

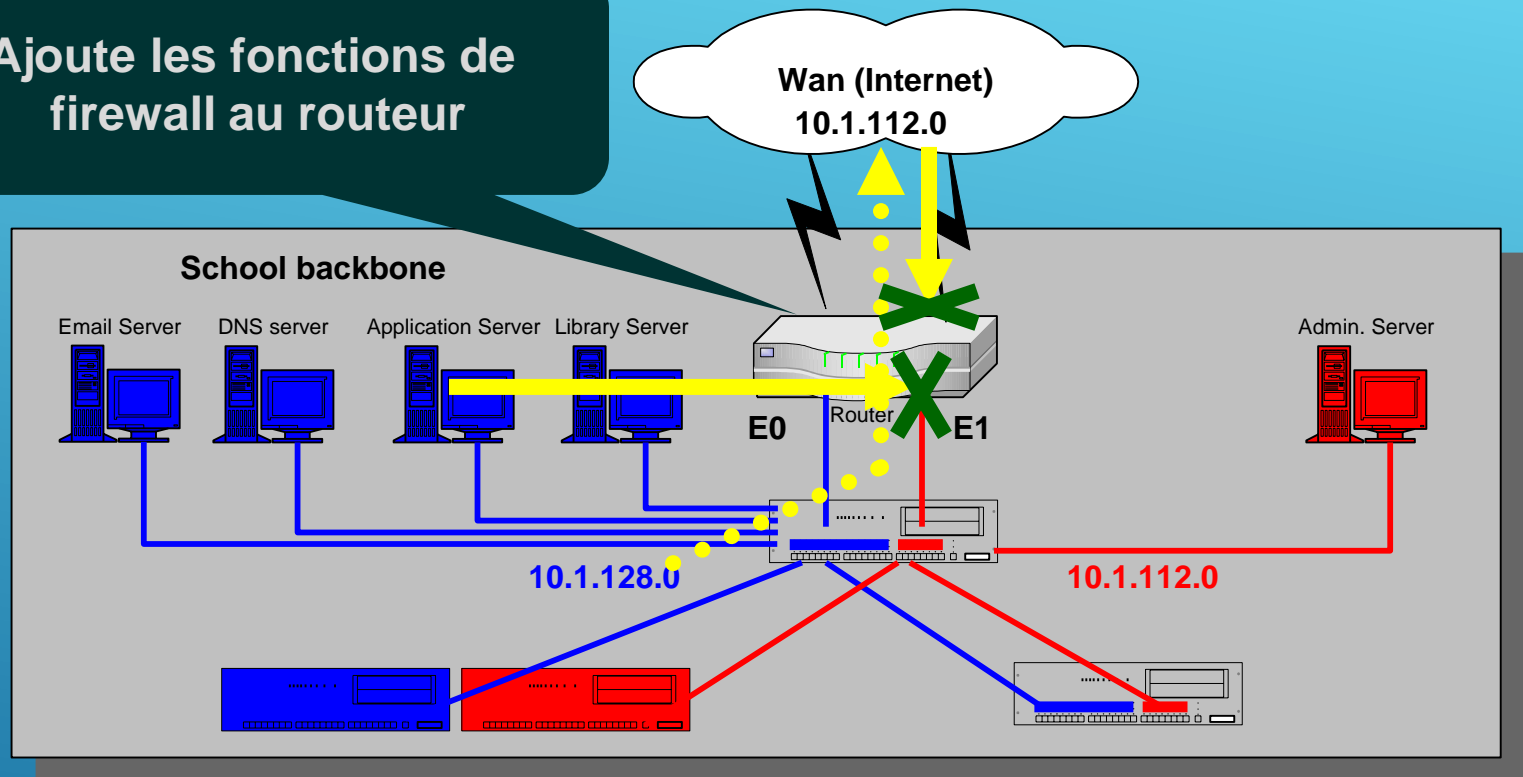
# **Les Listes de Contrôle d'Accès**

## **(ACL : Access Control Lists)**

- 1. Généralités Listes de Contrôle d'Accès**
- 2. Sécuriser l'accès au routeur**
- 3. Dynamic Access-List - Lock and Key**
- 4. Filtrage de sessions**
- 5. CBAC (Context Based Access Control)**
- 6. Alternative aux Listes de Contrôle d'Accès**

# ACL - A QUOI ÇA SERT

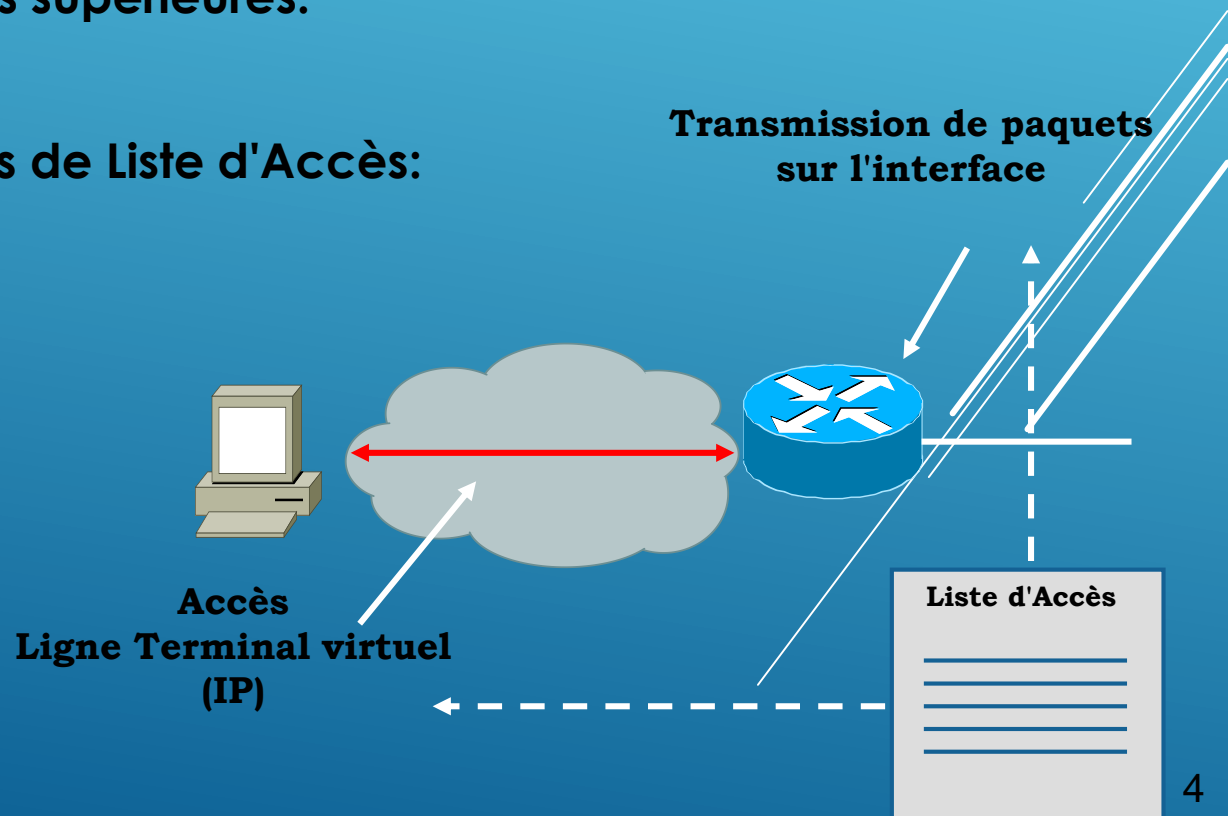
Ajoute les fonctions de firewall au routeur



