APS
Prístupové zoznamy

Vytvorené v rámci projektu KEGA 026TUKE-4/2021

Katedra počítačov a informatiky
Fakulta elektrotechniky a informatiky
Technická univerzita v Košiciach

Štandardné ACL a ich konfigurácia

Sériové a paralelné rozhrania

- ACL obsahuje zoznam postupných záznamov o **povolení** alebo **odmietnutí**, známe sú aj ako ACE resp. angl. *access control entries*.
- IPv4 ACE zahŕňajú použitie wildcard masiek, ide o reťazec 32bin číslic.

Príklad 1

	Decimal	Binary
IP Address	192.168.1.1	11000000.10101000.00000001.00000001
Wildcard Mask	0.0.0.0	00000000.00000000.00000000.00000000
Result	192.168.1.1	11000000.10101000.00000001.00000001

Príklad 2

	Decimal	Binary
IP Address	192.168.1.1	11000000.10101000.00000001.00000001
Wildcard Mask	255.255.255.255	11111111.11111111.11111111.11111111
Result	0.0.0.0	00000000.00000000.00000000.00000000

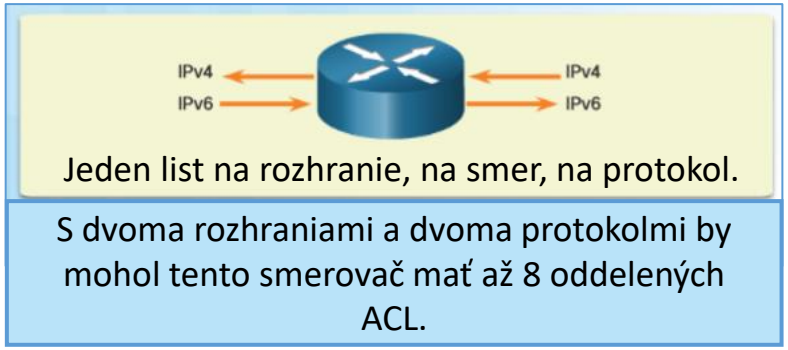
Príklad 3

	Decimal	Binary
IP Address	192.168.1.1	11000000.10101000.00000001.00000001
Wildcard Mask	0.0.0.255	00000000.00000000.00000000.11111111
Result	192.168.1.0	11000000.10101000.00000001.00000000

Fungovanie ACL

Konfigurovať je možné:

- **Jeden ACL na jeden protokol** – Pre riadenie toku na rozhraní musí byť ACL definované pre každý protokol povolený na rozhraní.
- **Jeden ACL na jeden smer** - ACL riadia prevádzku na rozhraní v jednom smere. Na kontrolu prichádzajúcej a odchádzajúcej prevádzky sa musia vytvoriť dve samostatné ACL zoznamy.
- **Jeden ACL na jedno rozhranie** – ACL riadia prevádzku pre rozhranie, napr. GigabitEthernet 0/0.



Fungovanie ACL (pokr.)

Rozšírené ACL môžu filtrovať prevádzku preskúmaním čísel TCP portov.

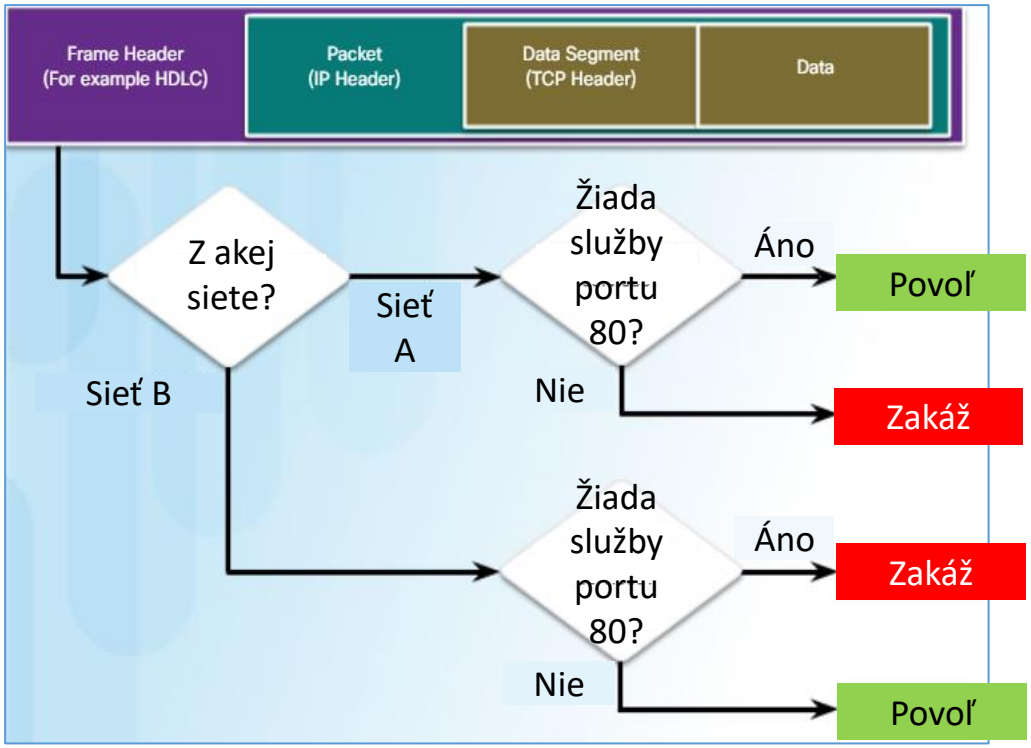
Časté čísla TCP a UDP portov zahŕňajú:

Číslo portu	Protokol	Aplikácia	Skratka
20	TCP	File Transfer Protocol (data)	FTP
21	TCP	File Transfer Protocol (control)	FTP
22	TCP	Secure Shell	SSH
23	TCP	Telnet	–
25	TCP	Simple Mail Transfer Protocol	SMTP
53	UDP, TCP	Domain Name Service	DNS
67	UDP	Dynamic Host Configuration Protocol (server)	DHCP
68	UDP	Dynamic Host Configuration Protocol (client)	DHCP
69	UDP	Trivial File Transfer Protocol	TFTP
80	TCP	Hypertext Transfer Protocol	HTTP
110	TCP	Post Office Protocol version 3	POP3
143	TCP	Internet Message Access Protocol	IMAP
161	UDP	Simple Network Management Protocol	SNMP
443	TCP	Hypertext Transfer Protocol Secure	HTTPS

Fungovanie ACL (príklad)

ACL bol nakonfigurovaný tak, aby:

- Umožnil prístup na web pre používateľov zo siete A, ale odoprel používateľom siete A všetky ostatné služby.
- Zakázal HTTP prístup k používateľom zo siete B, ale umožnil používateľom siete B mať akýkoľvek iný prístup.



