CCNA SECURITY

Implémentation de pare-feux

LES ACL BASIQUES

Définition des ACL

- **ACL** signifie Access Control List.

- Sur un routeur, un PIX ou un ASA : liste d'adresses, de ports ou de trafics **autorisés** ou **interdits**.

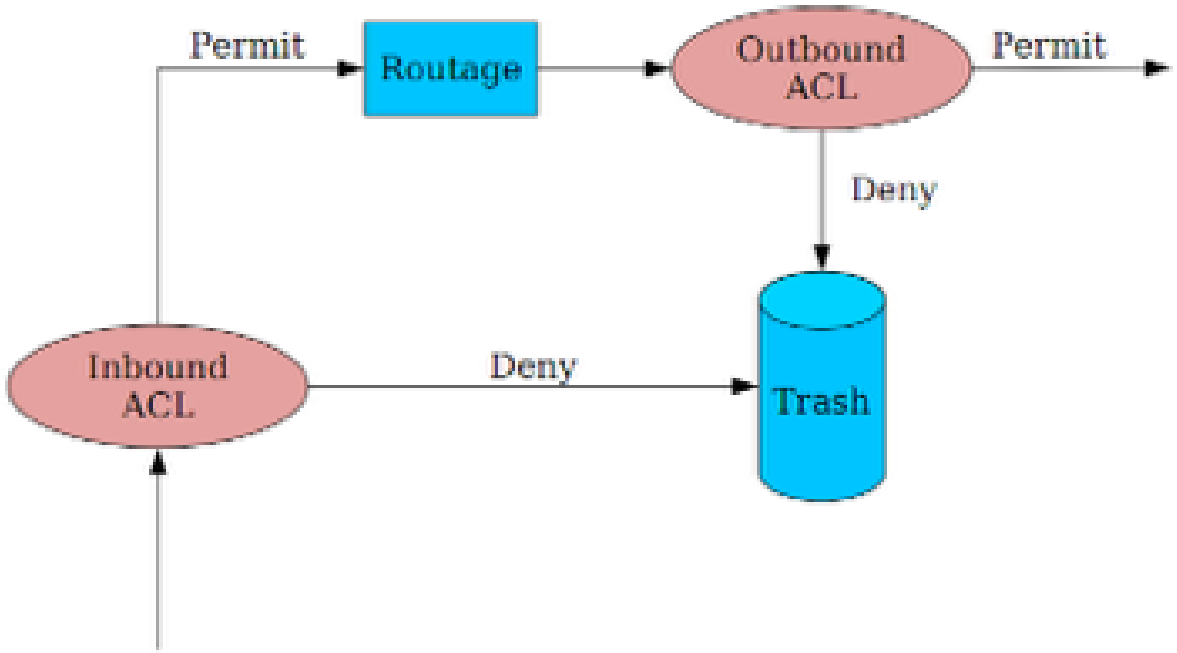
- Trois catégories d'ACL :
  - ACL standard : ne peut contrôler que l'adresse IP source, ou une partie de celle-ci via l'utilisation de masque générique.

  - ACL étendue : peut contrôler quasiment tous les champs présents dans les en-têtes IP, TCP et UDP (les adresses IP source et destination (ou une partie de celles-ci), le protocole, les ports source et destination, les priorités, ...).

  - ACL nommée-étendue : ACL étendue à laquelle on donne un nom.

Schéma du fonctionnement des ACLs

- Une ACL peut être appliquée sur du trafic entrant ou sortant :



Logique de fonctionnement des ACLs

- Vérification du paquet par rapport au premier critère défini.
- S'il vérifie le critère, application de l'action définie.
- S'il ne vérifie pas le critère, vérification du paquet par les ACLs suivantes.
- S'il ne vérifie aucun critère, le paquet est supprimé (**action "deny" par défaut**).

Les masques inversés

- Spécifie la partie de l'adresse IP devant être examinée.
- Exemple : Le masque inversé « 0.0.255.255 » veut dire que seuls les 16 premiers bits (deux premiers octets) de l'IP à laquelle il est associé doivent être examinés.
- Exemple :
  - « deny 192.168.1.0 0.0.0.255 » signifie que le système doit refuser toutes les adresses IP commençant par 192.168.1

Les ACLs standards

- **access-list** [numéro] {permit | deny} [ip source] [masque inversé]
  - Numéro : de 1 à 99 ou de 1300 à 1999.