- Contrôler le trafic à l'intérieur d'un réseau local
- Contrôler le trafic depuis l'extérieur (WAN) vers l'intérieur d'un réseau local

# ACL | GÉNÉRALITÉS

- ▶ Une Liste d'Accès est une séquence d'actions d'autorisation (permit) ou d'interdiction (deny) sur des adresses ou des protocoles de couches supérieures.
- ▶ Il existe différents types de Liste d'Accès:
  - ▶ Standard (Standard)
  - ▶ Étendue (Extended)
  - ▶ Nommée (Named)
  - ▶ Dynamiques



# ACL | NUMÉROS DE LISTES D'ACCÈS

<b>Numéro de Liste d'Accès</b>	<b>Description</b>
<b>1 à 99</b>	<b>Access List Standard IP</b>
<b>100 à 199</b>	<b>Access List Etendue IP</b>
<b>200 à 299</b>	<b>Protocole Type-Code Access List</b>
<b>300 à 399</b>	<b>DECnet Access List</b>
<b>400 à 499</b>	<b>XNS Access List Standard</b>
<b>500 à 599</b>	<b>XNS Access List Etendue</b>
<b>600 à 699</b>	<b>Apple Talk Access List</b>
<b>700 à 799</b>	<b>Adresses MAC Acces List Standard</b>
<b>800 à 899</b>	<b>IPX Access List Standard</b>
<b>900 à 999</b>	<b>IPX Access List Etendue</b>
<b>1000 à 1099</b>	<b>IPX SAP Access List</b>
<b>1100 à 1199</b>	<b>Adresses MAC Acces List Etendue</b>
<b>1200 à 1299</b>	<b>IPX Adresses agrégées Access list</b>
<b>1300 à 1399</b>	<b>Access List Standard IP (extension)</b>
<b>2000 à 2699</b>	<b>Access List Étendue IP (extension)</b>

# ACL STANDARD OU ÉTENDUE ?

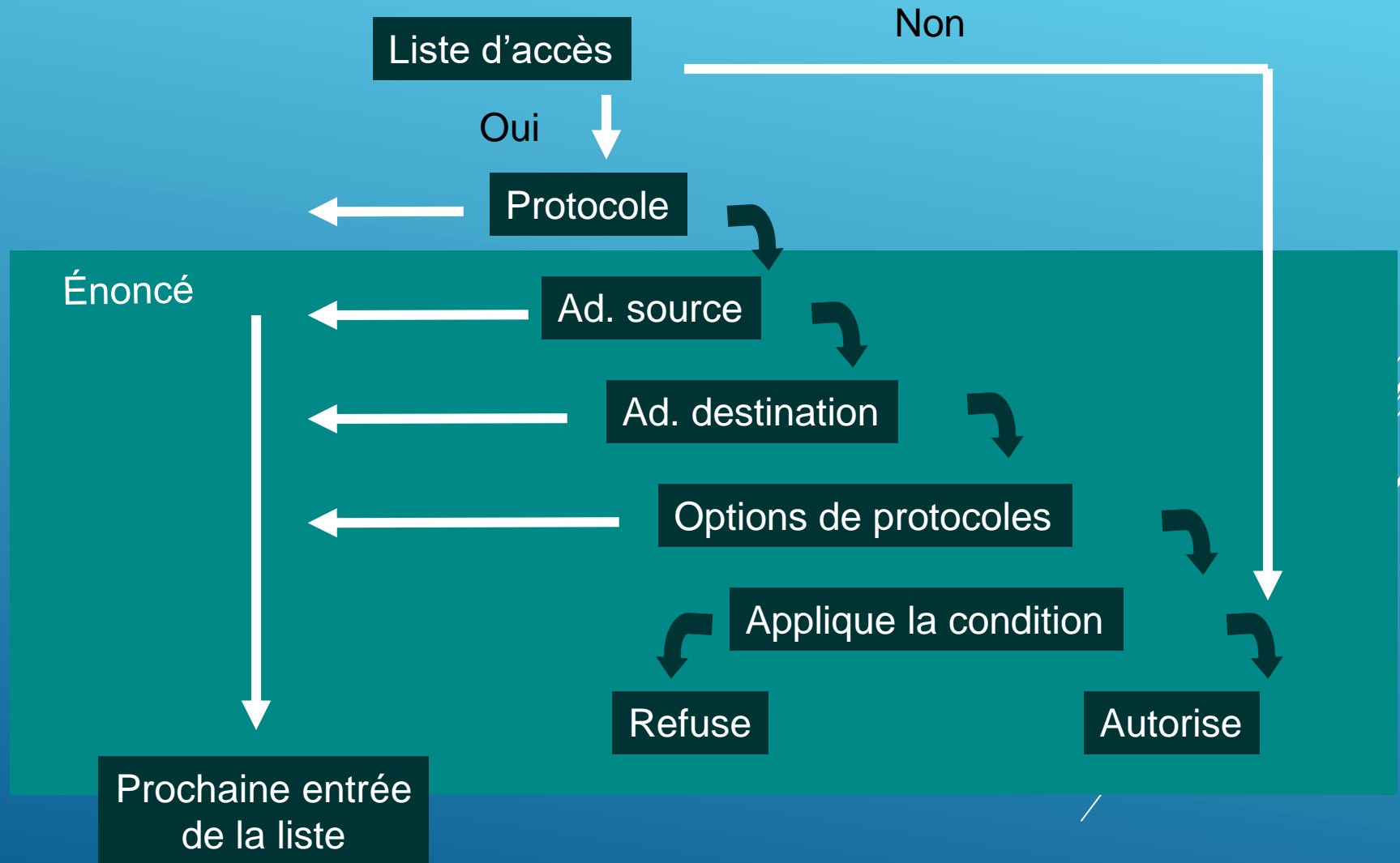
- ▶ **Les listes de contrôles standards filtrent l'accès :**
  - à partir de l'adresse source uniquement .

```
Router(config)#access-list access-list-number {deny|permit}  
source [source-wildcard] [log]
```

- ▶ **Les listes de contrôle étendues peuvent filtrer l'accès :**
  - selon l'adresse de source et de destination ;
  - selon des champs IP;
  - selon les types de protocole de transport (TCP, UDP);
  - et le numéro de port (couche application).

```
Router(config)#access-list access-list-number {deny|permit}  
source [source-wildcard] destination destination-wildcard  
eq num_port [established]
```

# LISTE DE CONDITIONS DE L'ACL ÉTENDUE



# ACL NOMMÉE

- Les Access-Lists nommées ne sont pas compatibles avec les releases inférieures à la Release IOS 11.2.
- Un nom ne peut pas être utilisé pour plusieurs type d'Access-lists.
- Usuellement, seuls les filtres de routes et de paquets peuvent utiliser les Access-lists nommées.
- Utilisez **no permit** et **no deny** pour retirer des entrées individuelles de la liste

```
RTA(config)#ip access-list extended WEBONLY
RTA(config-ext-nacl)#permit tcp any 10.0.0.0 0.255.255.255 eq 80
RTA(config-ext-nacl)#deny ip any 10.0.0.0 0.255.255.255
RTA(config-ext-nacl)#permit ip any any
RTA(config-ext-nacl)#^Z
```

```
RTA#show access-lists
Extended IP access list WEBONLY
    permit tcp any 10.0.0.0 0.255.255.255 eq www
    deny ip any 10.0.0.0 0.255.255.255
    permit ip any any
```