Typy ACL pre IPv4

Štandardné ACL filtrujú pakety len podľa zdrojovej adresy.

```
access-list 10 permit 192.168.30.0 0.0.0.255
```

Rozšírené ACL filtrujú pakety podľa:

- Typu protokolu / čísla protokolu (napr. IP, ICMP, UDP, TCP, ...).
- Zdrojovej a cieľovej IP adresy.
- Zdrojových a cieľových TCP a UDP portov.

```
access-list 103 permit tcp 192.168.30.0 0.0.0.255 any eq 80
```

Typy ACL pre IPv4 (pokr.)

Štandardné a rozšírené ACL je možné vytvoriť buď pomocou **čísla** alebo **mena**.

Číslované ACL	Pomenované ACL
Priraďte číslo podľa protokolu aký chcete filtrovať: • (1 do 99) a (1300 do 1999): Štandardné ACL • (100 do 199) a (2000 do 2699): Rozšírené ACL	Zadajte meno pre identifikáciu ACL: • Pomenovanie môže zahŕňať alfanumerické znaky. • Odporúča sa používať kapitálky. • Nie je možné použiť medzery a interpunkciu. • Záznamy je možné v rámci ACL pridávať a odoberať.

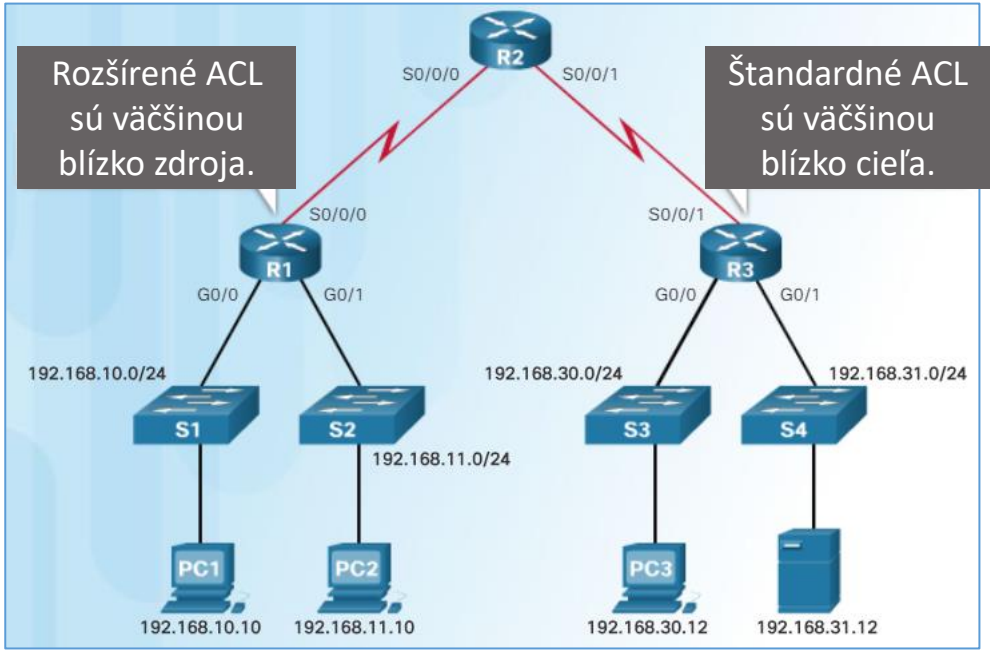
Rozšírené a štandardné ACL

Rozšírené ALC by mali byť umiestnené čo najbližšie k zdroju filtrovanej prevádzky.

Zakázaná prevádzka bližšie pri zdrojovej sieti neprechádza sieťovou infraštruktúrou.

Štandardné ACL by mali byť umiestnené čo najbližšie k cieľu.

Ak by bol štandardný ACL umiestnený pri zdroji prevádzky, filtrovala by sa prevádzka na základe danej zdrojovej adresy bez ohľadu na to, kde je určená.

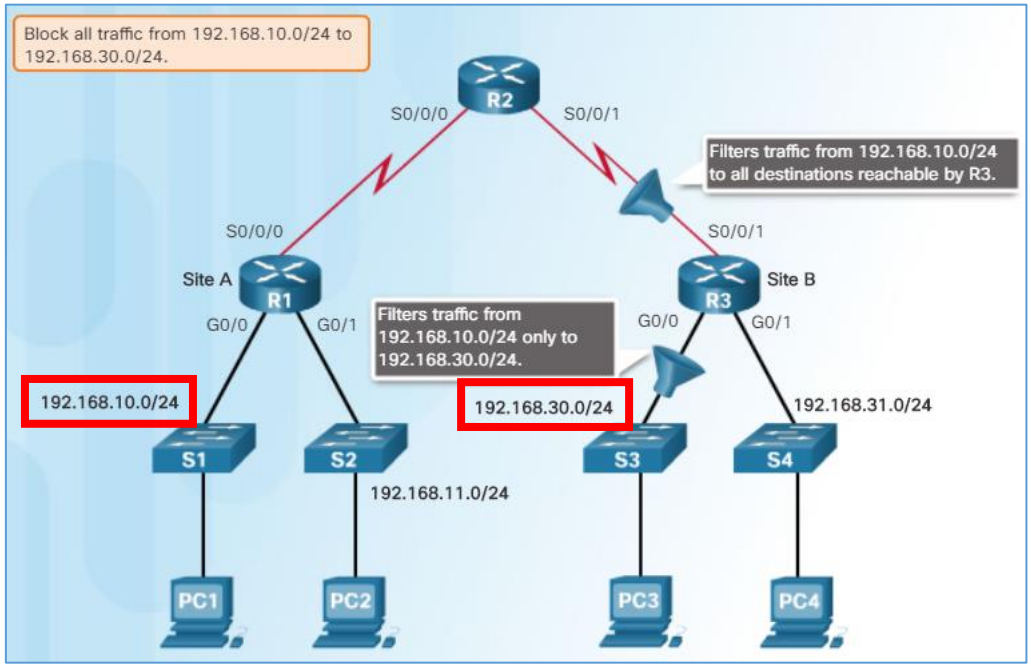


Rozšírené a štandardné ACL (pokr.)

Štandardný ACL bude nakonfigurovaný, aby zablokoval všetko zo siete **192.168.10.0/24** do siete **192.168.30.0/24**.

- Štandardný ACL by mal byť aplikovaný čo najbližšie k cieľu, a preto by mať byť nasadený na R3 výstup z rozhrania G0/0.

  Aplikovanie na R3 vstup do rozhrania S0/0/1 by zabránilo dosiahnuť sieť 192.168.31.0/24 a preto by sa tam nemal nasadiť.

