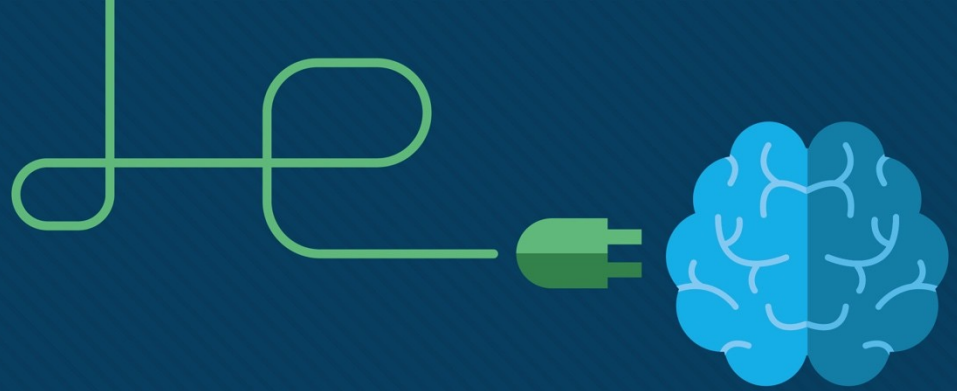


Module: Configuration de liste de contrôle d'accès pour IPv4

Réseau, Sécurité et
Automatisation D'entreprise
v7.0 (ENSA)

Objectifs du module

Module 5: Configuration de liste de contrôle d'accès pour IPv4

Objectif du module: Mettre en œuvre des listes de contrôle IPv4 pour filtrer le trafic et sécuriser l'accès des administrateurs.

Titre du Rubrique	Objectif du Rubrique
Configuration des listes de contrôle d'accès IPv4 standard	Configurer des listes de contrôle d'accès IPv4 standard pour filtrer le trafic afin de répondre aux besoins du réseau.
Modification des listes de contrôle d'accès IPv4	Utiliser les numéros de séquence pour modifier des listes de contrôle d'accès IPv4 standard.
Sécurisation des ports VTY à l'aide d'une liste de contrôle d'accès IPv4 standard	Configurer une liste de contrôle d'accès standard pour sécuriser l'accès VTY
Configuration de listes de contrôle d'accès IPv4 étendues	Configurer des listes de contrôle d'accès IPv4 étendues pour filtrer le trafic en fonction des besoins du réseau.

.1 Configurer les listes de contrôle d'accès IPv4 standard

Créer une ACL

Toutes les listes de contrôle d'accès (ACL) doivent être planifiées. Lors de la configuration d'une ACL complexe, il est suggéré de:

- Utiliser un éditeur de texte et écrire les spécificités de la stratégie à mettre en œuvre.
- Ajouter les commandes de configuration IOS pour accomplir ces tâches.
- Inclure des remarques pour documenter l'ACL.
- Copier et coller les commandes sur le périphérique.
- Tester toujours soigneusement une liste ACL pour vous assurer qu'elle applique correctement la stratégie souhaitée.

Syntaxe des listes de contrôle d'accès IPv4 standard numérotées

Pour créer une liste ACL standard numérotée, utilisez la commande **access-list** .

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

Paramètre	Description
access-list-number	La plage de nombres est de 1 à 99 ou de 1300 à 1999
deny	Refuse l'accès si les conditions sont respectées.
permit	Autorise l'accès si les conditions sont respectées.
remark *text*	(Facultatif) Ajoute une entrée de texte à des fins de documentation.
Source	Identifie l'adresse du réseau source ou de l'hôte à filtrer.
source-wildcard	(facultatif) Un masque générique de 32 bits qui est appliqué à la source
log	(Facultatif) Génère et envoie un message d'information lorsque l'ACE est apparié

Remarque: Utilisez la commande de configuration globale **no access-list** *access-list-number* pour supprimer une ACL standard numérotée.

Syntaxe des listes de contrôle d'accès IPv4 standard nommées

Pour créer une liste ACL standard numérotée, utilisez la commande **ip access-list standard** .

- Les noms des listes de contrôle d'accès doivent contenir uniquement des caractères alphanumériques, sont sensibles à la casse et doivent être uniques.
- Vous n'êtes pas obligés de mettre des majuscules aux noms des listes de contrôle d'accès. En revanche, si vous le faites, vous les verrez bien mieux en affichant la sortie de la comm

```
Router(config)# ip access-list standard access-list-name
R1(config)# ip access-list standard NO-ACCESS
R1(config-std-nacl)# ?
Standard Access List configuration commands:
  <1-2147483647>  Sequence Number
  default         Set a command to its defaults
  deny            Specify packets to reject
  exit            Exit from access-list configuration mode
  no              Negate a command or set its defaults
  permit          Specify packets to forward
  remark          Access list entry comment
R1(config-std-nacl)#
```

Appliquer une listes de contrôle d'accès IPv4 standard numérotées

Une fois qu'une ACL IPv4 standard est configurée, elle doit être liée à une interface ou à une fonctionnalité.