# ACL STANDARD | CONFIGURATION

- **Definit une liste d'accès IP standard en utilisant un nom:**

```
router(config)# ip access-list standard name
```

1. **Dans le mode de configuration access-list , spécifiez une ou plusieurs conditions *permit* ou *deny*:**

```
router(config-std-nacl)# deny| permit {source [source-wildcard] | any}[log]
```

2. **Exit access-list configuration mode:**

```
router(config-std-nacl)# exit
```

**Important!: Doit être appliquée sur le routeur le plus proche de la destination**

# ACL ÉTENDUE | CONFIGURATION

- **Définit une liste d'accès IP étendue en utilisant un nom :**

```
router(config)#ip access-list extended name
```

- 1. Dans le mode de configuration access-list , spécifiez une ou plusieurs conditions permit ou deny:**

```
router(config-ext-nacl)# deny | permit protocol source  
source-wildcard destination destination-wildcard  
[precedence precedence] [tos tos] [established] [log]  
[time-range time-range-name]
```

- 2. Sortie du mode de configuration access-list :**

```
router(config-ext-nacl)#exit
```

**Important!: Doit être appliquée sur le routeur le plus proche de la source**

# ACL ÉTENDUE | CONFIGURATION DE COMMENTAIRES

- ▶ **Rend la configuration des ACLs plus facile à lire**
- ▶ **Disponible depuis la release 12.0.2(T) de l'IOS Cisco**
- ▶ **Commande:**

**router (config) #access-list** *numéro* **remark** *remarque*  
**router (config-std-nacl) #remark** *remarque*

```
RTA(config)#access-list 101 remark Autorise Admin Sous-réseau Telnet vers Serveurs
RTA(config)#access-list 101 permit tcp 192.168.1 0.0.0.255 172.16.1.0 0.0.0.255 eq 23
RTA(config)#access-list 101 deny tcp any 172.16.1.0 0.0.0.255 eq 23
RTA(config)#access-list 101 remark autorise SNMP pour Admin du host uniquement
RTA(config)#access-list 101 permit udp host 192.168.1.250 any eq 161
RTA(config)#access-list 101 deny udp any any eq 161
RTA(config)#access-list 101 permit ip any any
```

# ACL | APPLIQUER DES ACCESS-LISTS

- ▶ Les listes de contrôle d'accès sont appliquées à:
  - ▶ Une ou plusieurs interfaces
  - ▶ Pour du trafic entrant ou sortant
- ▶ Rappelez-vous que vous pouvez appliquer une liste de contrôle d'accès par protocole, par interface et par direction (in ou out)
- ▶ Les Accès-List en sortie demandent moins de temps CPU que les listes d'accès en entrées et par conséquent sont préférables.
- ▶ Commande:

**Router(config-if)#ip access-group** *numéro\_ACL* | *Nom\_ACL* **in | out**

# TIME-BASED EXTENDED ACCESS-LIST

- ▶ Utilisation d'une "time-based Access List" depuis l'IOS release 12.01(T) la commande time-range
- ▶ Bénéfice de l'utilisation de "Time based" Access List :
  1. Permet plus de contrôle sur l'autorisation d'accès à des ressources pour un utilisateur
  2. Fonctions de politiques de routage et mise en file d'attente améliorées
  3. Efficacité pour le coût de reroutage automatique de trafic
  4. Supporte la qualité de service (QoS), service-level agreements (SLAs) quand les fournisseurs de service peuvent changer dynamiquement le Committed Access rate (CAR).
  5. Contrôle des messages de "logging"

# TIME-BASED EXTENDED ACCESS-LIST | CONFIGURATION

## 1. Définir un nom d'intervalle de temps:

```
router(config) # time-range time-range-name
```

## 2. Utilisez les commandes **periodic** et **absolute**:

```
router(config-time-range) # periodic days-of-the-week hh:mm  
                           to [days-of-the-week] hh:mm
```

```
router(config-time-range) # absolute [start time date]  
                           [end time date]
```

## 3. Sortie du mode de configuration intervalle de temps:

```
router(config-time-range) # exit
```

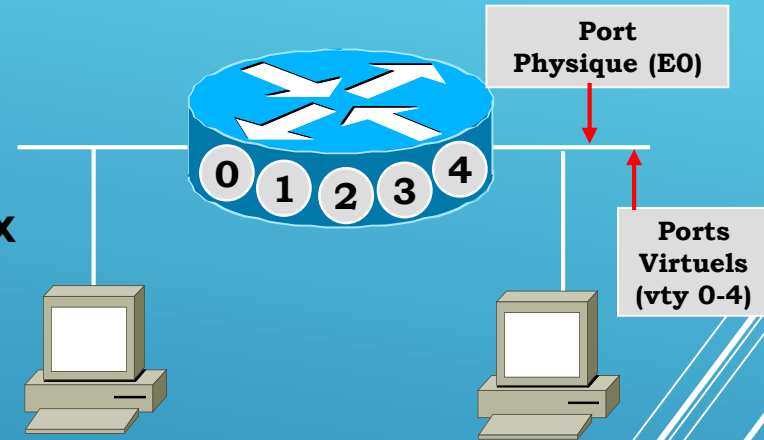
# TIME-BASED EXTENDED ACCESS-LIST | EXEMPLE DE CONFIGURATION

- **Seules les listes d'accès nommées ou étendues IP et IPX peuvent utiliser des intervalles de temps.**
- **Dans la configuration ci-dessous, RTA est configuré avec la liste-d'accès nommée STRICT et deux intervalles de temps, NO-HTTP et UDP-YES:**
  - **L'instruction deny empêche le trafic web les jours de la semaine de 8h à 18h.**
  - **L'instruction permit autorise le trafic UDP le week-end de 12h à 20h**

```
RTA(config)#time-range NO-HTTP
RTA(config-time-range)#periodic weekdays 8:00 to 18:00
RTA(config-time-range)#exit
RTA(config)#time-range UDP-YES
RTA(config-time-range)#periodic weekend 12:00 to 20:00
RTA(config-time-range)#exit
RTA(config)#ip access-list extended STRICT
RTA(config-ext-nacl)#deny tcp any any eq http time-range NO-HTTP
RTA(config-ext-nacl)#permit udp any any time-range UDP-YES
RTA(config-ext-nacl)#deny udp any any range netbios-ns netbios-ss
RTA(config-ext-nacl)#permit ip any any
```

# ACL | SÉCURISER L'ACCÈS AU ROUTEUR - LES PORTS "TERMINAL VIRTUEL"

- ▶ • Les listes d'accès étendues peuvent être utilisées pour bloquer Telnet (TCP 23)
  - Doivent être configurées pour chaque interface IP sur le routeur
- Appliquer une liste d'accès standard aux lignes vty est un meilleur choix.