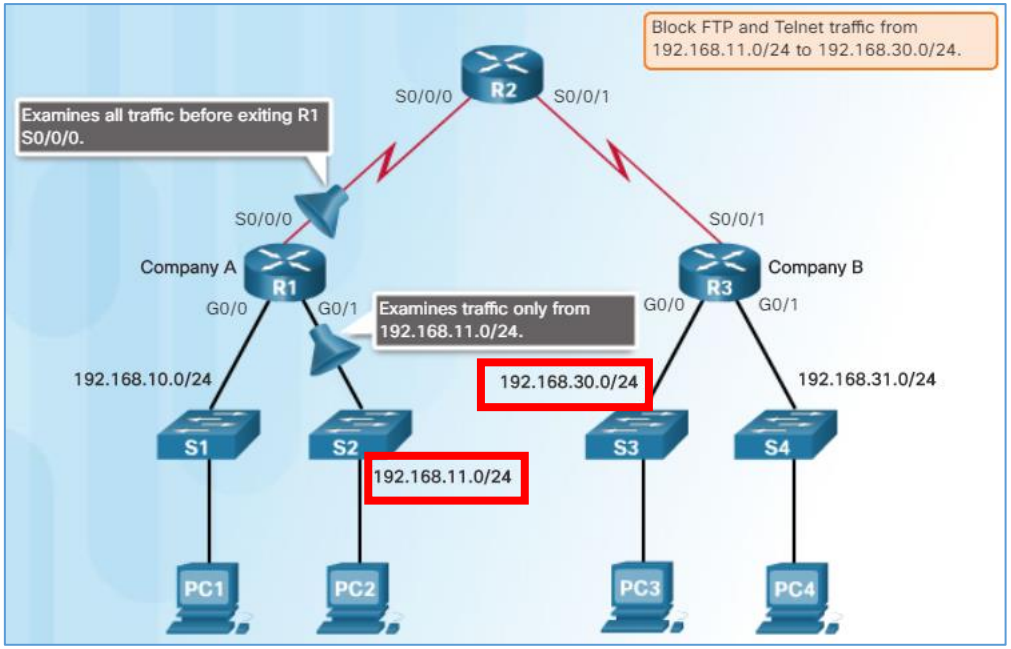


Rozšírené a štandardné ACL (pokr.)

Rozšírený ACL bude nakonfigurovaný, aby zablokoval FTP a Telnet zo siete **192.168.11.0/24** do siete **192.168.30.0/24**.

- Rozšírený ACL by mal byť aplikovaný čo najbližšie k zdroju, a preto by mať byť nasadený na R1 vstup do rozhrania G0/1.

  Aplikovanie na R1 výstup z rozhrania S0/0/1 by zabránilo dosiahnuť sieť 192.168.31.0/24 a taktiež by sa zbytočne spracovávali pakety z 192.168.10.0/24.



Nasadenie štandardného ACL

Úplná syntax štandardného ACL príkazu je nasledujúca:

access-list *ACL-#* {**deny** | **permit** | **remark**} *source* [*source-wildcard*][**log**]

Napríklad:

- Povoliť všetky IP adresy v sieti 192.168.10.0/24
- Použiť príkaz **no access-list 10** na odstránenie ACL.
- Použiť príkaz **remark** pre dokumentáciu ACL.

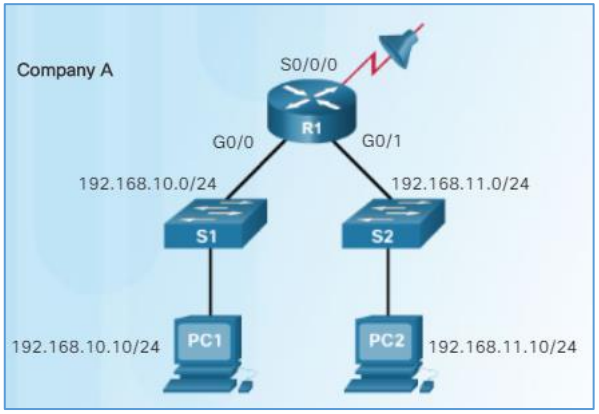
```
R1(config)# access-list 10 permit 192.168.10.0 0.0.0.255
R1(config)# exit
R1# show access-lists
Standard IP access list 10
    10 permit 192.168.10.0, wildcard bits 0.0.0.255
R1# conf t
Enter configuration commands, one per line. End with
CNTL/Z.
R1(config)# no access-list 10
R1(config)# exit
R1# show access-lists
R1#
```

```
R1(config)# access-list 10 remark Permit hosts from the 192.168.10.0 LAN
R1(config)# access-list 10 permit 192.168.10.0 0.0.0.255
R1(config)# exit
R1# show running-config | include access-list 10
access-list 10 remark Permit hosts from the 192.168.10.0 LAN
access-list 10 permit 192.168.10.0 0.0.0.255
R1#
```

Nasadenie štandardného ACL (pokr.)

IPv4 ACL sa prepája s rozhraním pomocou nasledujúceho príkazu:

ip access-group {*ACL-#* | *access-list-name*} {**in** | **out**}



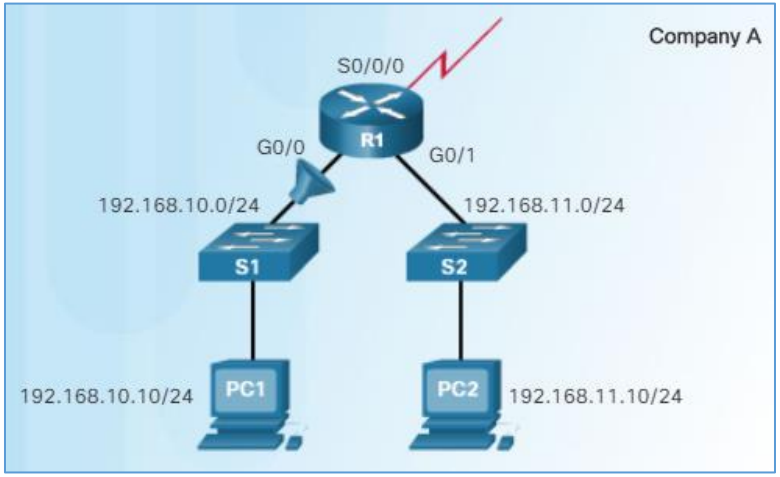

```
R1(config)# access-list 1 permit 192.168.10.0 0.0.0.255
R1(config)# interface s0/0/0
R1(config-if)# ip access-group 1 out
```

Pozn.: Pre odstránenie ACL z rozhrania, najprv zadajte na rozhraní príkaz **no ip access-group** a potom zadajte príkaz v globálnom móde **no access-list** pre odstránenie celého ACL.

Implementácia štandardného IPv4 ACL

Pre vytvorenie štandardného pomenovaného ACL.