- Associer une description à une ACL :
  - **access-list** [numéro] **remark** [description]

- Activation d'une ACL sur une interface :
  - **ip access-group** [numéro | nom] {in | out}

- Commandes de visualisation :
  - **show access-lists** [numéro | nom]

Exemples d'ACLs standards

- **`access-list 1 remark Stoppe tout le trafic venant du réseau 192.168.1.0/24`**
  - Description de l'access-list
- **`access-list 1 deny 192.168.1.0 0.0.0.255`**
  - Refuse tous les paquets provenant du réseau 192.168.1.0/24
- **`access-list 1 permit 0.0.0.0 255.255.255.255`**
  - Autorise tous les autres paquets
- **`interface Fa0/0`**
- **`ip access-group 1 out`**
  - Appliquer l'ACL sur le trafic sortant de l'interface fa0/0

Exemples d'ACLs standards (2)

- **`access-list 2 remark Stoppe tout le trafic venant de l'ordinateur 172.16.1.1`**
  - Description de l'access-list
- **`access-list 2 deny 172.16.1.1 0.0.0.0`**
  - Refuse tous les paquets provenant de l'IP 172.16.1.1
- **`access-list 2 permit 0.0.0.0 255.255.255.255`**
  - Autorise tous les autres paquets
- **`interface Fa0/0`**
- **`ip access-group 2 out`**
  - Appliquer l'ACL sur le trafic sortant de l'interface fa0/0

Exemples d'ACLs standards (2)

- **`access-list 3 remark Stoppe tout le trafic venant de l'ordinateur 172.16.1.1`**
  - Description de l'access-list
- **`access-list 3 deny host 172.16.1.1`**
  - Refuse tous les paquets provenant de l'IP 172.16.1.1
- **`access-list 3 permit 0.0.0.0 255.255.255.255`**
  - Autorise tous les autres paquets
- **`interface Fa0/0`**
- **`ip access-group 3 out`**
  - Appliquer l'ACL sur le trafic sortant de l'interface fa0/0

Exemples d'ACLs standards (3)

- `access-list 3 remark Stoppe tout le trafic venant de l'ordinateur 172.16.1.1`
- `access-list 3 deny host 172.16.1.1`
- `access-list 3 permit any`

- `access-list 1 remark N'autorise que les paquets du réseau 172.16.3.0/24`
- `access-list 1 permit 172.16.3.0 0.0.0.255`

- `interface Fa0/0`
- `ip access-group 3 out`
- `ip access-group 1 in`

Les ACLs étendues

- **access-list** [numéro] {permit | deny} [protocole] [ip source] [masque inversé] [ip destination] [masque inversé]
  - Numéro : de 100 à 199 ou de 2000 à 2699.

- Exemples :
  - Refuse les paquets IP provenant de n'importe quelle source et à destination du client 10.1.1.1 :
    - **access-list 101 deny ip any host 10.1.1.1**

  - Refuse les paquets TCP provenant de n'importe quel port supérieur à 1023 et à destination du port 23 du client 10.1.1.1 :
    - **access-list 101 deny tcp any gt 1023 host 10.1.1.1 eq 23**

Les ACLs nommées

- Une ACL peut être composée de plusieurs règles ; pour en modifier une partie, par défaut, la seule méthode consiste à supprimer entièrement l'ACL pour la refaire (commande : "**no access-list** [numéro]").

- Avec une ACL nommée-étendue, possibilité de ne supprimer qu'une seule règle.

Les ACLs nommées

- Exemple d'ACL nommée :
  - `Router(config)# ip access-list extended SafACL.`
  - `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`

- Exemple de suppression d'une règle dans cette ACL :
  - `Router(config)# ip access-list extended SafACL`
  - `Router(config-ext-nacl)# no deny ip 10.1.1.0 0.0.0.255 any`