- La commande **ip access-group** est utilisée pour lier une ACL IPv4 standard numérotée ou nommée à une interface.
- Pour supprimer une ACL d'une interface, entrez d'abord la commande de configuration de l'interface **no ip access-group**

```
Router(config-if) # ip access-group {access-list-number | access-list-name} {in | out}
```

Exemple de liste de contrôle d’accès standard numérotées

L'exemple ACL autorise le trafic à partir de l'hôte 192.168.10.10 et de tous les hôtes sur l'interface de sortie réseau 192.168.20.0/24 série 0/1/0 sur le routeur R1.

```
R1(config)# access-list 10 remark ACE permits ONLY host 192.168.10.10 to the internet
R1(config)# access-list 10 permit host 192.168.10.10
R1(config)# do show access-lists
Standard IP access list 10
    10 permit 192.168.10.10
R1(config)#
```

```
R1(config)# access-list 10 remark ACE permits all host in LAN 2
R1(config)# access-list 10 permit 192.168.20.0 0.0.0.255
R1(config)# do show access-lists
Standard IP access list 10
    10 permit 192.168.10.10
    20 permit 192.168.20.0, wildcard bits 0.0.0.255
R1(config)#
```

```
R1(config)# interface Serial 0/1/0
R1(config-if)# ip access-group 10 out
R1(config-if)# end
R1#
```

Exemple de liste de contrôle d’accès standard numérotées (Suite)

- Utilisez la commande **show running-config** pour consulter la configuration.
- Utilisez la commande **show ip interface** pour vérifier que l'ACL est appliquée à la bonne interface.

```
R1# show run | section access-list
access-list 10 remark ACE permits host 192.168.10.10
access-list 10 permit 192.168.10.10
access-list 10 remark ACE permits all host in LAN 2
access-list 10 permit 192.168.20.0 0.0.0.255
R1#
```

```
R1# show ip int Serial 0/1/0 | include access list
  Outgoing Common access list is not set
  Outgoing access list is 10
  Inbound Common access list is not set
  Inbound  access list is not set
R1#
```

Exemple de liste de contrôle d’accès standard nommées

L'exemple ACL autorise le trafic à partir de l'hôte 192.168.10.10 et de tous les hôtes sur l'interface de sortie réseau 192.168.20.0/24 série 0/1/0 sur le routeur R1.

```
R1(config)# no access-list 10
R1(config)# ip access-list standard PERMIT-ACCESS
R1(config-std-nacl)# remark ACE permits host 192.168.10.10
R1(config-std-nacl)# permit host 192.168.10.10
R1(config-std-nacl)#
```

```
R1(config-std-nacl)# remark ACE permits host 192.168.10.10
R1(config-std-nacl)# permit host 192.168.10.10
R1(config-std-nacl)# remark ACE permits all hosts in LAN 2
R1(config-std-nacl)# permit 192.168.20.0 0.0.0.255
R1(config-std-nacl)# exit
R1(config)#
```

```
R1(config)# interface Serial 0/1/0
R1(config-if)# ip access-group PERMIT-ACCESS out
R1(config-if)# end
R1#
```

Exemple de liste de contrôle d’accès standard nommées (Suite)

- Utilisez la commande **show access-list** pour consulter la configuration.
- Utilisez la commande **show ip interface** pour vérifier que l'ACL est appliquée à la bonne interface.

```
R1# show access-lists
Standard IP access list PERMIT-ACCESS
    10 permit 192.168.10.10
    20 permit 192.168.20.0, wildcard bits 0.0.0.255
R1# show run | section ip access-list
ip access-list standard PERMIT-ACCESS
 remark ACE permits host 192.168.10.10
 permit 192.168.10.10
 remark ACE permits all hosts in LAN 2
 permit 192.168.20.0 0.0.0.255
R1#
```

```
R1# show ip int Serial 0/1/0 | include access list
  Outgoing Common access list is not set
  Outgoing access list is PERMIT-ACCESS
  Inbound Common access list is not set
  Inbound  access list is not set
R1#
```

Packet Tracer - Configurer les listes ACL IPv4 standard numérotées

Dans ce Packet Tracer, vous atteindrez les objectifs suivants:

- Planifier la mise en œuvre d'une liste de contrôle d'accès
- Configurer, appliquer et vérifier une liste de contrôle d'accès standard

Packet Tracer - Configurer les listes ACL IPv4 standard nommées

Dans ce Packet Tracer, vous atteindrez les objectifs suivants:

- Configurer, appliquer et vérifier une liste de contrôle d'accès standard
- Vérifier l'implémentation de la liste de contrôle d'accès.