- Použite príkaz
  `ip access-list standard` *`[meno]`*.
  - Názvy sú alfanumerické, rozlišujú sa veľké a malé písmená a musia byť jedinečné.
  - Príkazom sa dostanete do konfiguračného režimu ACL.
- Použite **`permit`**, **`deny`** alebo **`remark`**.
- Aplikujte ACL na rozhranie pomocou príkazu
  `ip access-group` *`[meno]`*.




```
R1(config)# ip access-list standard NO_ACCESS
R1(config-std-nacl)# deny host 192.168.11.10
R1(config-std-nacl)# permit any
R1(config-std-nacl)# exit
R1(config)# interface g0/0
R1(config-if)# ip access-group NO_ACCESS out
```

Implementácia štandardného IPv4 ACL (pokr.)

Príkazom **`show ip interface`** overte ACL na rozhraní.

Výstup zahŕňa číslo alebo názov ACL a smer, v ktorom bol použitý.

Príkaz **`show access-lists`** [*ACL-# | access-list-name*] zobrazí obsah štandardného ACL.

Štandardné ACL spracováva záznamy v poradí, v akom boli zadané.

```
R1# show ip interface s0/0/0
Serial0/0/0 is up, line protocol is up
Internet address is 10.1.1.1/30
<output omitted>
  Outgoing access list is 1
  Inbound access list is not set
<output omitted>
R1# show ip interface g0/0
GigabitEthernet0/0 is up, line protocol is up
  Internet address is 192.168.10.1/24
<output omitted>
  Outgoing access list is NO_ACCESS
  Inbound access list is not set
<output omitted>
```

```
R1# show access-lists
Standard IP access list 1
  10 deny 192.168.10.10
  20 permit 192.168.0.0, wildcard bits 0.0.255.255
Standard IP access list NO_ACCESS
  15 deny 192.168.11.11
  10 deny 192.168.11.10
  20 permit 192.168.11.0, wildcard bits 0.0.0.255
R1#
```

Rozšírené ACL

Štruktúra rozšírených IPv4 ACL

Rozšírené IPv4 ACL poskytujú presnejšie filtrovanie.

- Rozšírené zoznamy ACL sú číslované od 100 do 199 a od 2000 do 2699, čo predstavuje celkovo 799 možných rozšírených číslovaných ACL.
- Rozšírené ACL môžu byť taktiež pomenované.
- Rozšírené ACL sa používajú častejšie ako štandardné ACL, pretože poskytujú väčšiu mieru kontroly.

Štruktúra rozšírených IPv4 ACL (pokr.)

- Rozšírené ACL môžu filtrovať protokol a číslo portu.
- Aplikáciu možno špecifikovať konfiguráciou, buď:

Čísla portu

```
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq 23
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq 21
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq 20
```

Názvu známeho portu

```
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq telnet
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq ftp
access-list 114 permit tcp 192.168.20.0 0.0.0.255 any eq ftp-data
```

Pozn.: Použite otazník (?) na zobrazenie dostupných názvov portov.

Napr. **`access-list 101 permit tcp any any eq ?`**

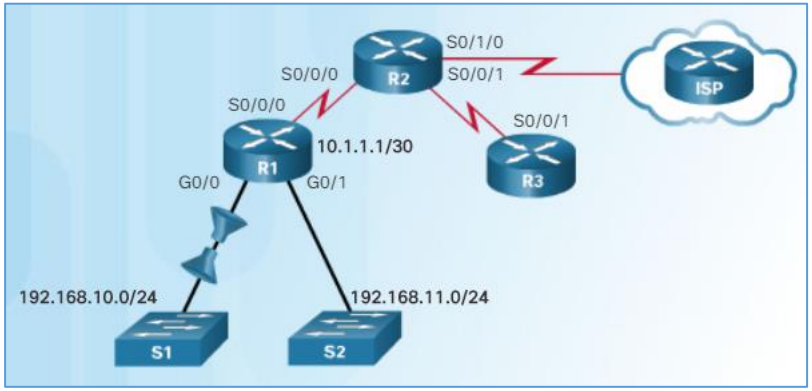
Štruktúra rozšírených IPv4 ACL (pokr.)

- Úplná syntax rozšíreného ACL príkazu je nasledujúca:

```
access-list ACL-# {deny | permit | remark} protocol {source source-
wildcard][operator [port-number | port-name]] {destination destination-
wildcard][operator [port-number | port-name]]
```

Napr.:

- ACL 103 povoľuje požiadavky na porty 80 a 443.
- ACL 104 povoľuje odpovedať na HTTP a HTTPS.
- Parameter **`established`** umožňuje odpovedať iba na prevádzku, ktorá pochádza zo siete 192.168.10.0/24 a vracia do tejto siete.

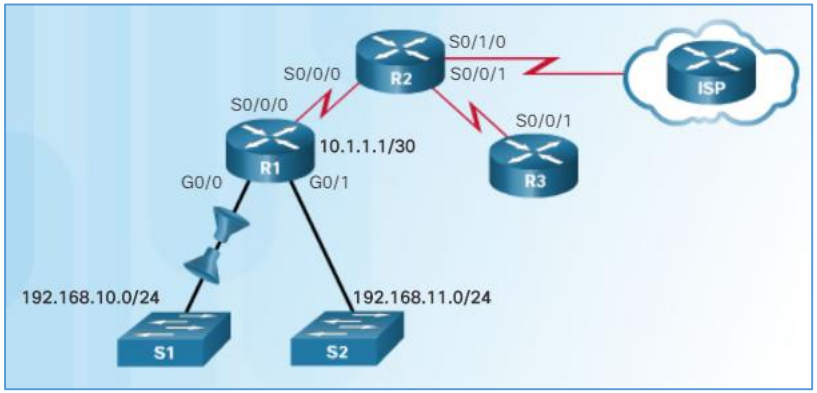



```
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config)# access-list 104 permit tcp any 192.168.10.0 0.0.0.255 established
```

Konfigurácia rozšírených ACL

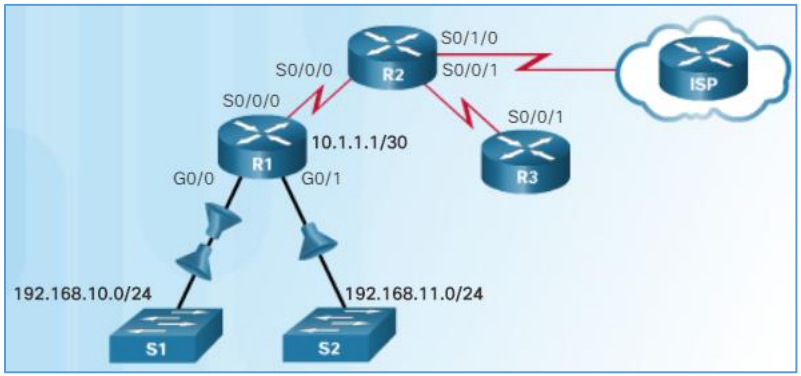
Použitie rozšírených ACL je podobné ako štandardné ACL, s výnimkou toho, že by sa mali aplikovať čo najbližšie k zdroju.

```
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config)# access-list 104 permit tcp any 192.168.10.0 0.0.0.255 established
R1(config)# interface g0/0
R1(config-if)# ip access-group 103 in
R1(config-if)# ip access-group 104 out
```



Konfigurácia rozšírených ACL (pokr.)