Activation d'une ACL sur une ligne vty

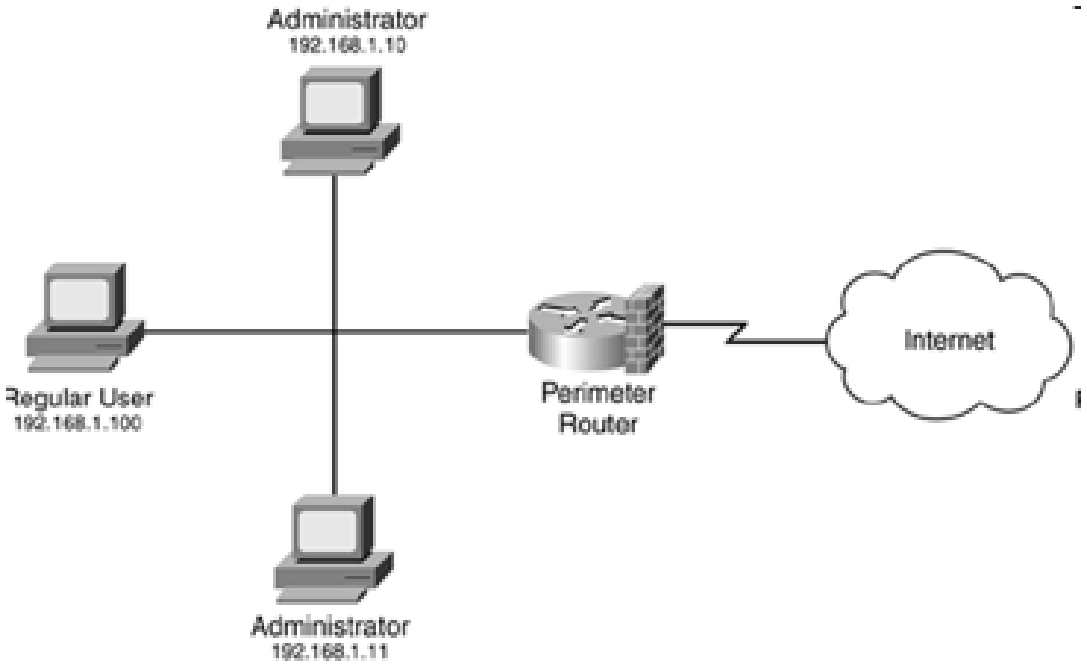
- Commande sur la ligne vty :
  - **access-class** [numéro] {in | out}

- Exemple :
  - **access-list 3 permit 10.1.1.0 0.0.0.255**
  - **line vty 0 4**
  - **login**
  - **password Cisco**
  - **access-class 3 in**

Conseils à suivre

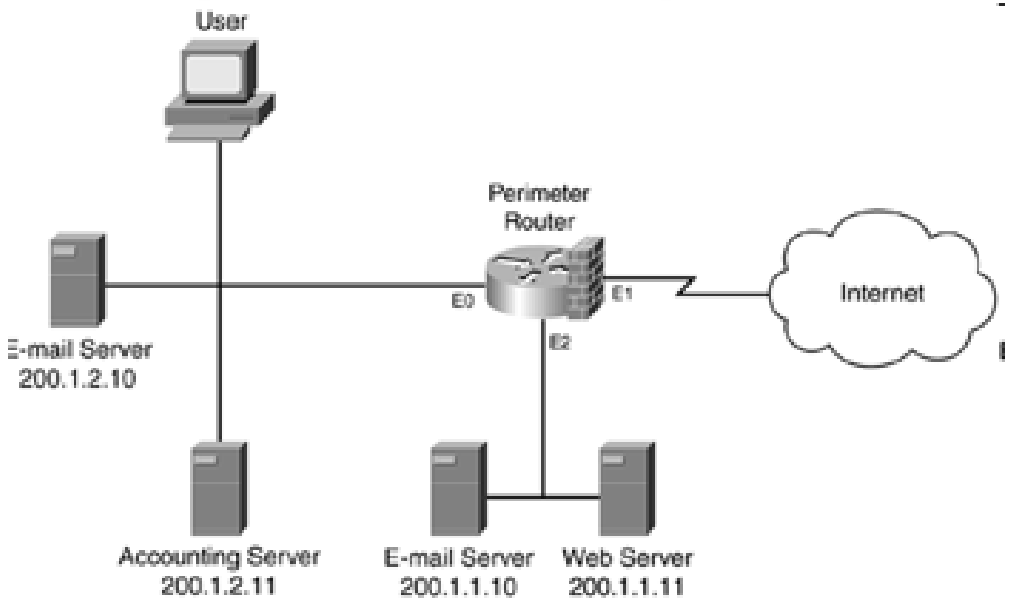
- Ne pas créer les ACL depuis un éditeur de texte pour les coller ensuite dans la configuration de l'équipement, mais travailler directement sur celui-ci.
- Placer les **ACLs étendues** au plus près de la **source du paquet** (pour le détruire le plus tôt possible).
- Placer les **ACLs standards** au plus près de la **destination du paquet** (pour ne pas détruire trop tôt un paquet en se basant sur des critères insuffisants).
- Placer la règle la plus spécifique en premier.
- Avant de faire un changement sur une ACL, la désactiver sur les interface où elle est appliquée (commande : "no ip access-group" en mode de configuration d'interface).