.2 Modifier les listes de contrôle d'accès IPv4

Deux méthodes pour modifier une ACL

Une fois qu'une liste ACL est configurée, il peut être nécessaire de la modifier. Les ACL avec plusieurs ACE peuvent être complexes à configurer. Parfois, l'ACE configuré ne donne pas les comportements attendus.

Il existe deux méthodes à utiliser pour modifier une liste ACL:

- Utiliser un éditeur de texte
- Utiliser les numéros de séquence

Méthode éditeur de texte

Les ACL avec plusieurs ACE doivent être créées dans un éditeur de texte. Cela vous permet de planifier les ACE nécessaires, de créer l'ACL, puis de le coller sur l'interface du routeur. Il simplifie également les tâches de modification et de correction d'une ACL.

Pour corriger une erreur dans une liste ACL:

- Copiez l'ACL à partir de la configuration en cours d'exécution et collez-la dans l'éditeur de texte.
- Effectuez les modifications nécessaires.
- Supprimez la liste ACL configurée précédemment sur le routeur.
- Copiez et collez la liste ACL modifiée sur le routeur.

```
R1# show run | section access-list
access-list 1 deny   19.168.10.10
access-list 1 permit 192.168.10.0 0.0.0.255
R1#
```

```
R1(config)# no access-list 1
R1(config)#
R1(config)# access-list 1 deny 192.168.10.10
R1(config)# access-list 1 permit 192.168.10.0 0.0.0.255
R1(config)#
```

Méthode numéros de séquence

Un ACE ACL peut être supprimé ou ajouté à l'aide des numéros de séquence ACL.

- Utilisez la commande **ip access-list standard** pour modifier une ACL.
- Les instructions ne peuvent pas être remplacées par des instructions associées à un numéro de séquence existant déjà. l'instruction actuelle doit être supprimée d'abord avec la commande **no 10** . Ensuite, le bon ACE peut être ajouté en utilisant le numéro de séquence.

```
R1# show access-lists
Standard IP access list 1
    10 deny   19.168.10.10
    20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#
```

```
R1# conf t
R1(config)# ip access-list standard 1
R1(config-std-nacl)# no 10
R1(config-std-nacl)# 10 deny host 192.168.10.10
R1(config-std-nacl)# end
R1# show access-lists
Standard IP access list 1
    10 deny   192.168.10.10
    20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#
```

Modifier une ACL nommée Exemple

Les ACL nommées peuvent également utiliser des numéros de séquence pour supprimer et ajouter des ACE. Dans l'exemple, un ACE est ajouté pour refuser les hôtes 192.168.10.11.

```
R1# show access-lists
Standard IP access list NO-ACCESS
    10 deny   192.168.10.10
    20 permit 192.168.10.0, wildcard bits 0.0.0.255
```

```
R1# configure terminal
R1(config)# ip access-list standard NO-ACCESS
R1(config-std-nacl)# 15 deny 192.168.10.5
R1(config-std-nacl)# end
R1#
R1# show access-lists
Standard IP access list NO-ACCESS
    15 deny   192.168.10.5
    10 deny   192.168.10.10
    20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#
```

Statistiques des listes de contrôle d'accès

La commande **show access-lists** de l'exemple affiche des statistiques pour chaque instruction qui a été mise en correspondance.

- L'ACE de refus a été apparié 20 fois et le permis ACE a été apparié 64 fois.
- Notez que le refus implicite d'une instruction n'affiche aucune statistique. Pour suivre le nombre de paquets refusés implicitement appariés, vous devez configurer manuellement la commande **deny any** .
- Utilisez la commande **clear access-list counters** pour effacer les statistiques ACL.

```
R1# show access-lists
Standard IP access list NO-ACCESS
    10 deny   192.168.10.10  (20 matches)
    20 permit 192.168.10.0, wildcard bits 0.0.0.255  (64 matches)
R1# clear access-list counters NO-ACCESS
R1# show access-lists
Standard IP access list NO-ACCESS
    10 deny   192.168.10.10
    20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#
```

Packet Tracer - Configurer et modifier les listes ACL IPv4 standard

Dans ce Packet Tracer, vous atteindrez les objectifs suivants:

- Configurer les périphériques et vérifier la connectivité
- Configurer et vérifier les listes de contrôle d'accès numérotées et nommées standard
- Modifier une liste de contrôle d'accès standard

.3 Sécuriser les ports VTY à l'aide d'une liste de contrôle d'accès IPv4 standard

La commande access-class

Une liste ACL standard peut sécuriser l'accès administratif à distance à un périphérique à l'aide des lignes vty en implémentant les deux étapes suivantes:

- Créez une liste ACL pour identifier les hôtes administratifs qui doivent être autorisés à accéder à distance.
- Appliquez l'ACL au trafic entrant sur les lignes vty.

```
R1(config-line)# access-class {access-list-number | access-list-name} { in | out }
```

Exemple d'accès sécurisé aux VTY

Cet exemple montre comment configurer une liste ACL pour filtrer le trafic vty.