- V tomto príklade ide o FTP prenos z podsiete 192.168.11.0 do podsiete 192.168.10.0. FTP prenos je zakázaný, ale iná prevádzka je povolená.
- FTP využíva dve čísla portov (TCP port 20 a 21), preto sú v ACL potrebné dva záznamy.
- Príklad požitia známych názvov portov je **`ftp`** a **`ftp-data`**.
- Bez aspoň jedného **`permit`** záznamu v ACL, by bola celá komunikácia na rozhraní zakázaná.
- ACL je aplikovaná na vstupe do R1 rozhranie G0/1.

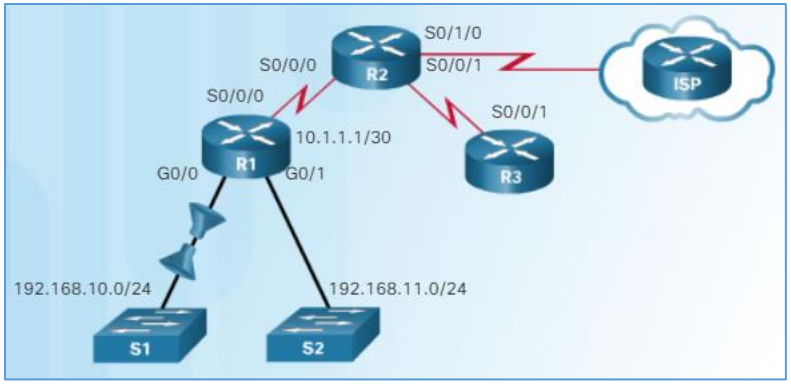



```
R1(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255
192.168.10.0 0.0.0.255 eq ftp
R1(config)# access-list 101 deny tcp 192.168.11.0 0.0.0.255
192.168.10.0 0.0.0.255 eq ftp-data
R1(config)# access-list 101 permit ip any any
R1(config)# interface g0/1
R1(config-if)#ip access-group 101 in
```

Konfigurácia rozšírených ACL (pokr.)

- Pomenované rozšírené ACL sa vytvárajú rovnakým spôsobom ako štandardné ACL.
- V tomto príklade sú vytvorené dva pomenované ACL:
  - SURFING umožňuje používateľom v sieti 192.168.10.0/24 ukončiť prístup na porty 80 a 443.
  - BROWSING umožňuje návrat HTTP a HTTPs komunikácie.

```
R1(config)# ip access-list extended SURFING
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)# ip access-list extended BROWSING
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
R1(config-if)# ip access-group SURFING in
R1(config-if)# ip access-group BROWSING out
```



Konfigurácia rozšírených ACL (pokr.)

Príkazy **`show ip interface`** a **`show access-lists`** je možné použiť na overenie obsahu rozšírených ACL.

- Výstup a poradové čísla zobrazené vo výstupe príkazu **`show access-lists`** definujú poradie, v ktorom boli zadané príkazy.
  - Na rozdiel od štandardných ACL, rozšírené ACL neimplementujú rovnakú internú logiku a funkciu hashovania.
  - Hostové záznamy nie sú automaticky dané pred záznamy siete.

```
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#
```

- Príkaz **`show ip interface`** sa používa na overenie ACL na rozhraní a smeru, v ktorom bol použitý.
  - Výstup tohto príkazu obsahuje číslo alebo názov ACL a smer, v ktorom bol použitý.

```
R1# show ip interface g0/0
GigabitEthernet0/0 is up, line protocol is up
  Internet address is 192.168.10.1/24
<output omitted>
  Outgoing access list is BROWSING
  Inbound access list is SURFING
<output omitted>
```

Úprava rozšírených ACL

Rozšírený ACL je možné upraviť 2. spôsobmi:

1. Spôsob: Textový editor

- ACL sa skopíruje a vloží do text. editora, kde sa vykonajú zmeny.
- Aktuálny ACL zoznam sa odstráni pomocou príkazu **`no access-list`**.
- Modifikovaný ACL sa potom vloží do konfigurácie.

2. Spôsob: Sekvenčné čísla

- Sekvenčné čísla je možné použiť na odstránenie alebo vloženie ACL záznamu.
- Príkaz **`ip access-list extended`** *[meno]* sa používa na vstup do ACL konfigurácie.
- ACE je možné vložiť al. odstrániť.

V tomto príklade sa na opravu pomenovaného ACL SURFING použije 2. spôsob. ACL nesprávne povoľuje 192.168.11.0/24 a je upravený, aby povolil 192.168.10.0/24.

```
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 192.168.11.0 0.0.0.255 any eq www
    20 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1#
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#
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
```

Malo byť 192.168.10.0

IPv6 ACL

Vytvorenie IPv6 ACL

IPL6 ACL sú podobné ako IPv4 ACL vo fungovaní i konfigurácii.

V IPv4 existujú dva typy ACL (štandardné a rozšírené) a oba typy ACL môžu byť buď očíslované alebo menované ACL.

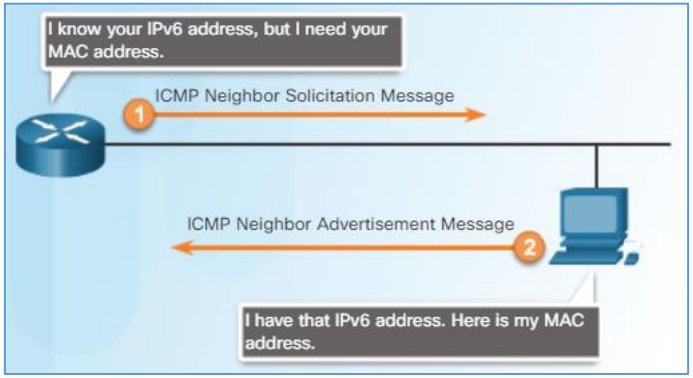
V IPv6 existuje iba jeden typ ACL, ktorý je ekvivalentný rozšírenému pomenovanému IPv4 ACL. V protokole IPv6 sa nenachádzajú žiadne číslované ACL.

Pozn.: IPv4 ACL a IPv6 ACL nemôžu zdieľať rovnaké meno.

Vytvorenie IPv6 ACL (pokr.)