- `RTA(config)#access-list 5 permit 200.100.50.0 0.0.0.255`
- `RTA(config)#access-list 5 permit host 192.168.1.1`
- `RTA(config)#line vty 0 4`
- `RTA(config-line)#access-class 5 in`



# ACL | SÉCURISER L'ACCÈS AU ROUTEUR - SERVEUR WEB IOS

```
RTA(config)#access-list 17 permit 202.206.100.0 0.0.0.255
```

```
RTA(config)#ip http server
```

```
RTA(config)#ip http access-class 17
```

- L'application d'une access-class à un serveur HTTP IOS ne nécessite pas le mot clé in car ce service accepte uniquement des connexions entrantes.

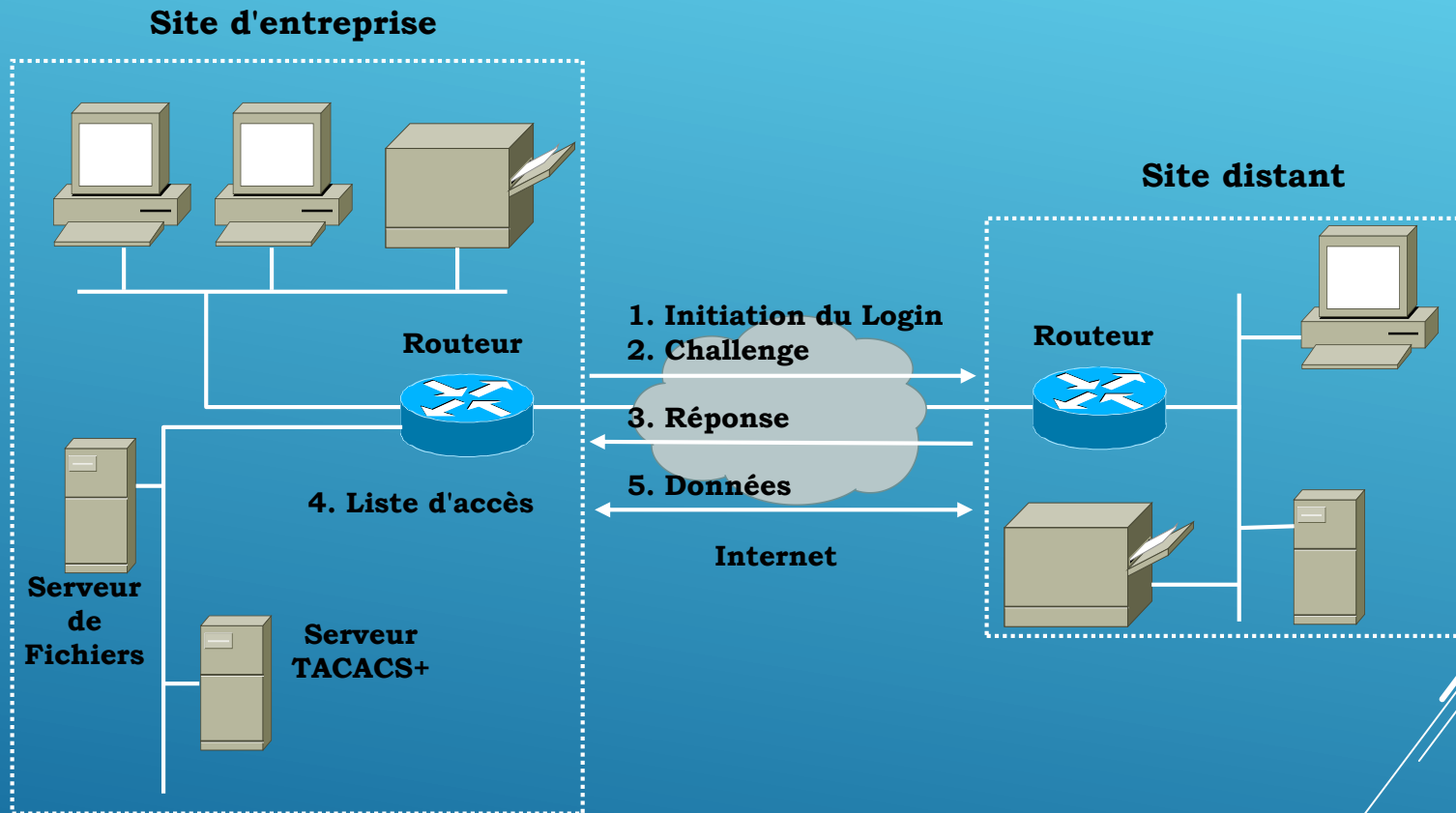
# ACL DYNAMIQUES | LOCK AND KEY

1. **Vous voulez permettre à un utilisateur ou à un groupe d'utilisateurs d'accéder de manière sécurisée à un host de votre réseau protégé via Internet.**
  - **"Lock-and-Key" authentifie l'utilisateur permet un accès limité au travers de votre routeur pare-feu mais uniquement pour ce host ou ce sous-réseau et pour une durée déterminée.**
2. **Vous voulez que certains utilisateurs d'un réseau distant accèdent à un host du réseau d'entreprise protégé par un pare-feu.**
  - **"Lock and Key" requiert l'authentification des utilisateurs avant d'autoriser l'accès à des hosts protégés.**

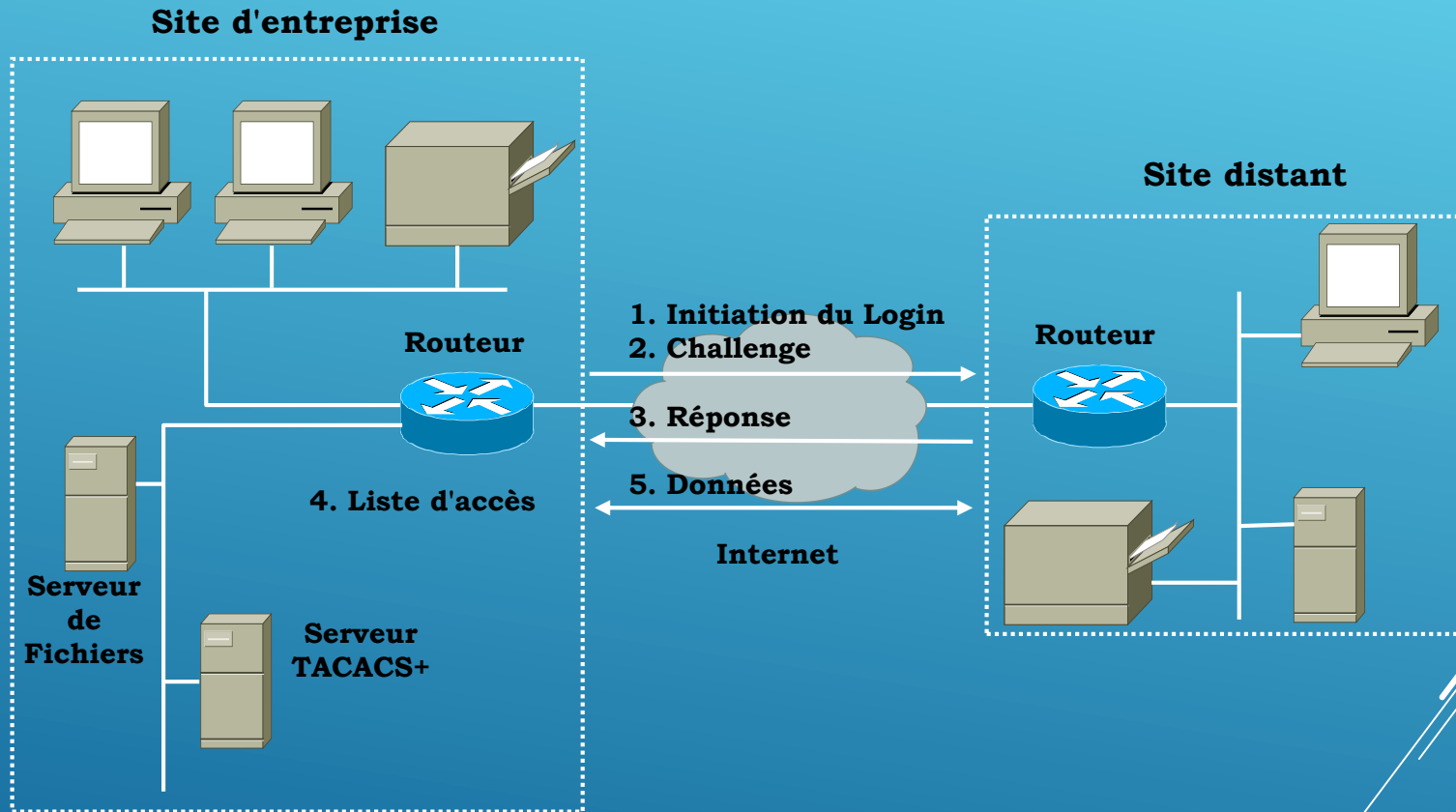
# ACL DYNAMIQUES | LOCK AND KEY FONCTIONNEMENT