- Tout d'abord, une entrée de base de données locale pour une utilisateur **ADMIN** et mot de passe **class** est configurée.
- Les lignes vty sur R1 sont configurées pour utiliser la base de données locale pour l'authentification, autoriser le trafic SSH et utiliser l'ACL ADMIN-HOST pour restreindre le trafic.

```
R1(config)# username ADMIN secret class
R1(config)# ip access-list standard ADMIN-HOST
R1(config-std-nacl)# remark This ACL secures incoming vty lines
R1(config-std-nacl)# permit 192.168.10.10
R1(config-std-nacl)# deny any
R1(config-std-nacl)# exit
R1(config)# line vty 0 4
R1(config-line)# login local
R1(config-line)# transport input telnet
R1(config-line)# access-class ADMIN-HOST in
R1(config-line)# end
R1#
```

Vérifier la sécurité du port VTY

Une fois que la liste de contrôle d'accès aux lignes VTY est configurée, il est important de vérifier qu'elle fonctionne correctement.

Pour vérifier les statistiques ACL, exécutez la commande **show access-lists** .

- La correspondance dans la ligne d'autorisation de la sortie est le résultat d'une connexion SSH réussie par l'hôte avec l'adresse IP 192.168.10.10.
- La correspondance à l'instruction «deny» est due à l'échec de la de la tentative de créer une connexion SSH à partir d'un appareil sur un autre réseau.

```
R1#
Oct  9 15:11:19.544: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: admin] [Source: 192.168.10.10]
[localport: 23] at 15:11:19 UTC Wed Oct 9 2019
R1# show access-lists
Standard IP access list ADMIN-HOST
    10 permit 192.168.10.10  (2 matches)
    20 deny   any  (2 matches)
R1#
```

.4 Configurer les listes de contrôle d'accès IPv4 étendues

Les ACL étendues

Les ACL étendues offrent un plus grand degré de contrôle. Ils peuvent filtrer sur l'adresse source, l'adresse de destination, le protocole (c'est-à-dire IP, TCP, UDP, ICMP) et le numéro de port.

Les ACL étendues peuvent être créées comme suit:

- **ACL étendu numérotée** - Créé à l'aide de la commande de configuration globale **access-list** *access-list-number* .
- **ACL étendu nommée** - Créé à l'aide de la commande **ip access-list extended** *access-list-name* .

Protocoles et ports

Les ACL étendues peuvent filtrer sur protocoles et ports d'internet. Utiliser le **?** pour obtenir de l'aide lors de la saisie d'un ACE complexe . Les quatre protocoles mis en évidence sont les options les plus populaires.

Options de protocole

```
R1(config)# access-list 100 permit ?
  <0-255>        An IP protocol number
  ahp            Authentication Header Protocol
  dvmrp          dvmrp
  eigrp          Cisco's EIGRP routing protocol
  esp            Encapsulation Security Payload
  gre            Cisco's GRE tunneling
  icmp           Internet Control Message Protocol
  igmp           Internet Gateway Message Protocol
  ip             Any Internet Protocol
  ipinip         IP in IP tunneling
  nos            KA9Q NOS compatible IP over IP tunneling
  object-group   Service object group
  ospf           OSPF routing protocol
  pcp            Payload Compression Protocol
  pim            Protocol Independent Multicast
  tcp            Transmission Control Protocol
  udp            User Datagram Protocol
R1(config)# access-list 100 permit
```

Protocoles et ports (Suite)

La sélection d'un protocole influence les options de port. De nombreuses options de port TCP sont disponibles, comme indiqué dans la sortie.

```
R1(config)# access-list 100 permit tcp any any eq ?
  <0-65535>    Port number
  bgp          Border Gateway Protocol (179)
  chargen      Character generator (19)
  cmd          Remote commands (rcmd, 514)
  daytime      Daytime (13)
  discard      Discard (9)
  domain       Domain Name Service (53)
  echo         Echo (7)
  exec         Exec (rsh, 512)
  finger       Finger (79)
  ftp          File Transfer Protocol (21)
  ftp-data     FTP data connections (20)
  gopher       Gopher (70)
  hostname     NIC hostname server (101)
  ident        Ident Protocol (113)
  irc          Internet Relay Chat (194)
  klogin       Kerberos login (543)
  kshell       Kerberos shell (544)
  login        Login (rlogin, 513)
  lpd          Printer service (515)
  msrpc        MS Remote Procedure Call (135)
  nntp         Network News Transport Protocol (119)
  onep-plain   Onep Cleartext (15001)
  onep-tls     Onep TLS (15002)
  pim-auto-rp  PIM Auto-RP (496)
  pop2         Post Office Protocol v2 (109)
  pop3         Post Office Protocol v3 (110)
  smtp         Simple Mail Transport Protocol (25)
  sunrpc       Sun Remote Procedure Call (111)
  syslog       Syslog (514)
  tacacs       TAC Access Control System (49)
  talk         Talk (517)
  telnet       Telnet (23)
  time         Time (37)
  uucp         Unix-to-Unix Copy Program (540)
  whois        Nicname (43)
  www          World Wide Web (HTTP, 80)
```