Medzi ACL IPv4 a IPv6 existujú tri významné rozdiely:

- Príkaz používaný pre aplikáciu IPv6 ACL na rozhranie: **`ipv6 traffic-filter`**.
- IPv6 ACL nepoužívajú wildcard masky, ale namiesto toho určujú dĺžku prefixu – označenie koľko zdrojovej al. cieľovej IPv6 adresy sa má zhodovať.
- IPv6 ACL pridáva na konci každého IPv6 ACL zoznamu dva implicitné permit príkazy:
  - **`permit icmp any any nd-na`**
  - **`permit icmp any any nd-ns`**
  - **`deny ipv6 any any statement`**
- Tieto dva dodatočné príkazy umožňujú, aby správy IPv6 ICMP Neighbor Discovery (ND) a Neighbor Solicitation (NS) vykonávali to isté ako IPv4 ARP.

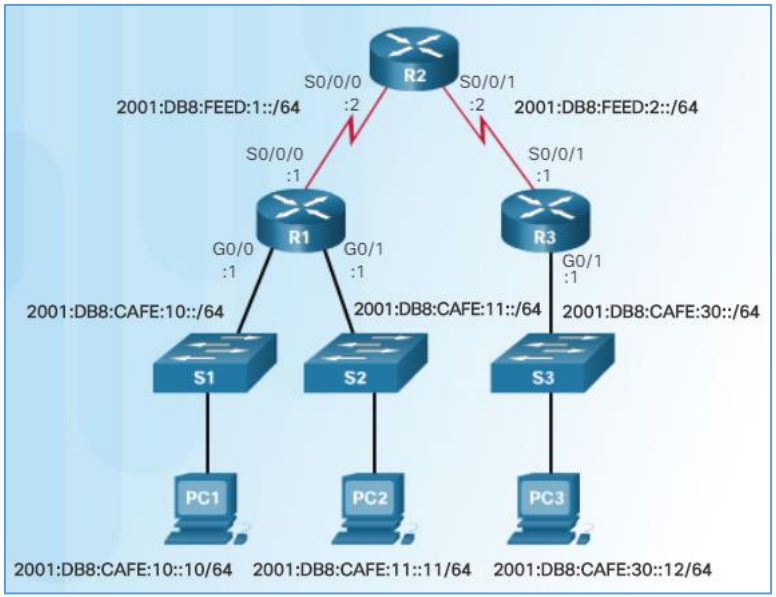


Konfigurácia IPv6 ACL (príklad)

Vzorová topológia pre ukážku IPv6 ACL.

Všetky rozhrania sú nakonfigurované a aktívne.

```
R1# show ipv6 interface brief
GigabitEthernet0/0      [up/up]
    FE80::FE99:47FF:FE75:C3E0
    2001:DB8:CAFE:10::1
GigabitEthernet0/1      [up/up]
    FE80::FE99:47FF:FE75:C3E1
    2001:DB8:CAFE:11::1
Serial0/0/0             [up/up]
    FE80::FE99:47FF:FE75:C3E0
    2001:DB8:FEED:1::1
 <output omitted>
R1#
```




```
R2# show ipv6 interface brief
Serial0/0/0             [up/up]
    FE80::FE99:47FF:FE71:78A0
    2001:DB8:FEED:1::2
Serial0/0/1             [up/up]
    FE80::FE99:47FF:FE71:78A0
    2001:DB8:FEED:2::2
 <output omitted>
R2#
```

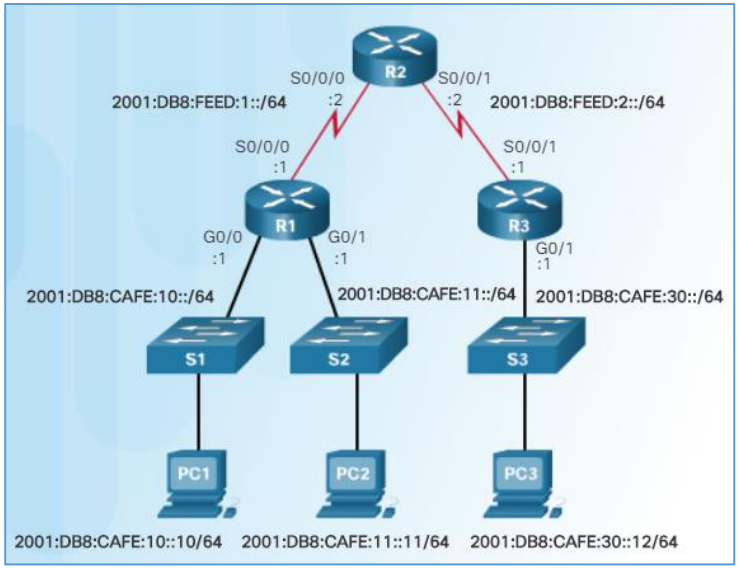
```
R3# show ipv6 interface brief
GigabitEthernet0/0      [up/up]
    FE80::FE99:47FF:FE71:7A20
    2001:DB8:CAFE:30::1
Serial0/0/1             [up/up]
    FE80::FE99:47FF:FE71:7A20
    2001:DB8:FEED:2::1
R3#
```

Konfigurácia IPv6 ACL (príklad) (pokr.)

Pri IPv6 sú iba pomenované ACL a konfigurácia je podobná rozšíreným IPv4 ACL.

```
R1(config)# ipv6 access-list access-list-name
R1(config-ipv6-acl)# deny | permit protocol {source-ipv6-prefix/prefix-length | any | host source-ipv6-address}
[operator [port-number]] {destination-ipv6-prefix/prefix-length | any | host destination-ipv6-address} [operator
[port-number]]
```

```
R1(config)# ipv6 access-list NO-R3-LAN-ACCESS
R1(config-ipv6-acl)# deny ipv6 2001:db8:cafe:30::/64 any
R1(config-ipv6-acl)# permit ipv6 any any
R1(config-ipv6-acl)# end
R1#
```



V tomto príklade:

1. Záznam označuje meno IPv6 ACL **NO-R3-LAN-ACCESS**.
2. Záznam zakazuje všetky IPv6 pakety z 2001:DB8:CAFE:30::/64 určené pre akúkoľvek IPv6 sieť.
3. Záznam povoľuje všetky ostatné IPv6 pakety.

Konfigurácia IPv6 ACL (príklad) (pokr.)