ACLs basiques exemple



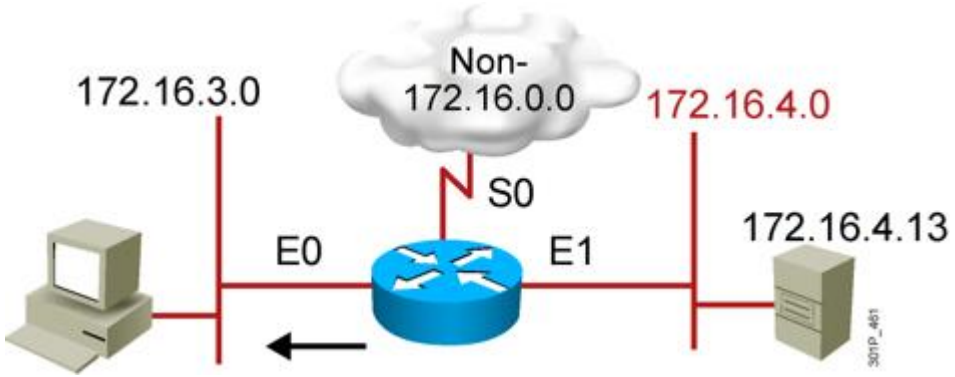
- Router(config)# ip access-list standard restrict_VTY
- Router(config-std-nacl)# permit 192.168.1.10
- Router(config-std-nacl)# permit 192.168.1.11
- Router(config-std-nacl)# exit
- Router(config)# line vty 0 4
- Router(config-line)# access-class restrict_VTY in

ACLs basiques exemple (2)



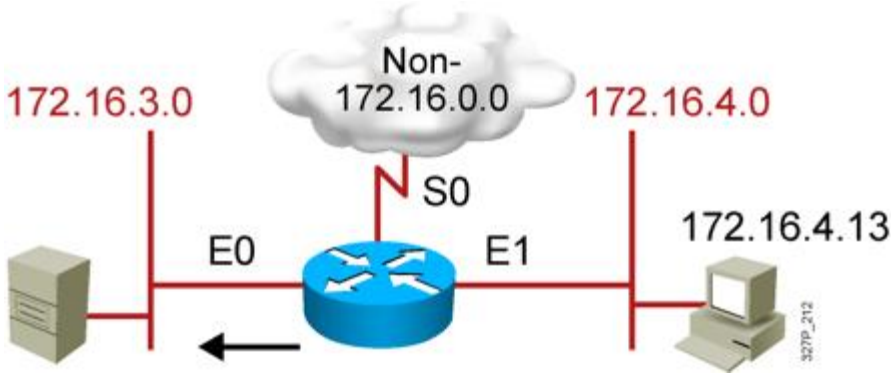
- Router(config)# ip access-list extended EX1
- Router(config-ext-nacl)# deny ip any 200.1.2.10 0.0.0.0
- Router(config-ext-nacl)# permit tcp any host 200.1.1.11 eq 80
- Router(config-ext-nacl)# permit tcp any host 200.1.1.10 eq 25
- Router(config-ext-nacl)# permit tcp any eq 25 host 200.1.1.10 any
- Router(config-ext-nacl)# permit tcp any 200.1.2.0 0.0.0.255
- Router(config-ext-nacl)# permit udp any eq 53 200.1.2.0 0.0.0.255
- Router(config-ext-nacl)# deny ip any any
- !
- Router(config-ext-nacl)# interface FastEthernet 1
- Router(config-if)# ip access-group EX1 in
- Router(config-if)# exit

ACLs basiques exemple (3)



- Router1(config)# access-list 1 deny 172.16.4.0 0.0.0.255
- Router1(config)# access-list 1 permit any
- !
- Router1(config)# interface FastEthernet 0
- Router1(config-if)# ip access-group 1 out

ACLs basiques exemple (4)



- Router1(config)# access-list 101 deny tcp 172.16.4.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 21
- Router1(config)# access-list 101 deny tcp 172.16.4.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 20
- Router1(config)# access-list 101 permit ip any any
- !
- Router1(config)# interface FastEthernet 1
- Router1(config-if)# ip access-group 101 in

ACLs basiques – utilisation des protocoles

- Autorise les paquets DNS, SMTP et FTP vers l'adresse 192.168.20.2 :
  - R1(config)#access-list 122 permit udp any host 192.168.20.2 eq domain
  - R1(config)#access-list 122 permit tcp any host 192.168.20.2 eq smtp
  - R1(config)#access-list 122 permit tcp any host 192.168.20.2 eq ftp

