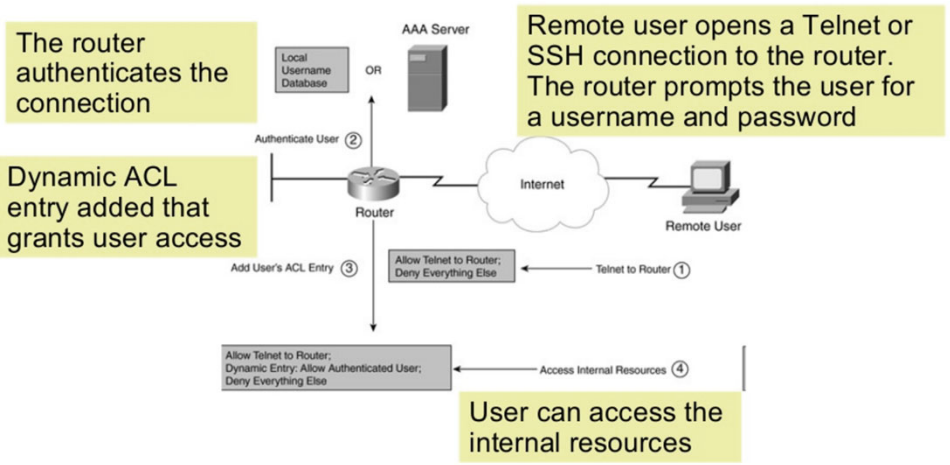
ACL dynamique

- Les ACL classiques utilisent l'adresse IP pour déterminer quelle machine communique
  - mais il n'y a pas de vérification de l'identité de l'utilisateur lui-même
- Il est souvent utile de demander à l'utilisateur de s'identifier :
  - nom d'utilisateur
  - mot de passe
- On peut alors utiliser une ACL dynamique

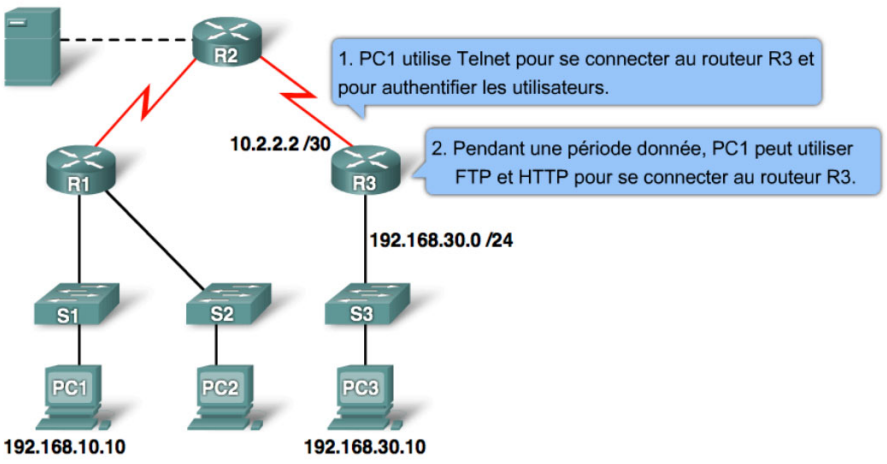
ACL dynamique

- les ACL dynamiques (lock and Key) se mettent en place après authentification de l'utilisateur
  - Les utilisateurs souhaitant traverser les routeur sont bloqués tant qu'ils n'utilisent pas Telnet ou SSH pour se connecter au routeur et tant qu'ils n'ont pas été authentifiés.