# ACL DYNAMIQUES | LOCK AND KEY FONCTIONNEMENT

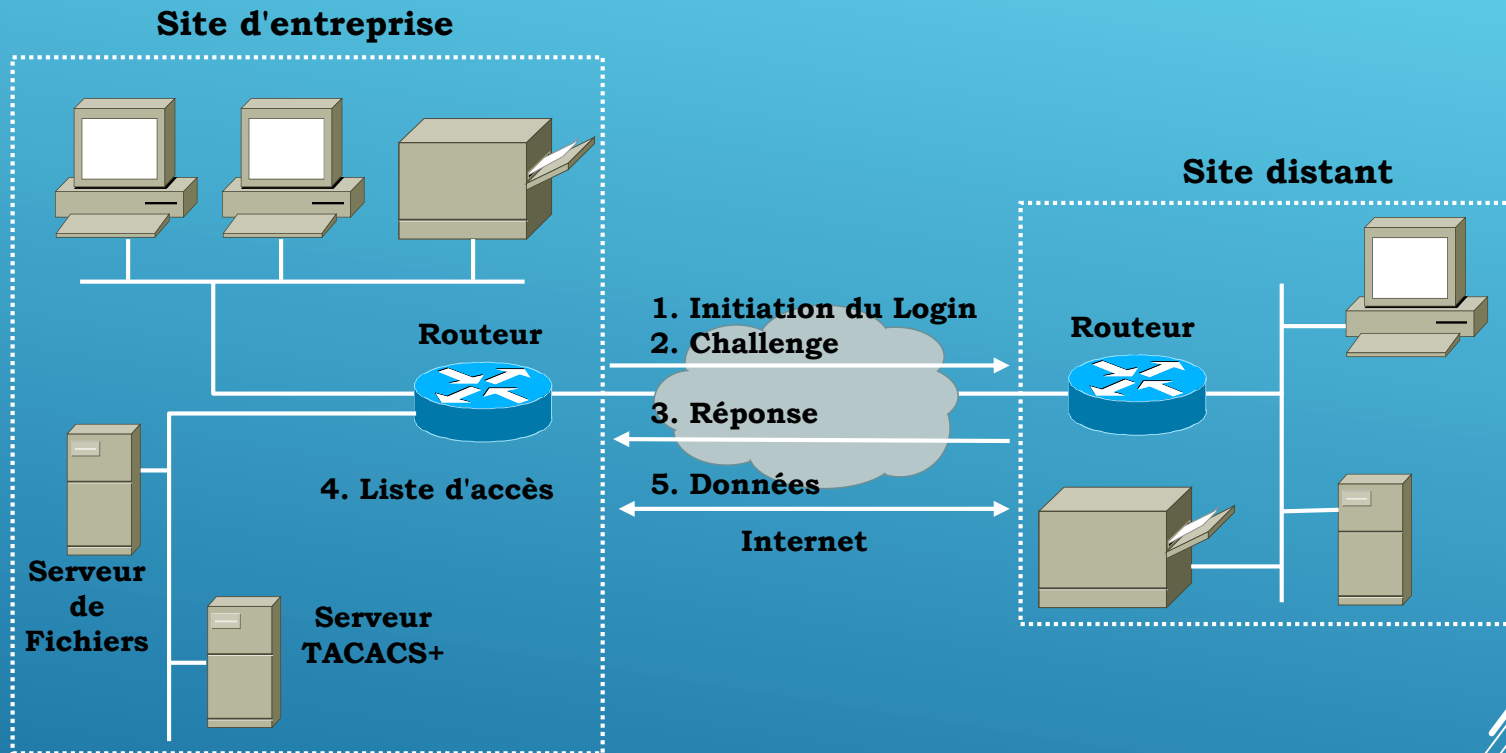


# ACL DYNAMIQUES | LOCK AND KEY FONCTIONNEMENT



**1. L'utilisateur ouvre une session Telnet vers un routeur pare-feu configuré pour "Lock and Key"**

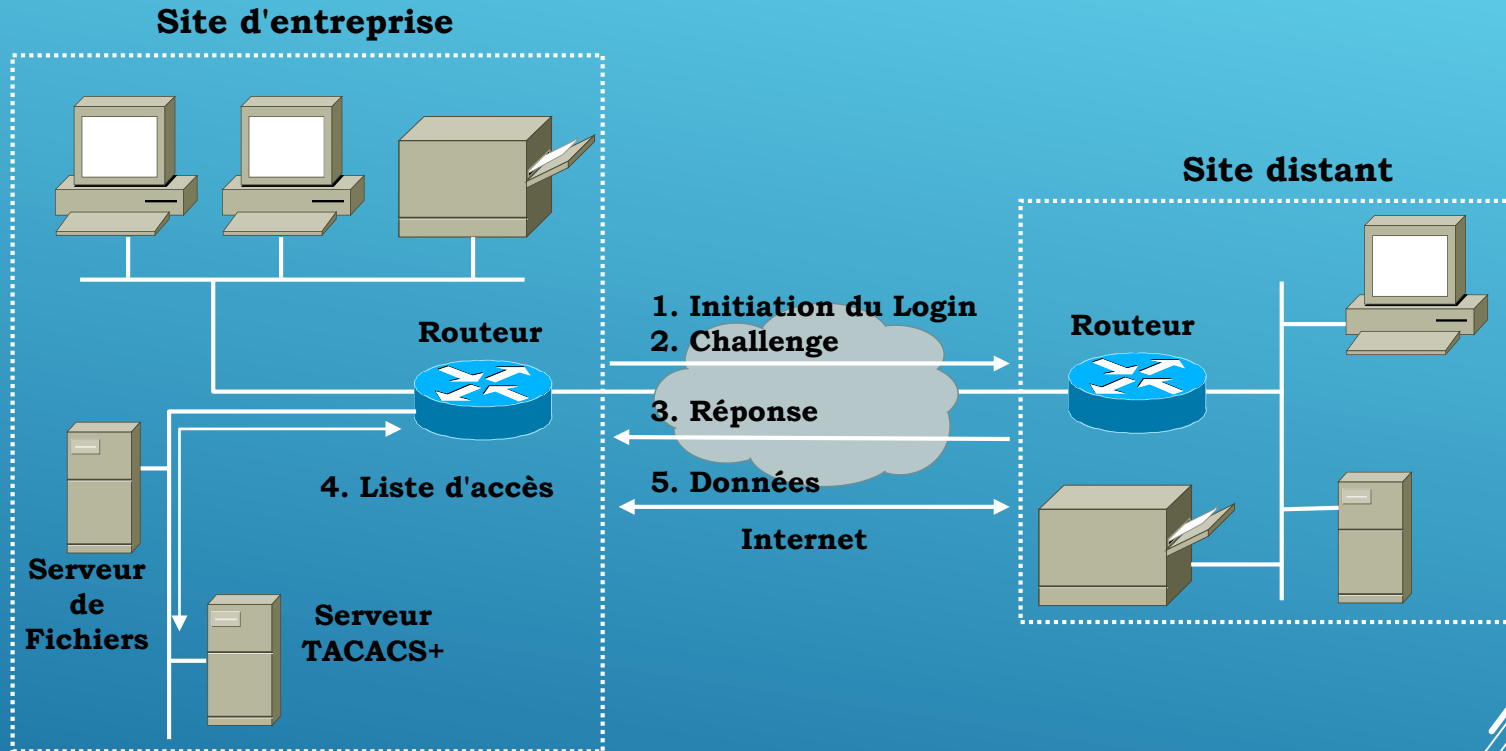
# ACL DYNAMIQUES | LOCK AND KEY FONCTIONNEMENT



