ACL

Bevezetés

Ez a leírás a forrásban megjelölt anyag alapján készült. Az utasítások át lettek alakítva a Cisco PacketTracer 7.0 routereihez. A CPT-ben például nincs ftp-data port, helyette 20 számot írtam. A forrásban megadott dinamikus útválasztó protokollok egyáltalán nem használhatók.

Fogalmak

- helyettesítő_maszk = wilcard_maszk

Helyettesítő maszk például: Ha a normál maszk 255.255.255.0, akkor a wildcard_maszk: 0.0.0.255

Előfeltevés

Mielőtt elkezdesz ACL-t állítgatni, a következő ismeretek szükségesek:

- alapvető IP címzési ismertek
- szoftveres portok ismerete
- TCP és UDP protokoll ismerete
- HTTP, FTP, SMTP, POP3, IMAP, Telnet, SSH alapszintű ismerete

Az ACL-ek osztályozása

- szabványos ACL-ek
- kiterjesztett ACL-ek
- dinamikus ACL-ek (lock és key)
- IP alapú ACL-ek
- rugalmas ACL-ek
- idő alapú ACL-ek, időtartományok
- magyarázott IP ACL-ek
- kontextus alapú ACL-ek
- proxy azonosítás
- terjesztett idő alapú ACL-ek

Ez a dokumentum a szabványos és kiterjesztett ACL-ekről szól.

Szabványos ACL-ek

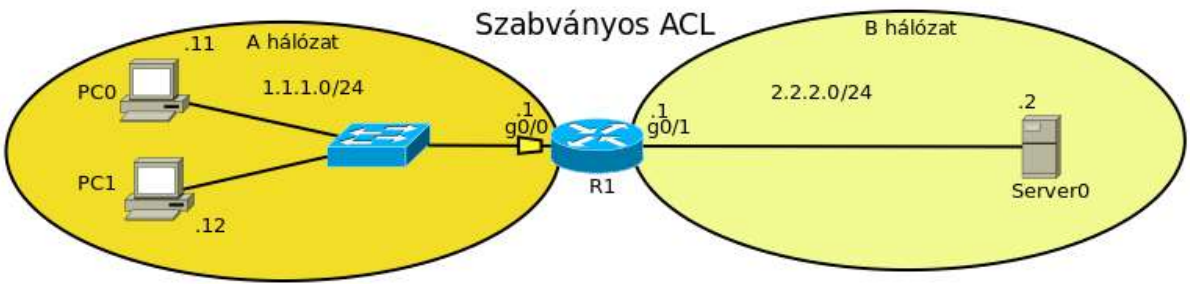
Az ACL-ek általános szintaxisa:

```
access-list lista_száma {permit|deny}
  {forrásIpCím|forrásIpTartomány helyettesítő_maszk|any}
```

A szabványos ACL-ek sorba, egymás után, összehasonlítják az IP csomagok forrás IP címét a beállítottakkal, ami alapján szabályozzák a forgalmat. A lista száma 1 és 99 közé kell essen szabványos ACL esetén. 100 és 199 közötti azonosítót a kiterjesztett ACL-ekhez használhatók.

A példa kedvéért legyen két gép, amely forgalomirányítón keresztül éri el a szervert:

- PC0: 1.1.1.11
- PC1: 1.1.1.12
- Server0: 2.2.2.2



Csak a PC0 gép számára szeretném engedélyezni a szerver elérését.

Elsőként létre kell hoznunk egy szabályt, egy sorszámmal, majd hozzá kell rendelni az egyik hálózati interfészhez.

```
R1(config)#access-list 1 permit 1.1.1.11
R1(config)#int g0/0
R1(config-if)#ip access-group 1 in
R1(config-if)#end
```

Az első parancs létrehoz egy 1-s azonosító számú hozzáférési listát, egyetlen szabállyal, amely megengedi a 1.1.1.11-s gépnek, hogy átmenjen a forgalomirányítón. A második parancs kiválasztja a g0/0 interfészt. A következő hozzárendeli az 1-es azonosítójú listát az interfészhez. Az alapértelmezett irányelv a tiltás, így minden tiltva van, amit nem engedünk meg.