ACL dynamique



ACL dynamique : Exemple



ACL dynamique : Exemple

Étape 1	`R3(config)#username Student password 0 cisco`

Étape 2	`R3(config)# access-list 101 permit any host 10.2.2.2 eq telnet`
	`R3(config)#access-list 101 dynamic testlist timeout 15`
	`permit ip 192.168.10.0 0.0.0.255 192.168.30.0 0.0.0.255`

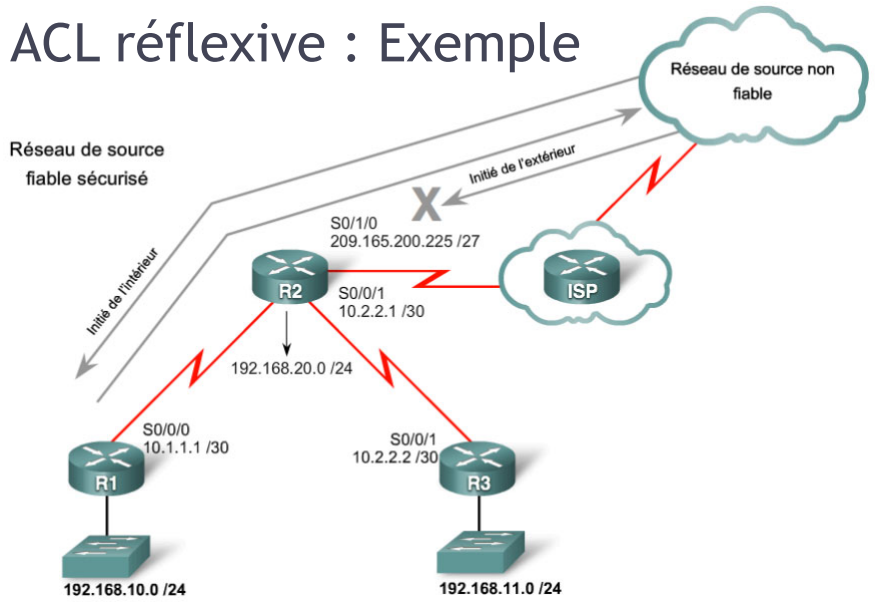
Étape 3	`R3(config)#interface serial 0/0/1` `R3(config-if)#ip access-group 101 in`

Étape 4	`R3(config)#line vty 0 4`
	`R3(config-line)#login local`
	`R3(config-line)# autocommand access-enable host timeout 5`

ACL réflexive

- ACLs réflexives autorisent le trafic sortant et limitent le trafic entrant en réponse aux sessions provenant du routeur lui-même.
  - ACLs réflexives permettant de faire ce type de filtrage avec TCP, mais aussi UDP et ICMP.

ACL réflexive : Exemple



ACL réflexive : Exemple

Étape 1	`R2(config)#ip access-list extended OUTBOUNDFILTERS`
	`R2(config-ext-nacl)# permit tcp 192.168.0.0 0.0.255.255 any reflect TCPTRAFFIC`
	`R2(config-ext-nacl)# permit icmp 192.168.0.0 0.0.255.255 any reflect ICMPTRAFFIC`

Étape 2	`R2(config)#ip access-list extended INBOUNDFILTERS`
	`R2(config-ext-nacl)# evaluate TCPTRAFFIC`
	`R2(config-ext-nacl)# evaluate ICMPTRAFFIC`

Étape 3	`R2(config)#interface S0/1/0`
	`R2(config-if)#ip access-group INBOUNDFILTERS in`
	`R2(config-if)#ip access-group OUTBOUNDFILTERS out`

ACL réflexive

- Les limitations des ACL réflexives sont :
  - elles ne fonctionnent pas avec les applications qui négocient les ports
  - elles ne permettent pas de limiter le nombre de sessions autorisées simultanément
  - elles ne tiennent pas compte des informations du protocole de niveau application