## 2. IOS sur Routeur

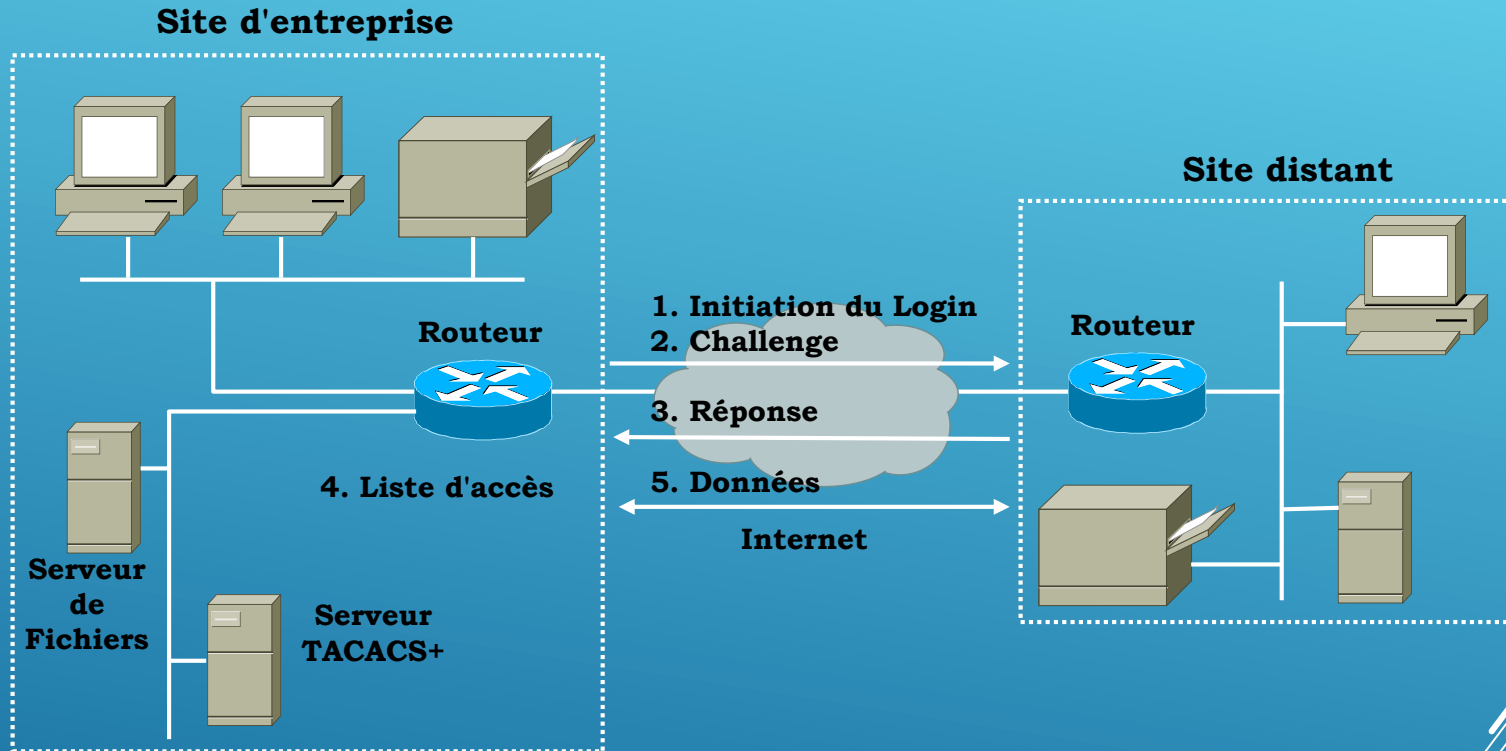
- a- Le routeur reçoit un paquet Telnet
- b- Le routeur ouvre une session Telnet
- c- Le routeur demande un nom d'utilisateur et un mot de passe
- d- Le routeur réalise l'authentification avec le serveur TACACS+
- e- Si l'authentification est réussie, la connexion Telnet est libérée
- f- L'IOS Cisco crée une entrée temporaire dans la liste d'accès dynamique

# ACL DYNAMIQUES | LOCK AND KEY FONCTIONNEMENT



3. L'utilisateur échange des données à travers le pare-feu

# ACL DYNAMIQUES | LOCK AND KEY FONCTIONNEMENT



**4. L'IOS efface l'entrée temporaire de la liste d'accès quand une durée configurée est atteinte**

**ou lorsque l'administrateur efface cette entrée.**

- La durée peut un temps donné ou une valeur absolue.
- L'entrée temporaire de la liste d'accès n'est pas automatiquement effacée lorsque l'utilisateur termine la session .