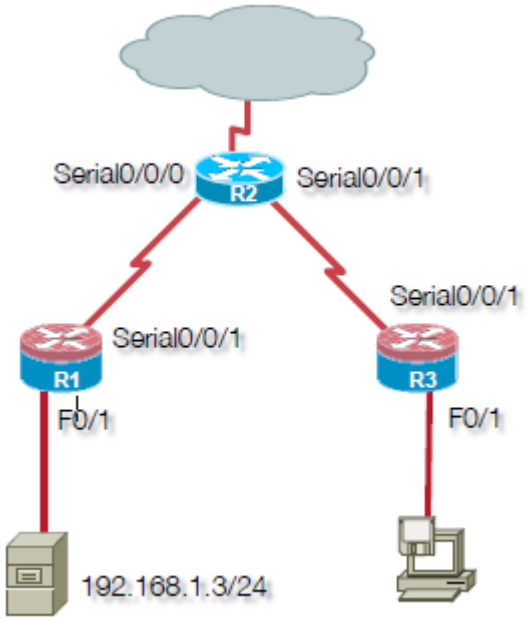
- Autorise les paquets Telnet, SSH, Syslog, et SNMP depuis 200.5.5.5 vers 10.0.1.1
  - R1(config)#access-list 180 permit tcp host 200.5.5.5 host 10.0.1.1 eq telnet
  - R1(config)#access-list 180 permit tcp host 200.5.5.5 host 10.0.1.1 eq 22
  - R1(config)#access-list 180 permit udp host 200.5.5.5 host 10.0.1.1 eq syslog
  - R1(config)#access-list 180 permit udp host 200.5.5.5 host 10.0.1.1 eq snmptrap

LES ACL AVANCÉES

Qu’est-ce que les ACLs avancées ?

- ACL étendues avec utilisation de “TCP established”.
- ACL réflexives.
- ACL dynamiques.
- ACL avec gestion du temps (Time-Based).
- ACL Context-Based Access Control (CBAC).

ACL « TCP established » - Exemple



- access-list 100 permit tcp any eq 443 192.168.1.0 0.0.0.255 established
- access-list 100 permit tcp any 192.168.1.3 eq 22
- access-list 100 deny ip any any
- !
- interface Serial0/0/0
- ip access-group 100 in

ACL « TCP established »

- Consiste à rajouter le mot-clé “**established**” à la fin d’une ACL avancée.
- Sur les en-têtes TCP, force une vérification de la présence des drapeaux **URG, ACK, PSH, RST, SYN** et **FIN**.
- Option ne s’appliquant pas aux trafics UDP et ICMP.
- Permet de se protéger des attaques “**man-in-the-middle**”.

ACLs Réflexives

- Création d'une ACL sur le trafic sortant qui recherche des nouvelles sessions sortantes et crée automatiquement des **ACE** (Access Control Entries) réflexives associées à un timeout (trois minutes par défaut).

- Création d'une ACL sur le trafic entrant qui utilise les ACLs réflexives pour examiner le trafic entrant de retour.

- Activation des ACLs sur les interfaces appropriées.

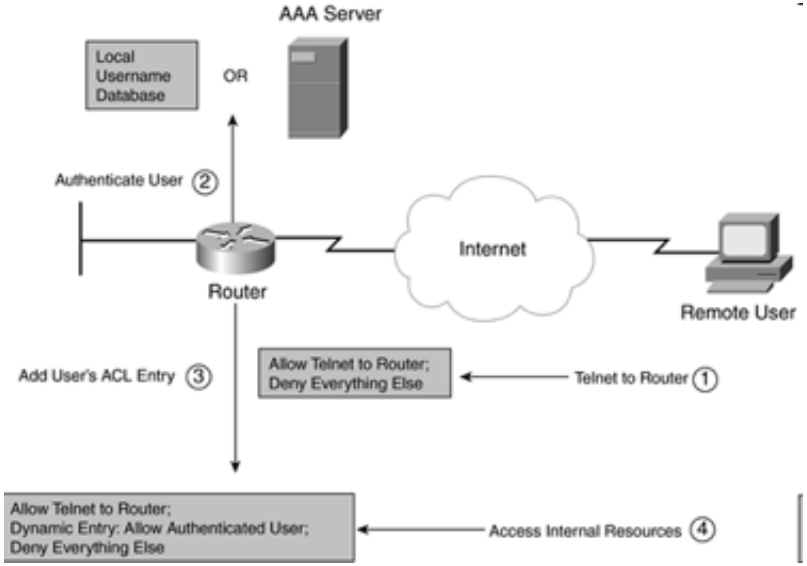
ACLs Réflexives - Exemples

- Création d'une ACL réflexive « RetourDNS »
  - ip access-list extended ToVlan5
    - permit udp any host 192.168.5.12 reflect RetourDNS
- Application de l'ACL réflexive « RetourDNS »
  - ip access-list extended ToVlan15
    - permit icmp any any echo-reply
    - permit tcp any any established
    - evaluate RetourDNS
    - deny ip any any