Exemples de configuration de protocoles et de numéros de ports (Suite)

Les ACL étendues peuvent filtrer sur différentes options de numéro de port et de nom de port.

Cet exemple montre comment configurer une ACL 100 étendue pour filtrer le trafic HTTP. Le premier ACE utilise le nom de port **www** . Le deuxième ACE utilise le numéro de port **80**. Les deux ACE obtiennent exactement le même résultat.

```
R1(config)# access-list 100 permit tcp any any eq www
!or...
R1(config)# access-list 100 permit tcp any any eq 80
```

La configuration du numéro de port est requise lorsqu'aucun nom de protocole spécifique n'est répertorié tel que SSH (numéro de port 22) ou HTTPS (numéro de port 443), comme indiqué dans l'exemple suivant.

```
R1(config)# access-list 100 permit tcp any any eq 22
R1(config)# access-list 100 permit tcp any any eq 443
R1(config)#
```

Appliquer une ACL IPv4 étendue numérotée

Dans cet exemple, l'ACL permet à la fois le trafic HTTP et HTTPS à partir du réseau 192.168.10.0 d'accéder à n'importe quelle destination.

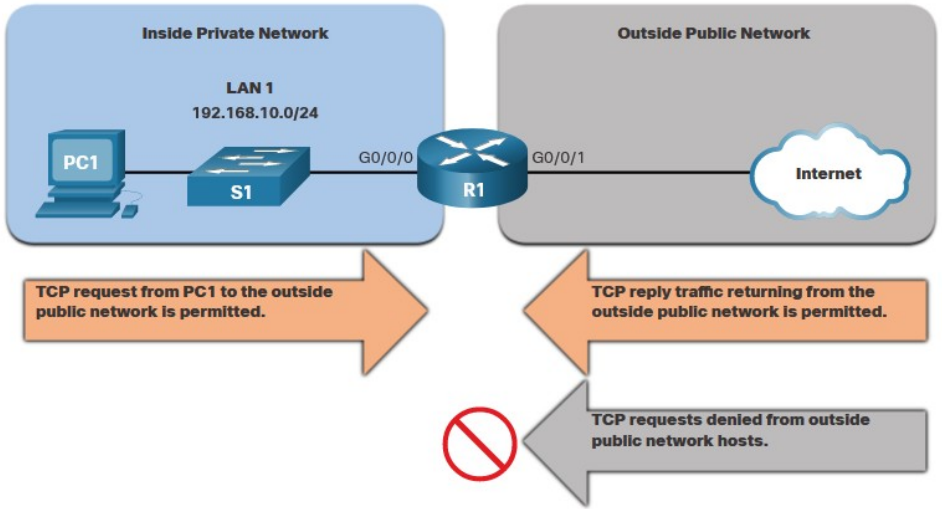
Les ACL étendues peuvent être appliquées à différents endroits. Cependant, elles sont couramment appliquées près de la source. Ici ACL 110 est appliquée en entrant sur l'interface R1 G0/0/0.

```
R1(config)# access-list 110 permit tcp 192.168.10.0 0.0.0.255 any eq www
R1(config)# access-list 110 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config)# interface g0/0/0
R1(config-if)# ip access-group 110 in
R1(config-if)# exit
R1(config)#
```

ACL étendue établie par TCP

TCP peut également effectuer des services de pare-feu avec état de base à l'aide du mot-clé TCP **established** .

- Le mot-clé **established** permet au trafic intérieur de quitter le réseau privé intérieur et permet au trafic de réponse de retourner d'entrer dans le réseau privé intérieur.
- Le trafic TCP généré par un hôte externe et la tentative de communication avec un hôte interne est refusé.



ACL étendue établie par TCP (Suite)

- ACL 120 est configurée pour autoriser uniquement le retour du trafic Web vers les hôtes internes. L'ACL est ensuite appliquée sortante sur l'interface R1 G0/0/0.
- La commande **show access-lists** indique que les hôtes internes accèdent aux ressources Web sécurisées à partir d'Internet.