# ACL DYNAMIQUES | LOCK AND KEY CONFIGURATION

## ► Définition d'une liste d'accès dynamique

```
RTA(config)#access-list 101 permit tcp any host 192.168.1.1 eq telnet
```

```
RTA(config)#access-list 101 dynamic UNLOCK timeout 120 permit ip any any
```

```
RTA(config)#int s0
```

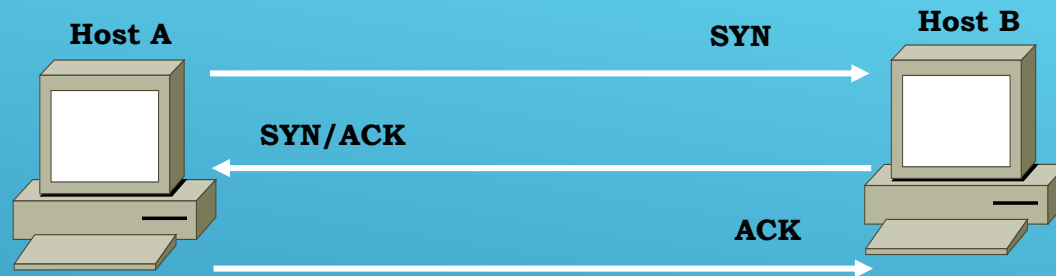
```
RTA(config-if)#ip acces-group 101 in
```

## ► timeout xxx , dans cette commande est un timeout absolu (en minutes)

# ACL DYNAMIQUES | LOCK AND KEY CONFIGURATION DE L' AUTHENTIFICATION

- **Utilisation de sa propre base de données locale ou centralisée sur un serveur RADIUS ou TACACS+.**
- **Configuration utilisant une base de données locale pour l'authentification**  
RTA(config)#username usms password ensak  
RTA(config)#line vty 0 4  
RTA(config-line)#login local
- **L'étape finale pour configurer "Lock and Key" est d'autoriser le routeur à créer une entrée d'accès temporaire dans la liste d'accès dynamique.**
  - **commande : access-enable [host] [timeout minutes]**  
La valeur "idle timeout" en minutes (dans cette commande) doit être inférieur à la valeur absolue du timeout.
- **Configuration des lignes VTY**  
RTA(config)#line vty 0 4  
RTA(config-line)#autocommand access-enable host timeout 20
  - **autocommand access-enable : L'accès au travers du pare-feu est crée chaque fois que l'utilisateur s'authentifie via Telnet.**

# ACL PAR FILTRAGE DE SESSIONS



- **Comment autoriser le trafic permis à entrer et d'interdire le trafic non autorisé**
- **Filtrage du trafic basé 6 bits de code TCP :**
  - URG (Urgent)
  - PSH (Push)
  - RST (Reset)
  - FIN (Finish)
  - ACK (Acknowledgement)
  - SYN (Synchronization)
- **Les hosts utilisent TCP pour établir une connexion en trois étapes. Cet échange utilise les bits SYN et ACK.**
- **La liste d'accès étendue peut vérifier si un paquet fait partie d'une connexion déjà établie (Established)**

# ACL PAR FILTRAGE DE SESSIONS | ARGUMENT « ESTABLISHED »

- ▶ established argument utilisé avec le mot clé TCP dans une liste d'accès étendue (UDP, ICMP et tous les autres protocoles ne peuvent pas utiliser cet argument)

router(config)#access-list access-list-number permit tcp  
source-address source-mask destination-address destination-  
mask established

- ▶ Exemple:

```
access-list 101 permit tcp any 192.168.1.0 0.0.0.255 established
access-list 101 permit icmp any any
access-list 101 permit udp any any eq 53
access-list 101 deny ip any 192.168.1.0 0.0.0.255
access-list 101 permit ip any any
```

# ACL RÉFLEXIVE

- Les listes d'accès "Reflexive" permettent de filtrer le trafic du réseau sur la base des informations de session des protocoles situés au-dessus d'IP.
- Comme pour l'argument *established*
  - Les listes d'accès "Reflexive":
    - Autorisent les sessions issues de l'intérieur du réseau.
    - Interdisent les sessions issues de l'extérieur du réseau.
- Contrairement à l'argument *established*
  - Les listes d'accès "Reflexive":
    - Fonctionnent pour tous les protocoles et pas uniquement TCP.
    - Réalisent une correspondance dynamique entre le trafic entrant et les paramètres du trafic sortant .
- Peuvent être définies uniquement avec les listes d'accès étendues nommées
- Moyen très efficace pour sécuriser un réseau car elle évite la majorité des cas de "spoofing" ou de déni de service (DoS).
- Fournit un contrôle accru sur le trafic entrant et simple à utiliser

# ACL RÉFLEXIVE | CONFIGURATION

```
interface Serial 1
  description Accès à Internet via cette interface
  ip access-group filtresentrants in
  ip access-group filtressortants out
  !
  ip reflexive-list timeout 120
  !
  ip access-list extended filtresortants
    permit ip any any reflect TRAFIC
  !
  ip access-list extended filtresentrants
    permit bgp any any
    permit eigrp any any
    deny icmp any any
    evaluate TRAFIC
```



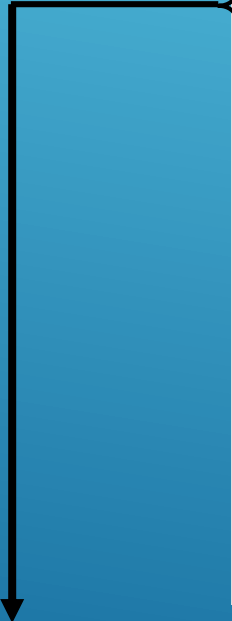
1. Crée la liste d'accès

# ACL RÉFLEXIVE | CONFIGURATION

```
interface Serial 1
  description Accès à Internet via cette interface
  ip access-group filtresentrants in
  ip access-group filtressortants out
  !
  ip reflexive-list timeout 120
  !
  ip access-list extended filtressortants
    permit tcp any any reflect tcptraffic
  !
  ip access-list extended filtresentrants
    permit bgp any any
    permit eigrp any any
    deny icmp any any
    evaluate tcptraffic
```

- 
1. Crée la liste d'accès
  2. Crée le filtre

# ACL RÉFLEXIVE | CONFIGURATION



```
interface Serial 1
  description Accès à Internet via cette interface
  ip access-group FILTRESEINTRANTS in
  ip access-group FILTRESSORTANTS out
  !
  ip reflexive-list timeout 120
  !
  ip access-list extended FILTRESSORTANTS
    permit udp any any reflect UDP_TRAFFIC
  !
  ip access-list extended FILTRESEINTRANTS
    permit bgp any any
    permit eigrp any any
    deny icmp any any
    evaluate UDP_TRAFFIC
```

1. Crée la liste d'accès
2. Crée le filtre
3. l'applique à une interface



# ACL RÉFLEXIVE | FONCTIONS DES ACL

## « RÉFLEXIVE »

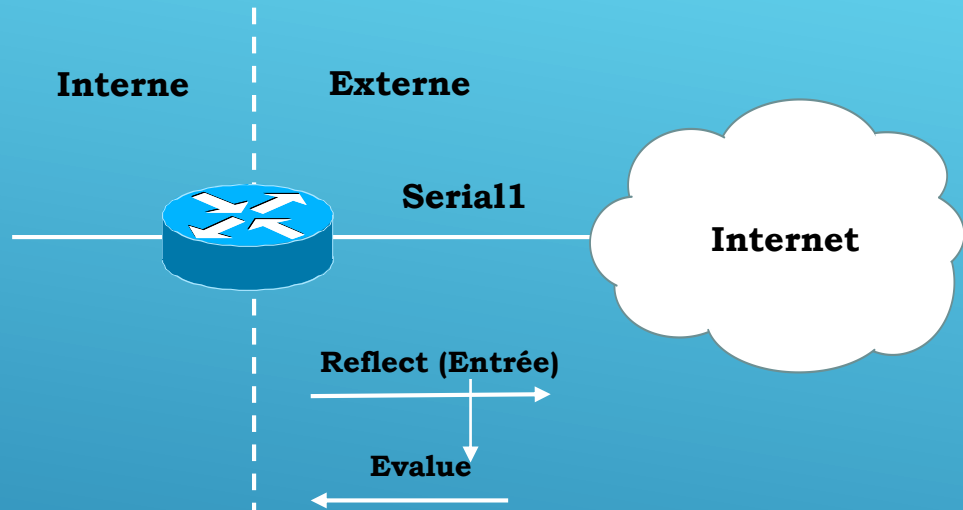
- ▶ **Les listes d'accès "Réflexive" contiennent des critères pour définir des accès ou entrées conditionnelles et temporaires.**
  - ▶ Création d'une nouvelle session depuis l'intérieur du réseau et fermeture de celle-ci quand elle se termine.
- ▶ **Évaluation de ces entrées en séquence et lorsqu'une correspondance est trouvée, l'évaluation est terminée.**
- ▶ **Difficile à tromper car plusieurs critères de filtrage sont vérifiés avant de permettre l'accès**