Po nakonfigurovaní sa IPv6 ACL prepája s rozhraním pomocou nasledujúceho príkazu:

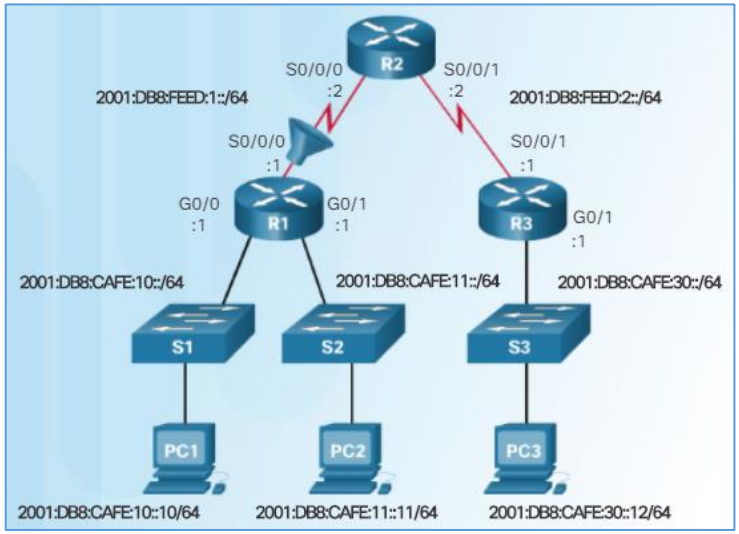
ipv6 traffic-filter *access-list-name* {**in** | **out**}

Príkaz aplikuje IPv6 ACL **NO-R3-LAN-ACCESS** na vstup do R1 rozhrania S0/0/0.

```
R1(config)# interface s0/0/0
R1(config-if)# ipv6 traffic-filter NO-R3-LAN-ACCESS in
```

Pre odstránenie IPv6 ACL, zadajte na rozhraní príkaz **no ipv6 traffic-filter** a následne v globálnom móde príkaz **no ipv6 access-list** na odstránenie ACL zoznamu.

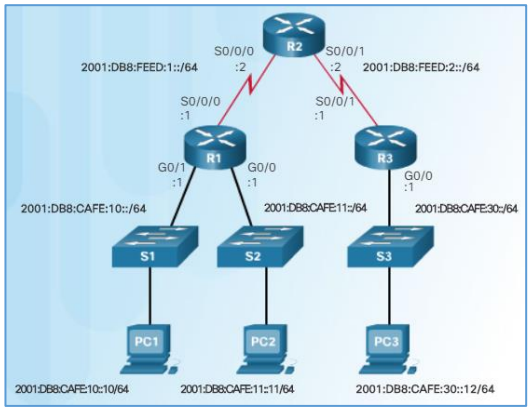
Všimnite si, že IPv4 aj IPv6 používajú príkaz **access-class** pre nasadenie ACL na VTY porty.



Konfigurácia IPv6 ACL (príklad)

V tomto príklade IPv6 ACL umožňuje používateľom v LAN R3 obmedzený prístup k LAN na R1.

1. Tieto záznamy umožňujú prístup z ľubovoľného zariadenia na webový server (2001: DB8:CAFE:10::10).
2. Všetkým ostatným zariadeniam sa odoprie prístup do siete 2001:DB8:CAFE:10::/64.
3. PC3 (2001:DB8:CAFE:30::12) má povolený Telnet prístup na PC2 (2001:DB8:CAFE:11::11).
4. Všetkým ostatným sa odoprie Telnet prístup k PC2.
5. Iná IPv6 prevádzka je povolená pre všetky iné ciele.
6. IPv6 ACL je aplikovaný vstup do rozhrania na G0/0, takže je ovplyvnená iba sieť 2001:DB8:CAFE:30::/64.




```
R3(config)# ipv6 access-list RESTRICTED-ACCESS
R3(config-ipv6-acl)# remark Permit access only HTTP and HTTPS to Network 10
R3(config-ipv6-acl)# permit tcp any host 2001:db8:cafe:10::10 eq 80          ] 1
R3(config-ipv6-acl)# permit tcp any host 2001:db8:cafe:10::10 eq 443
R3(config-ipv6-acl)# remark Deny all other traffic to Network 10
R3(config-ipv6-acl)# deny ipv6 any 2001:db8:cafe:10::/64                     2
R3(config-ipv6-acl)# remark Permit PC3 telnet access to PC2
R3(config-ipv6-acl)# permit tcp host 2001:DB8:CAFE:30::12 host 2001:DB8:CAFE:11::11 eq 23   3
R3(config-ipv6-acl)# remark Deny telnet access to PC2 for all other devices
R3(config-ipv6-acl)# deny tcp any host 2001:db8:cafe:11::11 eq 23             4
R3(config-ipv6-acl)# remark Permit access to everything else
R3(config-ipv6-acl)# permit ipv6 any any                                      5
R3(config-ipv6-acl)# exit
R3(config)# interface g0/0
R3(config-if)# ipv6 traffic-filter RESTRICTED-ACCESS in                       6
R3(config-if)#
```

Konfigurácia IPv6 ACL (show)

- Príkazy používané na overenie IPv6 ACL sú podobné ako IPv4 ACL.
- Príkaz **`show ipv6 interface`** zobrazuje ACL a smer v ktorom je nakonfigurovaný na rozhraní.
- Príkaz **`show access-lists`** zobrazuje všetky nakonfigurované IPv4 a IPv6 ACL.
- Príkaz **`show running-config`** zobrazuje všetky ACL záznamy vrátane poznámok.

```
R3# show ipv6 interface g0/0
GigabitEthernet0/0 is up, line protocol is up
 Global unicast address(es):
  2001:DB8:CAFE:30::1, subnet is 2001:DB8:CAFE:30::/64
 Input features: Access List
 Inbound access list RESTRICTED-ACCESS
<output omitted>
```

```
R3# show access-lists
IPv6 access list RESTRICTED-ACCESS
  permit tcp any host 2001:DB8:CAFE:10::10 eq www sequence 20
  permit tcp any host 2001:DB8:CAFE:10::10 eq 443 sequence 30
  deny ipv6 any 2001:DB8:CAFE:10::/64 sequence 50
  permit tcp host 2001:DB8:CAFE:30::12 host 2001:DB8:CAFE:11::11 eq
  telnet sequence 70
  deny tcp any host 2001:DB8:CAFE:11::11 eq telnet sequence 90
  permit ipv6 any any sequence 110
R3#
```

Riešenie problémov s ACL

Spracovanie paketov s ACL

Potrebné je zvážiť, ako sa spracováva ACL na vstupe a na výstupe.

ACL na **vstupe** (angl. *inbound*) pracuje nasledovne:

- Ak sa informácie **zhodujú** (hlavička paketu a ACL záznam), ostatné záznamy v ACL sa preskočia a paket je **povolený** alebo **odmietnutý** tak, ako je to uvedené v zodpovedajúcom zázname.
- Ak sa hlavička paketu **nezhoduje** s ACL záznamom, paket sa testuje voči **ďalšiemu záznamu** a tento proces zhody pokračuje až do konca ACL zoznamu.
- **Na konci** každého ACL je **implicitný zákaz** (`deny any`). Kvôli tomu by ACL malo mať aspoň jeden `permit`; inak bude ACL všetko blokovať.