ACLs dynamiques

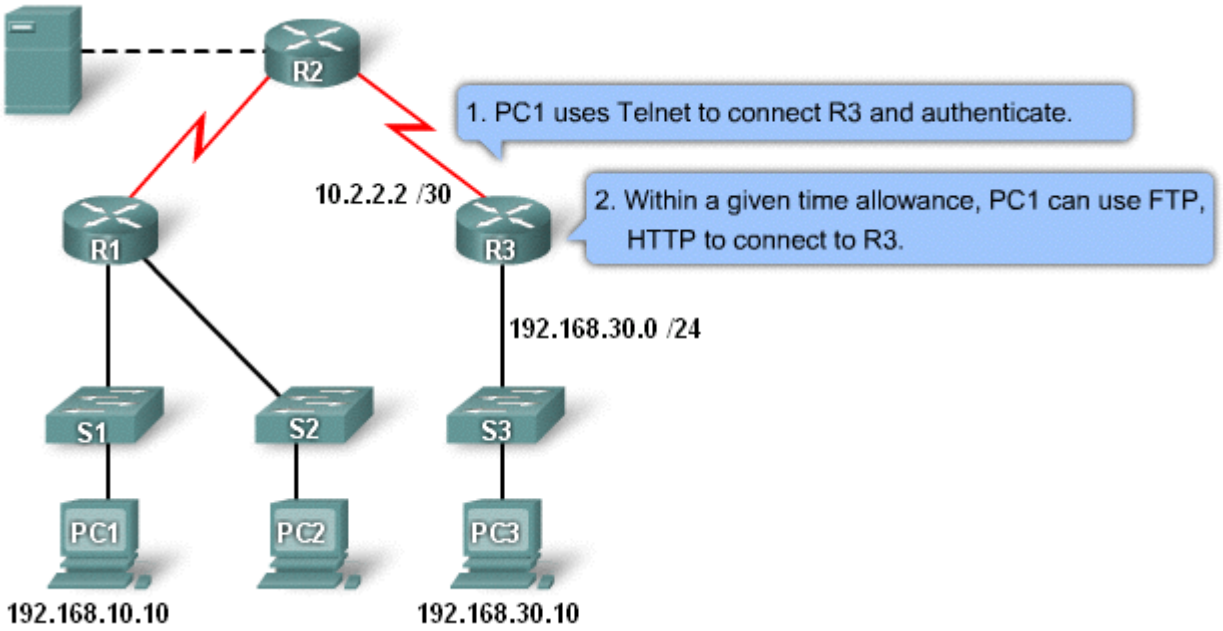
- Disponible seulement pour le trafic IP.
- ACL orientée connexion.
- Principe : à chaque connexion légitime d'un utilisateur, une ACL est créé pour lui ouvrir des droits spécifiques.
- **access-list** [numéro] **dynamic** [nom-acl-dynamique] **timeout** [minutes] {permit | deny} [protocole] [ip-source] [masque-générique] [ip-destination] [masque-générique].

ACLs dynamiques - exemple



- 1. Un utilisateur distant ouvre une connexion Telnet ou SSH sur le routeur et saisit ses identifiants
- 2. Le routeur authentifie l'utilisateur
- 3. Une règle d'ACL dynamique est ajoutée pour autoriser les accès à cet utilisateur
- 4. L'utilisateur peut enfin accéder aux ressources

ACLs dynamiques - exemple



- R3(config)#username jordan password 0 jordan
- R3(config)#access-list 101 permit tcp 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

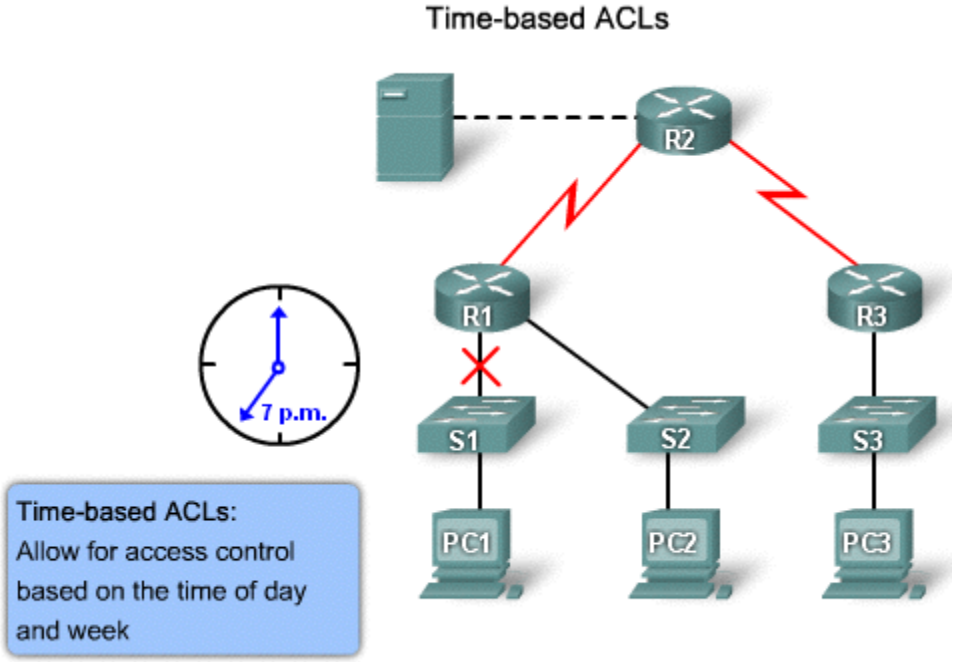
- R3(config)#interface serial 0/0/1
- R3(config-if)#ip access-group 101 in