ACL CBAC

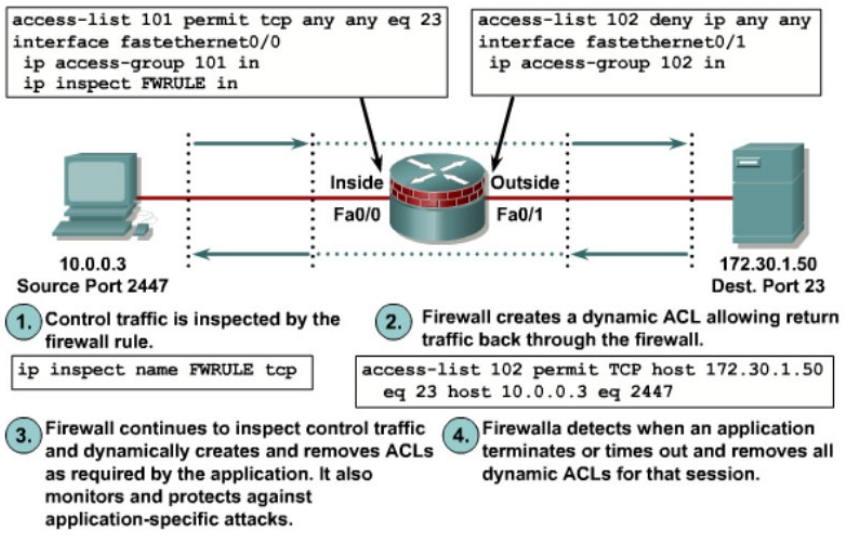
- ACL basée sur le contexte
  - Appelée aussi ACL CBAC (Context Based Acces Control)
- Associée à une ACL étendue, cela permet de tracer les sessions (tcp, udp, telnet…) qui demanderont un retour et de leur ouvrir l'accès.
- Utile pour configurer un pare-feu ou tout peut sortir mais rien rentrer.

ACL CBAC : Configuration

- Sur les interfaces depuis lesquelles le trafic sera initié
  - appliquer l'ACL dans le sens entrant qui autorise uniquement le trafic désiré
  - appliquer les règles d'inspection dans le sens entrant, ces règles s'appliquant au trafic autorisé
- Sur les autres interfaces
  - appliquer l'ACL dans le sens entrant qui interdit tout le trafic non désiré

➔Le firewall appliquera le filtrage dynamique au sens du CBAC

ACL CBAC : Configuration



ACL CBAC

- Les limitations des ACL CBAC sont :
  - Le CBAC n'inspecte que le trafic explicitement spécifié.
    - C'est un avantage car le contrôle peut se faire finement, mais il faut souvent beaucoup d'entrées « ip inspect » pour couvrir tous les types de connexions
  - Le CBAC n'est pas très simple d'utilisation et demande une bonne connaissance des protocoles et des applications utilisés
  - CBAC ne peut inspecter les données cryptées (IPSec). Il peut cependant inspecter les canaux VPN dont il est à l'origine
  - Seul le mode passif de FTP est compatible avec le CBAC

ACL Zone-Based Firewall

- Les ACLs « Zone-Based Firewall » (ZBF, ZPF ou ZFW) se basent sur le principe de définition de zones et de création de règles d'une zone à une autre
- Avec ZBF, les interfaces sont assignées à une des zones sur lesquelles une règle d'inspection du trafic (inspection policy) est appliquée. Elle vérifie le trafic qui transite entre les zones.
- Une règle par défaut bloque tout trafic tant qu'une règle explicite ne contredit pas ce comportement.

ACL ZBF : Principe 1/2

- Une zone doit être configurée (créée) avant qu'une interface puisse en faire partie.
- Une interface ne peut être assignée qu'à une seule zone.
- Tout le trafic vers ou venant d'une interface donnée est bloqué quand elle est assignée à une zone sauf pour le trafic entre interfaces d'une même zone et pour le trafic du routeur lui-même (Self zone).
- Une politique de sécurité (zone-pair) peut contrôler le trafic entre deux zones en faisant référence à un en ensemble de règles (policy-map).

ACL ZBF : Principe 2/2

- Un policy-map prend des actions et fait référence à des critères de filtrage (class-maps).
- Quand du trafic passe d'une zone à une autre (zone-pair), un policy-map est appliqué.
- Pour chaque class-map (critère de filtrage) du policy-map, une action est prise : pass, inspect ou drop de manière séquentielle.

Trois actions sur les class-maps

- Inspect
  - Met en place un pare-feu à état (équivalent à la commande ip inspect) .
  - Capable de suivre les protocoles comme ICMP ou FTP (avec de multiples connexion data et session)
- Pass
  - Équivalent à l'action permit d'un ACL.
  - Ne suit pas l'état des connexions ou des sessions.
  - Nécessite une règle correspondante pour du trafic de retour.
- Drop
  - Équivalent à l'action deny d'un ACL.
  - Une option log est possible pour journaliser les paquets rejetés.