Remarque: Il y a concordance si les bits ACK ou RST (réinitialisation) du segment TCP de retour sont définis, indiquant que le paquet appartient à une connexion existante.

```
R1(config)# access-list 120 permit tcp any 192.168.10.0 0.0.0.255 established
R1(config)# interface g0/0/0
R1(config-if)# ip access-group 120 out
R1(config-if)# end
R1# show access-lists
Extended IP access list 110
    10 permit tcp 192.168.10.0 0.0.0.255 any eq www
    20 permit tcp 192.168.10.0 0.0.0.255 any eq 443 (657 matches)
Extended IP access list 120
    10 permit tcp any 192.168.10.0 0.0.0.255 established (1166 matches)
R1#
```

Syntaxe ACL étendue IPv4 nommée

Si vous attribuez un nom à une liste de contrôle d'accès, il vous sera plus facile d'en comprendre la fonction. Pour créer une liste ACL étendue nommée, utilisez la commande de configuration **ip access-list extended** .

Dans l'exemple, une liste ACL étendue nommée NO-FTP-ACCESS est créée et l'invite est modifiée en mode de configuration ACL étendue nommée. Les instructions ACE sont entrées dans le mode de sous-configuration ACL étendu nommé.

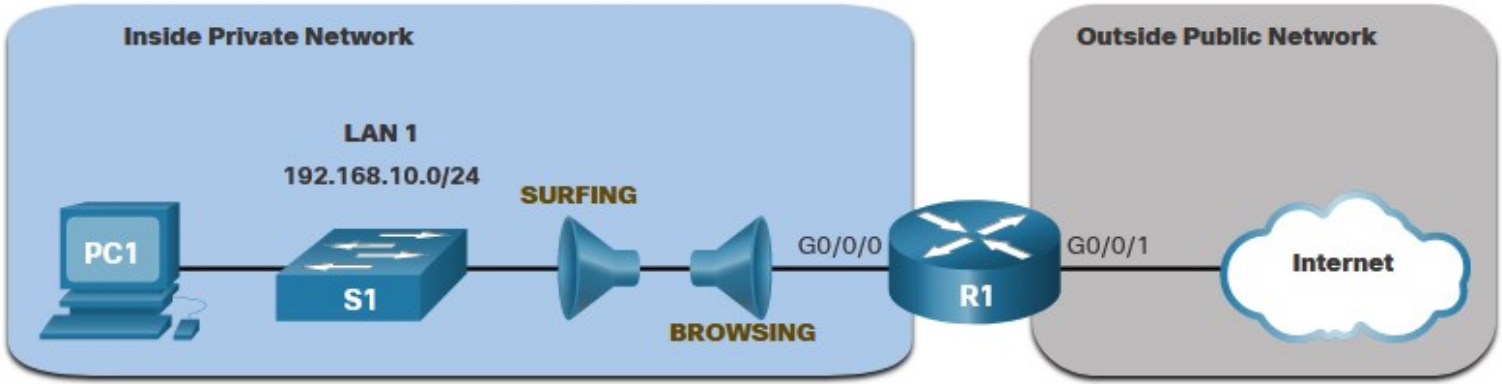
```
Router(config)# ip access-list extended access-list-name
```

```
R1(config)# ip access-list extended NO-FTP-ACCESS
R1(config-ext-nacl)#
```

Exemple d'ACL étendue IPv4 nommée

La topologie ci-dessous permet de démontrer la configuration et l'application de deux ACL étendues IPv4 nommées à une interface:

- SURFING - Cela permettra à l'intérieur du trafic HTTP et HTTPS de quitter l'internet.
- **BROWSING** - Cela permettra uniquement de renvoyer le trafic Web aux hôtes internes alors que tout autre trafic sortant de l'interface R1 G0/0/0 est implicitement refusé.



Exemple de liste ACL étendue IPv4 nommée (suite)

- L'ACL SURFING permet au trafic HTTP et HTTPS des utilisateurs internes de quitter l'interface G0/0/1 connectée à l'internet. Le trafic Web revenant de l'internet est autorisé à revenir sur le réseau privé interne par l'ACL BROWSING.
- La liste ACL SURVING est appliquée entrante et la liste ACL BROWSING est appliquée sortante sur l'interface R1 G0/0/0.

```
R1(config)# ip access-list extended SURFING
R1(config-ext-nacl)# Remark Permits inside HTTP and HTTPS traffic
R1(config-ext-nacl)# permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config-ext-nacl)# permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config-ext-nacl)# exit
R1(config)#
R1(config)# ip access-list extended BROWSING
R1(config-ext-nacl)# Remark Only permit returning HTTP and HTTPS traffic
R1(config-ext-nacl)# permit tcp any 192.168.10.0 0.0.0.255 established
R1(config-ext-nacl)# exit
R1(config)# interface g0/0/0
R1(config-if)# ip access-group SURFING in
R1(config-if)# ip access-group BROWSING out
R1(config-if)# end
R1# show access-lists
Extended IP access list SURFING
    10 permit tcp 192.168.10.0 0.0.0.255 any eq www
    20 permit tcp 192.168.10.0 0.0.0.255 any eq 443 (124 matches)
Extended IP access list BROWSING
    10 permit tcp any 192.168.10.0 0.0.0.255 established (369 matches)
R1#
```