ACL na **výstupe** (angl. outbound) pracuje nasledovne:

- Smerovač kontroluje **smerovaciu tabuľku**, aby zistil či je paket smerovateľný.
- Smerovač skontroluje, či je odchádzajúcemu rozhraniu priradený ACL zoznam.
- Ak je, tak záznamy ACL sa testujú voči hlavičke paketu.
- Na základe týchto testov je paket povolený alebo zahodený.

Bežné chyby pri konfig. ACL

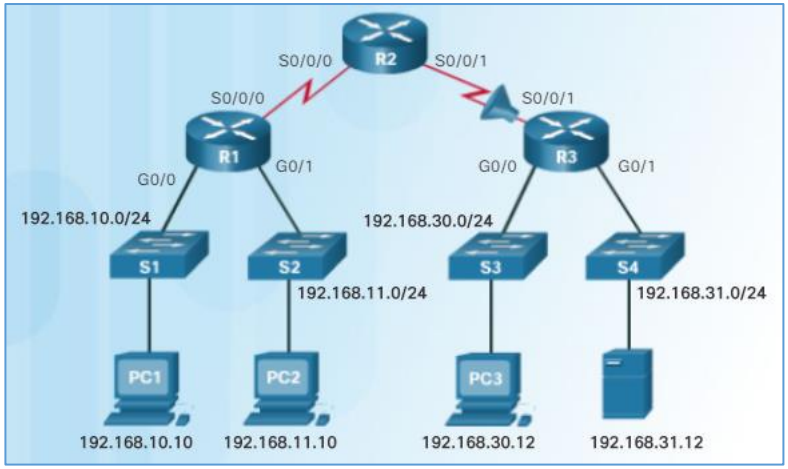
Najčastejšou chybou pri ACL je zadávanie záznamov v nesprávnom poradí al. neuplatňujú vhodných kritérií.

V tomto príklade PC 192.168.10.10 nemá Telnet pripojenie s 192.168.30.12.

- Príkaz **`show access-lists`** zobrazuje zhodu pre prvý zákaz, čo naznačuje, že tento záznam sa zhoduje s danou prevádzkou.

Riešenie:

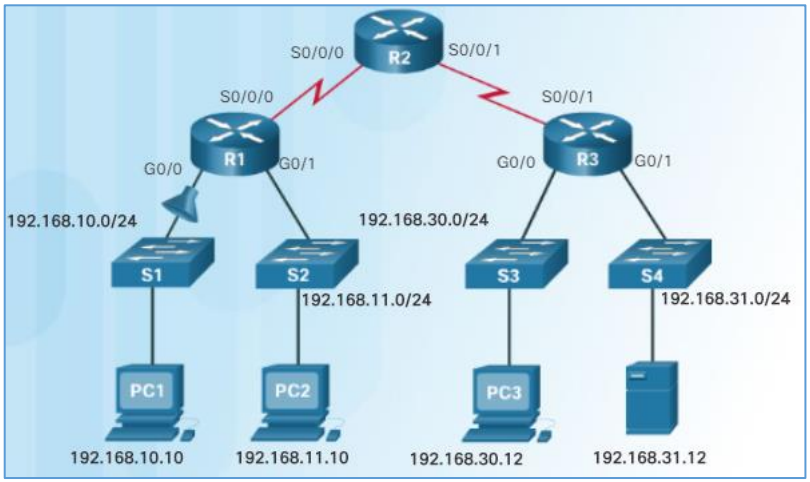
- Hostiteľ 192.168.10.10 nemá žiadnu konektivitu s 192.168.30.12, pretože záznam č. 10 zakazuje PC 192.168.10.10, preto záznam 20 nemôže byť nikdy aplikovaný.
- Záznamy 10 a 20 by mali byť prehodené.




```
R3# show access-lists
Extended IP access list 110
    10 deny tcp 192.168.10.0 0.0.0.255 any (12 match(es))
    20 permit tcp 192.168.10.0 0.0.0.255 any eq telnet
    30 permit ip any any
```

Bežné chyby pri konfig. ACL (pokr.)

V tomto príklade nemôže sieť 192.168.10.0/24 použiť TFTP na pripojenie k sieti 192.168.30.0/24.



Riešenie:

- Záznam 30 v ACL 120 povoľuje všetko.
- Avšak, TFTP používa UDP namiesto TCP, a preto je implicitne zakázané.
- Záznam 30 by mal byť **`permit ip any any`**.

```
R3# show access-lists 120
Extended IP access list 120
    10 deny tcp 192.168.10.0 0.0.0.255 any eq telnet
    20 deny tcp 192.168.10.0 0.0.0.255 host 192.168.31.12 eq smtp
    30 permit tcp any any
```

Bežné chyby pri konfig. ACL (pokr.)

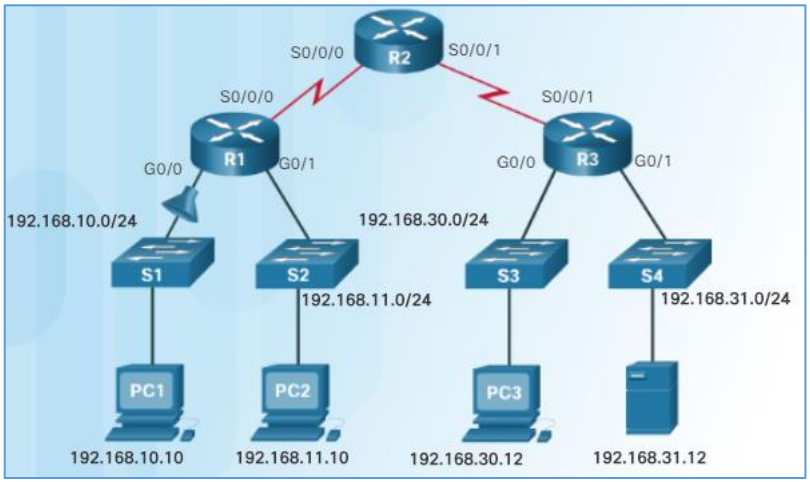
V tomto príklade je na R1 konfigurovaný IPv6 ACL, kt. má zakázať na prístup FTP zo siete :10 do siete :11.

- Avšak po nakonfigurovaní ACL sa PC1 stále dokáže pripojiť k serveru FTP na PC2.
- Výstup príkazu **`show ipv6 access-list`** zobrazuje zhodu pre **`permit`** záznam.

Riešenie:

ACL bola nasadená so správnym názvom, ale nie správnym smerom.

Na riešenie je potrebné odstrániť
`ipv6 traffic-filter NO-FTP-TO-11 out`
a nahradiť ho
`ipv6 traffic-filter NO-FTP-TO-11 in`.




```
R3# show access-lists 120
Extended IP access list 120
    10 deny tcp 192.168.10.0 0.0.0.255 any eq telnet
    20 deny tcp 192.168.10.0 0.0.0.255 host 192.168.31.12 eq smtp
    30 permit tcp any any
```

Ďakujem za pozornosť