Règles de filtrage : policy-maps

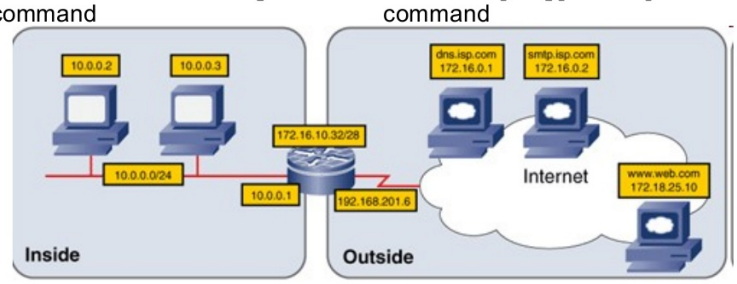
- Les actions Inspect, Pass et Drop ne peuvent être appliquées qu'entre des interfaces appartenant à des zones distinctes.
- La Self zone, la zone du routeur/pare feu comme source ou destination est une exception à ce refus implicite de tout.
  - Tout le trafic vers n'importe quelle interface du routeur est autorisé jusqu'au moment où il est implicitement refusé.
- Les interfaces qui ne participent pas à ZBF fonctionnent comme des ports classiques et peuvent utiliser une configuration SPI/CBAC.

Cisco Policy Language (CPL)

- Règles:
  - Définir des class-maps (critères de filtrage) qui décrivent le trafic que la politique de sécurité va vérifier à travers un policy-map.
  - Définir les policy-maps qui définissent les politiques de sécurités : le trafic filtré et l'action à prendre : drop, pass, inspect
- Zones :
  - Définir les zones (zone security)
  - Assigner les interfaces aux zones (zone-member security)
  - Définir les zone-pairs (zone-pair security)
- Application :
  - Appliquer les policy-maps aux zone-pairs (service-policy)

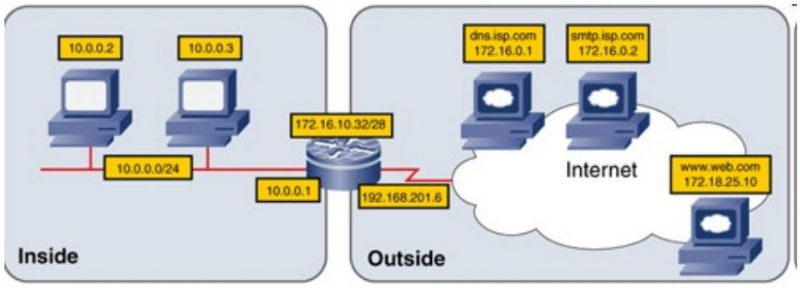
ACL ZBF : Exemple

1. Create the zones for the firewall with the `zone security` command
2. Define traffic classes with the `class-map type inspect` command



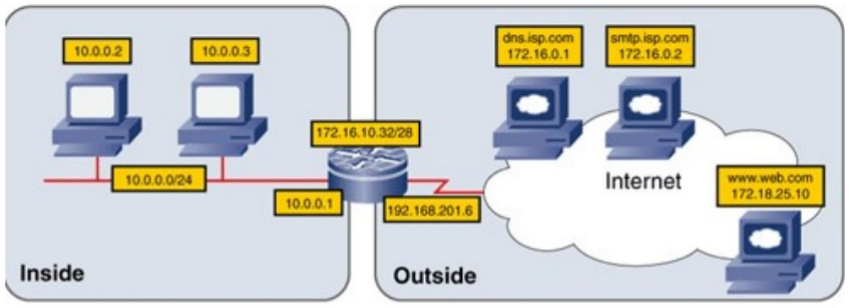
3. Specify firewall policies with the `policy-map type inspect` command
4. Apply firewall policies to pairs of source and destination zones with `zone-pair security`
5. Assign router interfaces to zones using the `zone-member security` interface command