Exemple de liste ACL étendue IPv4 nommée (suite)

Pour vérifier les statistiques ACL, exécutez la commande show access-lists . Notez que les compteurs HTTPS sécurisés par permis (c.-à-d., eq 443) dans l'ACL SURFING et les compteurs de retour établis dans l'ACL BROWSING ont augmenté.

```
R1# show access-lists
Extended IP access list BROWSING
    10 permit tcp any 192.168.10.0 0.0.0.255 established
Extended IP access list SURFING
    10 permit tcp 19.168.10.0 0.0.0.255 any eq www
    20 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1#
```

Modifier les ACL étendues

Une liste ACL étendue peut être modifiée à l'aide d'un éditeur de texte lorsque de nombreuses modifications sont nécessaires. Ou, si l'édition s'applique à un ou deux ACE, les numéros de séquence peuvent être utilisés.

Exemple:

- Le numéro de séquence ACE 10 dans l'ACL SURFING a une adresse de réseau IP source incorrecte.

```
R1# show access-lists
Extended IP access list BROWSING
    10 permit tcp any 192.168.10.0 0.0.0.255 established
Extended IP access list SURFING
    10 permit tcp 19.168.10.0 0.0.0.255 any eq www
    20 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1#
```

Modifier les ACL étendues (Suite)

- Pour corriger cette erreur, l'instruction d'origine est supprimée avec la commande **no** *sequence_#* et l'instruction corrigée est ajoutée en remplacement de l'instruction d'origine.
- La sortie de la commande **show access-lists** vérifie le changement de configuration.

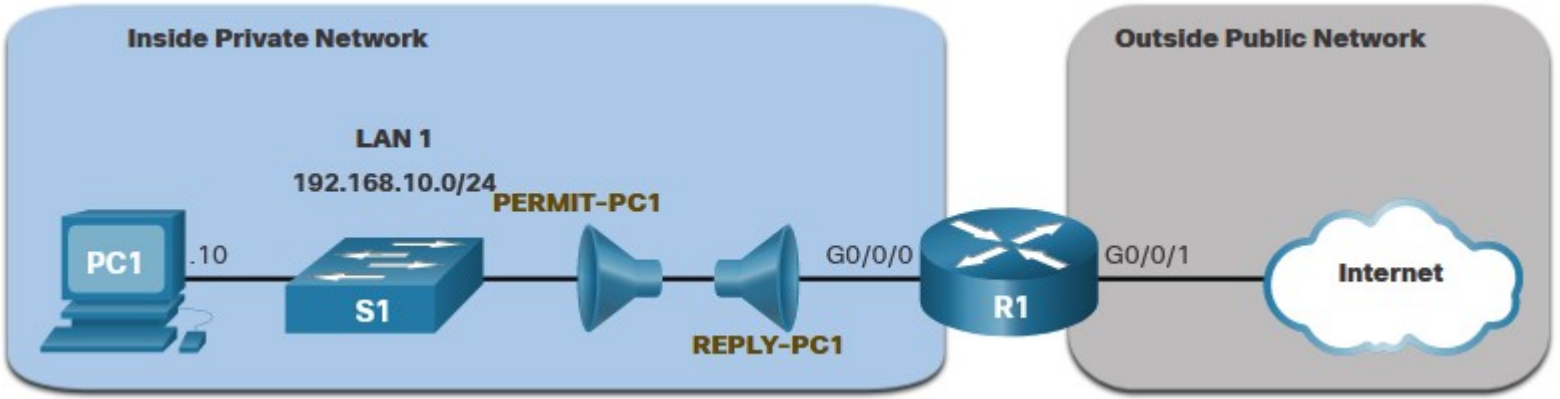
```
R1# configure terminal
R1(config)# ip access-list extended SURFING
R1(config-ext-nacl)# no 10
R1(config-ext-nacl)# 10 permit tcp 192.168.10.0 0.0.0.255 any eq www
R1(config-ext-nacl)# end
```

```
R1# show access-lists
Extended IP access list BROWSING
    10 permit tcp any 192.168.10.0 0.0.0.255 established
Extended IP access list SURFING
    10 permit tcp 192.168.10.0 0.0.0.255 any eq www
    20 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1#
```

Autre exemple d'ACL étendue IPv4 nommée

Deux ACL étendues nommées seront créées:

- **PERMIT-PC1** - Cela permettra uniquement l'accès PC1 TCP à l'internet et refusera tous les autres hôtes du réseau privé.
- **REPLY-PC1** - Cela permettra uniquement le retour du trafic TCP spécifié à PC1 refuser implicitement tout autre trafic.