# ACL RÉFLEXIVE | CARACTÉRISTIQUES D'UNE ENTRÉE TEMPORAIRE

- Toujours une entrée de type "**permit**"
- Spécifie le même **protocole** que le paquet original sortant
- Spécifie les mêmes **ports source** et **destination** (TCP et UDP) que le paquet original sortant, sauf que les numéros de ports sont croisés.
- Pour les protocoles qui n'ont pas de numéros de ports tel ICMP et IGMP d'autres critères sont spécifiés. Par exemple le type de paquet pour ICMP.
- Le trafic entrant est évalué avec l'entrée de la liste d'accès "Reflexive" jusqu'à ce que cette entrée expire. Si le paquet entrant correspond avec les critères de l'entrée celui-ci est acheminé dans le réseau.
- L'entrée expirera lorsque le dernier paquet de la session passera sur l'interface.
- Si aucun paquet pour cette session n'est détecté pendant une durée déterminée (timeout) l'entrée expire. Les entrées temporaires de la liste d'accès "Reflexive" sont effacées en fin de session.

# ACL RÉFLEXIVE | EXEMPLE

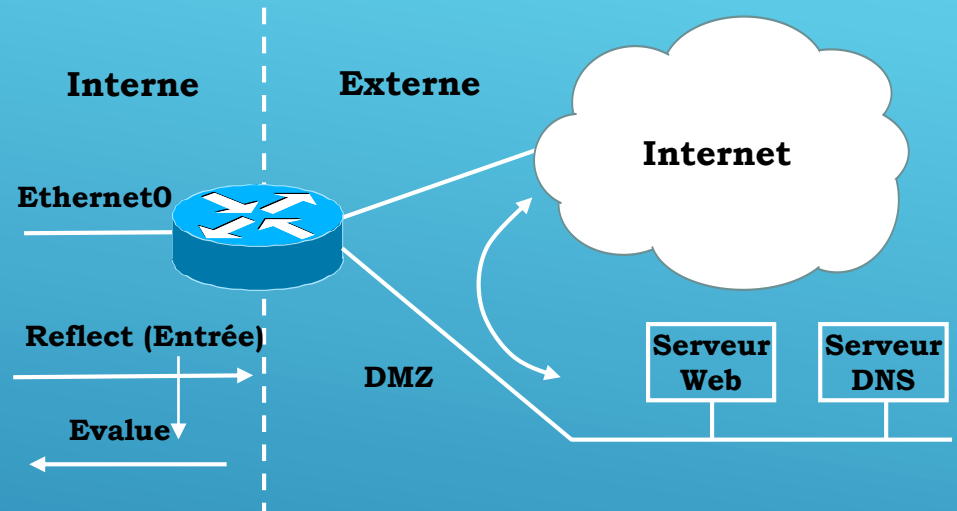
- Les listes d'accès "Reflexive" sont configurées sur l'interface externe Serial 1



- Ceci évite que du trafic IP entre par le routeur et dans le réseau interne sans qu'une session soit déjà établie depuis l'intérieur du réseau

# ACL RÉFLEXIVE | EXEMPLE

- Les listes d'accès "Reflexive" sont configurées sur l'interface interne Ethernet0



- Ceci autorise le trafic externe à accéder aux services de la DMZ, tels le DNS, mais évite que du trafic IP entre par le routeur et dans le réseau interne sans qu'une session soit déjà établie depuis l'intérieur du réseau.

# ACL RÉFLEXIVE | COMMANDES DE CONFIGURATION

1. Définition d'une liste d'accès étendue nommée qui sera appliquée à l'interface de sortie.

```
Router(config)#ip access-list extended extended-list-name
```

2. Configuration de la liste d'accès étendue nommée pour une entrée décrivant du trafic.

```
Router(config-ext-nacl)#permit ip-protocol any any reflect  
name [timeout seconds]
```

3. Application de la liste d'accès à l'interface de sortie

```
Router(config-if)#ip access-group extended-list-name out
```

# ACL RÉFLEXIVE | COMMANDES DE CONFIGURATION

4. Définition d'une liste d'accès étendue nommée qui va filtrer le trafic entrant.

```
Router(config)#ip access-list extended extended-list-name
```

5. Configuration cette liste d'accès étendue nommée pour évaluer le trafic entrant.

```
Router(config-ext-nacl)#evaluate name
```

# ACL RÉFLEXIVE | COMMANDES DE CONFIGURATION

6. Application de la liste d'accès étendue nommée à l'interface externe pour du trafic entrant.

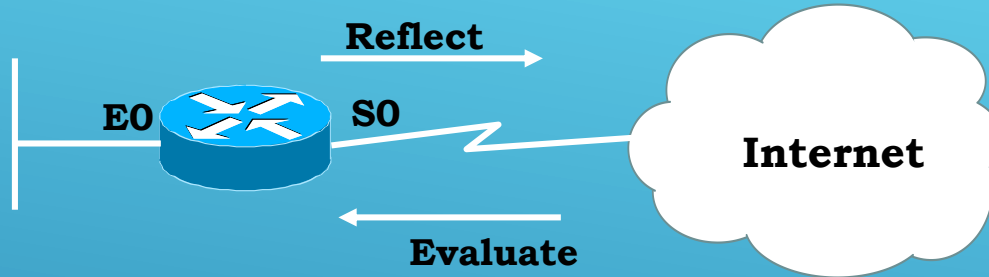
**Router(config-if)#ip access-group extended-list-name in**

7. (Option) Spécification d'un timeout global pour les entrées de la liste d'accès "Reflexive".

**Router(config)#ip reflexive-list timeout seconds**

- ▶ Les entrées des listes d'accès "réflexive" expirent si aucun paquet pour la session n'a été détecté pendant une durée prédéterminée.
- ▶ Le timeout global par défaut est de 300 sec.

# ACL RÉFLEXIVE | EXEMPLE DE CONFIGURATION



- **Création de la liste d'accès "reflexive"**
  - `RTA(config)#ip access-list extended SORTIE`
  - `RTA(config-ext-nacl)#permit ip any any reflect Traffic_permiss`
  - `RTA(config-ext-nacl)#exit`
  - `RTA(config)#interface serial0`
  - `RTA(config-if)#ip access-group SORTIE out`
- **Création d'une liste d'accès qui correspond au trafic entrant**
  - `RTA(config)#ip access-list extended ENTREE`
  - `RTA(config-ext-nacl)#evaluate Traffic_permiss`
  - `RTA(config-ext-nacl)#exit`
  - `RTA(config)#interface serial0`
  - `RTA(config-if)#ip access-group ENTREE in`
- **Si on le désire, un timeout global peut être fixé**
  - `RTA(config)#ip reflexive-list timeout 200`