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

ACL avec gestion du temps

- Time-Based ACL.
- Permet à une ACL de n’être activée qu’à certains moments.
- Utile pour restreindre les accès à des périodes particulières.

ACL avec gestion du temps - Exemple



- R3(config)#time-range PERIOD
- R3(config-time-range)#periodic Monday Wednesday Friday 8:00 to 17:00

- R3(config)#access-list 101 permit tcp 192.168.10.0 0.0.0.255 any eq telnet time-range PERIOD

- R3(config)#interface serial 0/0/0
- R3(config-if)#ip access-group 101 out

ACL avec gestion du temps - Exemple

- R3(config)#time-range employee-time
- R3(config-time-range)#periodic weekdays 12:00 to 13:00
- R3(config-time)# periodic weekdays 17:00 to 19:00
- R3(config-time)# exit

- R3(config)# access-list 100 permit tcp any host 200.1.1.11 eq 25
- R3(config)# access-list 100 permit tcp any eq 25 host 200.1.1.11 established
- R3(config)# access-list 100 permit udp any host 200.1.1.12 eq 53
- R3(config)# access-list 100 permit udp any eq 53 host 200.1.1.12
- R3(config)# access-list 100 permit tcp any 200.1.1.0 0.0.0.255 established time-range employee-time
- R3(config)# access-list 100 deny ip any any

- R3(config)# interface FastEthernet0/1
- R3(config-if)# ip access-group 100 in
- R3(config-if)# exit

- R3(config)# access-list 101 permit tcp host 200.1.1.11 eq 25 any
- R3(config)# access-list 101 permit tcp host 200.1.1.11 any eq 25
- R3(config)# access-list 101 permit udp host 200.1.1.12 eq 53 any
- R3(config)# access-list 101 permit udp host 200.1.1.12 any eq 53
- R3(config)# access-list 101 permit tcp 200.1.1.0 0.0.0.255 any time-range employee-time
- R3(config)# access-list 101 deny ip any any
- R3(config)# interface FastEthernet0/1
- R3(config-if)# ip access-group 101 out

LE PACKET FILTERING

Propriétés communes aux pare-feux

- Résistant aux attaques.
- Unique point de transit entre réseaux différents.
- Empêche l'exposition d'hôtes et d'applications sensibles à des utilisateurs non fiables.
- Prévient l'exploitation des failles de protocoles.
- Empêches les données "malveillantes" de parvenir aux serveurs et clients.
- Rend l'application de la politique de sécurité plus simple, évolutive et robuste.
- Simplifie la gestion de la sécurité en déchargeant la plupart des contrôles d'accès au réseau sur des matériels dédiés.

Type de filtrage

- Filtrage de paquets (packet-filtering) : par exemple, un routeur qui filtre sur les couches 3 et parfois 4.
- **Statefull** (orienté connexion) : se base sur l'état de la connexion (initiation, transfert de données en cours, ou terminaison) grâce aux ACLs réflexives/dynamiques.
- **Passerelles applicatives** (Proxy) : filtrage sur les couches 3, 4, 5, 6 et 7 (contrôle et filtre les actions effectuées dans les logiciels).
- **Translation d'adresses** : étend le nombre d'IP disponibles et masque la réalité du design de l'adressage réseau.
- **Host-based** : pare-feu applicatif installé directement sur un poste serveur ou client.
- **Hybride** : combinaison des autres types de filtrage.