Etape 1 : Définir les zones



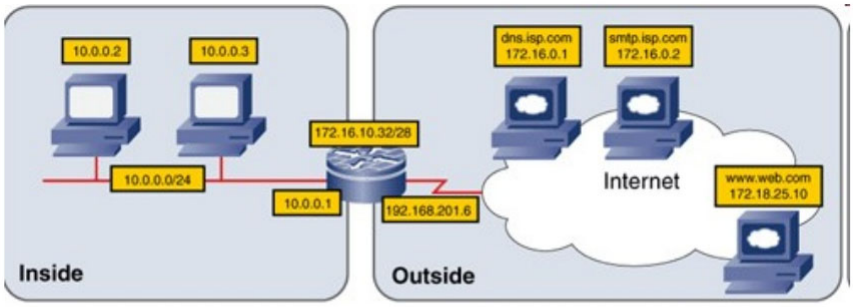

```
FW(config)# zone security Inside
FW(config-sec-zone)# description Inside network
FW(config)# zone security Outside
FW(config-sec-zone)# description Outside network
```

Etape 2 : Définir les class-maps



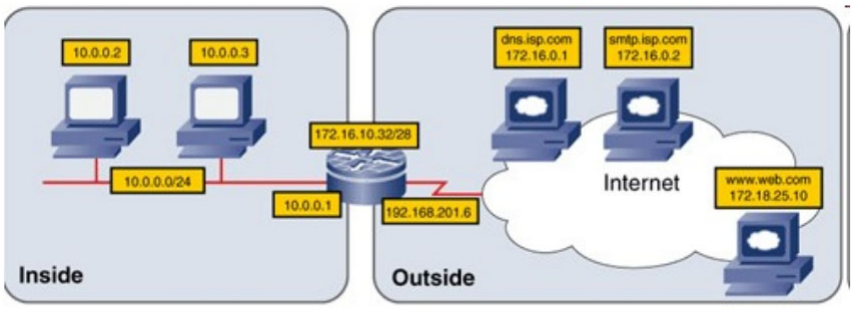

```
FW(config)# class-map type inspect FOREXAMPLE
FW(config-cmap)# match access-group 101
FW(config-cmap)# match protocol tcp
FW(config-cmap)# match protocol udp
FW(config-cmap)# match protocol icmp
FW(config-cmap)# exit
FW(config)# access-list 101 permit ip 10.0.0.0
0.0.0.255 any
```

Etape 3 : Définir les policy-maps



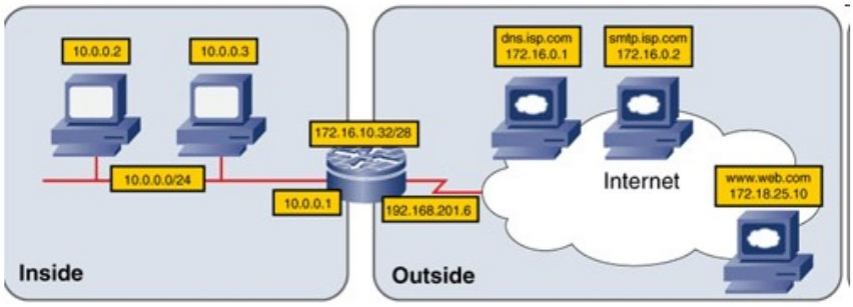

```
FW(config)# policy-map type inspect InsideToOutside
FW(config-pmap)# class type inspect FOREXAMPLE
FW(config-pmap-c)# inspect
```

Etape 4 : Appliquer les policy-maps aux zone-pairs




```
FW(config)# zone-pair security InsideToOutside source Inside
destination Outside
FW(config-sec-zone-pair)# description Internet Access
FW(config-sec-zone-pair)# service-policy type inspect
InsideToOutside
```

Etape 5: Assigner les interfaces aux zones




```
FW(config-sec-zone-pair)# interface F0/0
FW(config-if)# zone-member security Inside
FW(config-if)# interface S0/0/0
FW(config-if)# zone-member security Outside
```