Autre exemple d'ACL étendue IPv4 nommée (Suite)

- L'ACL **PERMIT-PC1** autorise PC1 (192.168.10.10) l'accès TCP au trafic FTP, SSH, Telnet, DNS, HTTP et HTTPS.
- La liste ACL **REPLY-PC1** permettra le retour du trafic vers PC1.
- La liste ACL **PERMIT-PC1** est appliquée en entrée et la liste ACL **REPLY-PC1** est appliquée en sortie sur l'interface R1 G0/0/0.

```
R1(config)# ip access-list extended PERMIT-PC1
R1(config-ext-nacl)# Remark Permit PC1 TCP access to internet
R1(config-ext-nacl)# permit tcp host 192.168.10.10 any eq 20
R1(config-ext-nacl)# permit tcp host 192.168.10.10 any eq 21
R1(config-ext-nacl)# permit tcp host 192.168.10.10 any eq 22
R1(config-ext-nacl)# permit tcp host 192.168.10.10 any eq 23
R1(config-ext-nacl)# permit tcp host 192.168.10.10 any eq 53
R1(config-ext-nacl)# permit tcp host 192.168.10.10 any eq 80
R1(config-ext-nacl)# permit tcp host 192.168.10.10 any eq 443
R1(config-ext-nacl)# deny ip 192.168.10.0 0.0.0.255 any
R1(config-ext-nacl)# exit
R1(config)#
R1(config)# ip access-list extended REPLY-PC1
R1(config-ext-nacl)# Remark Only permit returning traffic to PC1
R1(config-ext-nacl)# permit tcp any host 192.168.10.10 established
R1(config-ext-nacl)# exit
R1(config)# interface g0/0/0
R1(config-if)# ip access-group PERMIT-PC1 in
R1(config-if)# ip access-group REPLY-PC1 out
R1(config-if)# end
R1#
```

Vérifier les listes de contrôle d'accès étendues

La commande **show ip interface** permet de vérifier la liste de contrôle d'accès sur l'interface et la direction dans laquelle elle a été appliquée.

```
R1# show ip interface g0/0/0
GigabitEthernet0/0/0 is up, line protocol is up (connected)
  Internet address is 192.168.10.1/24
  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 REPLY-PC1
  Inbound  access list is PERMIT-PC1
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is disabled
  IP fast switching on the same interface is disabled
  IP Flow switching is disabled
  IP Fast switching turbo vector
  IP multicast fast switching is disabled
  IP multicast distributed fast switching is disabled
  Router Discovery is disabled
R1#
R1# show ip interface g0/0/0 | include access list
Outgoing access list is REPLY-PC1
Inbound access list is PERMIT-PC1
R1#
```

Vérifier les listes de contrôle d'accès étendues (Suite)

La commande **show access-lists** peut être utilisée pour confirmer que les ACL fonctionnent comme prévu. La commande affiche les compteurs statistiques qui augmentent chaque fois qu'un ACE est apparié.

Remarque: Le trafic doit être généré pour vérifier le fonctionnement de l'ACL.

```
R1# show access-lists
Extended IP access list PERMIT-PC1
10 permit tcp host 192.168.10.10 any eq 20
20 permit tcp host 192.168.10.10 any eq ftp
30 permit tcp host 192.168.10.10 any eq 22
40 permit tcp host 192.168.10.10 any eq telnet
50 permit tcp host 192.168.10.10 any eq domain
60 permit tcp host 192.168.10.10 any eq www
70 permit tcp host 192.168.10.10 any eq 443
80 deny ip 192.168.10.0 0.0.0.255 any
Extended IP access list REPLY-PC1
10 permit tcp any host 192.168.10.10 established
R1#
```

Vérifier les listes de contrôle d'accès étendues (Suite)

La commande **show running-config** peut être utilisée pour valider ce qui a été configuré. La commande affiche également les remarques configurées.

```
R1# show running-config | begin ip access-list
ip access-list extended PERMIT-PC1
remark Permit PC1 TCP access to internet
permit tcp host 192.168.10.10 any eq 20
permit tcp host 192.168.10.10 any eq ftp
permit tcp host 192.168.10.10 any eq 22
permit tcp host 192.168.10.10 any eq telnet
permit tcp host 192.168.10.10 any eq domain
permit tcp host 192.168.10.10 any eq www
permit tcp host 192.168.10.10 any eq 443
deny ip 192.168.10.0 0.0.0.255 any
ip access-list extended REPLY-PC1
remark Only permit returning traffic to PC1
permit tcp any host 192.168.10.10 established
!
```


CISCO