Avantages du Packet-Filtering

- Basé sur des règles “permit” ou “deny”.
- Faible impact sur les performances réseau.
- Simple à mettre à en place.
- Supporté par la majorité des routeurs.
- Offre un premier niveau de sécurité sur des couches OSI basses (3 et parfois 4).
- Gère 90% des fonctionnalités des pare-feux haut de gamme à un coût beaucoup plus faible.

Inconvénients du Packet-Filtering

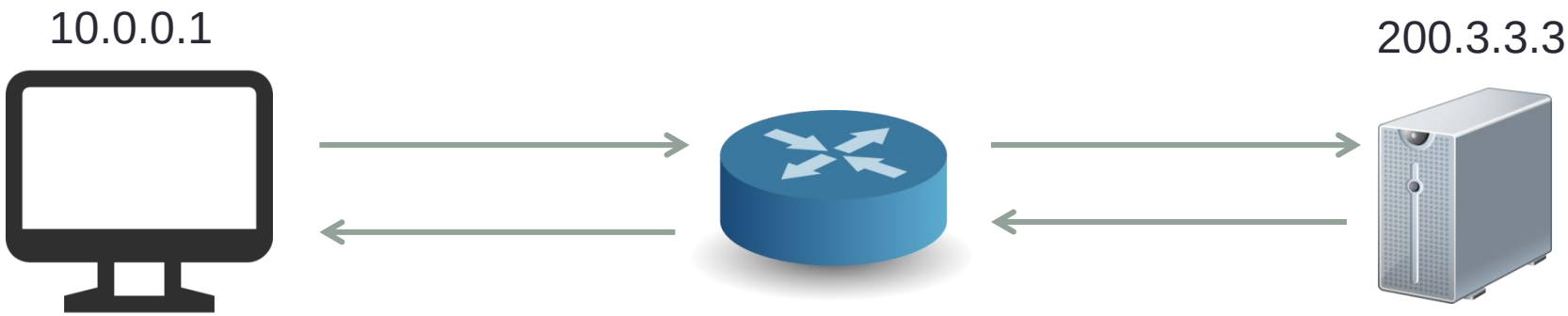
- Ne permet pas de se prémunir de l'IP spoofing (les pirates envoient des paquets aléatoires dont certains passent à travers les règles des ACL).
- Ne fonctionne pas bien avec des paquets fragmentés (si le premier paquet contenant les en-têtes de couche 4 est accepté, les autres le sont également sans condition).
- Les ACLs très complexes sont difficiles à appliquer correctement et à maintenir.
- Ne permet pas de filtrer dynamiquement certains services.

PARE-FEUX STATEFUL

Les pare-feux Stateful

ACLs entrantes (trafic sortant du LAN)	ACLs dynamiques (trafic à destination du LAN)

`permit ip 10.0.0.0 0.0.0.255 any`	`permit tcp host 10.0.0.1 eq 1500 host 200.3.3.3 eq 80`
	
	`deny ip any any`



Port source : 1500 → Port destination : 80

Les avantages du pare-feu stateful

- Souvent utilisé comme premier moyen de défense pour filtrer les trafics non désirés ou non nécessaires.
- Renforce le filtrage en apportant des règles plus strictes que le filtrage de paquets.
- Protège contre les attaques de spoofing ou de DoS.
- Donne davantage de logs que le filtrage de paquets.

Les inconvénients du pare-feu stateful

- Ne permet pas de se prémunir des attaques en couche application.
- Incompatible avec les protocoles qui ne sont pas orientés connexion (UDP, ICMP, ...).
- Difficile à maintenir avec des applications actuelles ouvrant toujours plus de connexions simultanément.

Les solutions Cisco ASA

- Politique de gestion des zones assez intuitive.
- Filtrage des applications de messagerie instantanée et de P2P.
- Protection des protocoles de VoIP/ToIP.
- Protection des VRF (Virtual Routing and Forwarding).
- Intégration de la sécurité des réseaux sans fil (Wifi).
- Gestion des “listes blanches” et listes noires” d’URL.
- Inspection des trafics Internet et mail.
- Implémentation de technologies de pare-feu
- Les pare-feux

Les règles de bonnes pratiques

- Positionner les pare-feux en **bordure de réseau**.
- Ne pas s'appuyer exclusivement sur les pare-feux, mais utiliser d'autres dispositifs de sécurité.
- **Refuser tous les trafics par défaut**, puis n'autoriser que ceux qui sont nécessaires.
- S'assurer que les pare-feux sont physiquement installés dans **un espace sécurisé.**
- Monitorer les pare-feux avec grande attention.
- Utiliser des outils d'historisation des modifications de configuration sur les pare-feux (exemple : **résilience de la configuration**).
- Se souvenir que les pare-feux ne protègent pas des attaques provenant de l'intérieur du réseau.