A 1.1.1.11 gép így eléri a szervert, az 1.1.1.12 gép viszont nem.

Ha később a PC1 számára is szeretném megengedni a szerver elérést, akkor írjuk be:

```
R1(config)#access-list 1 permit 1.1.1.12
```

A szabályokat úgy is összeállíthatjuk, hogy alapértelmezetten mindent engedünk, és beállítjuk mit tiltunk:

```
R1(config)#access-list 1 deny host 1.1.1.12
R1(config)#access-list 1 permit any
```

A sorrend nem mindegy:

```
R1(config)#access-list 1 permit any
R1(config)#access-list 1 deny host 1.1.1.12
```

Ha egy szabály illeszkedik egy csomagra, akkor az a szabály alkalmazásra kerül, és az utána következő szabályokat már figyelmen kívül hagyja. A fenti utóbbi tiltás, tehát nem fog működni, mert hamarabb van egy mindent megengedő szabály.

A vty terminálokon az „ip access-group" helyett használjuk az „access-class" utasítást:

```
R1(config)#access-list 1 permit 1.1.1.11
R1(config)#line vty 0 15
R1(config-if)#access-class 1 in
R1(config-if)#end
R1# show access-list
```

Számozott ACL javítása

A számozott ACL-ek a nevesített módban javíthatók, az alábbi módon:

```
R1(config)# access-list 1 deny 192.168.10.5 0.0.0.0
R1(config)# access-list 1 permit 192.168.10.0 0.0.0.255
R1(config)# exit
R1#
R1# show access-lists 1
Standard IP access list TILT
    10 deny  192.168.10.5
    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 192.168.10.11
R1(config-std-nacl)# end
R1#
R1# show access-lists
Standard IP access list 1
    10 deny  192.168.10.11
    20 permit 192.1678.10.0, wildcard bits 0.0.0.255
R1#
```

A forgalomirányítók a show access-list parancs esetén megmutatják a sorszámot is, ahogy az előbb kódban látható. A **Packet Tracerben** nem jelennek meg a sorszámok. Ha azonban megadjuk a lista számát:

```
# show access-list 1
```

Ebben az esetben megjelenik a sorszám is.

Ha belépünk ACL javítómódba:

```
R1(config)# ip access-list standard 1
R1(config-std-nacl)#
```

A „do show access-list" paranccsal szintén megnézhetők az beállítások. A **Packet Tracer** itt fordítva működik; A „do show access-list 1" esetén nem mutatja a sorszámot; A „do show access-list" esetén mutatja.

Kiterjesztett ACL-ek

A kiterjesztett ACL-ek

A kiterjesztett ACL-ek sorba egymás után összehasonlítják az IP csomagok forrás és cél címeit az ACL-ben beállítottakkal, ami alapján szabályozzák a forgalmat. A kiterjesztett ACL segítségével lehetőség van a következők szűrésére is:

- protokoll
- port szám
- különböző szolgáltatáskódok (DSCP)
- precedencia alapján
- a szinkronizáló sorszám állapot alapján (SYN bit)

A parancs általános szintaxisa:

```
access-list lista_száma {deny | permit} protocol
forrás forrás_helyettesítő_maszk [forrásport]
cél cél_helyettesítő_maszk [célport]
```

A következőkben olyan szintaktikát látunk, amely a fizikai eszközök IOS rendszerein elérhető, de Packet Tracerben nem.

IP esetén:

```
access-list lista_száma [dynamic dynamic_név [timeout perc]]
{deny | permit}
protokoll
forrás forrás_wildcard_maszk
cél cél_wildcard_maszk
[precedence precedencia] [tos tos] [log | log-input]
[time-range idő_intervallum][töredékek]
```

ICMP esetén

```
access-list lista_száma [dynamic dynamic-név [timeout perc]]
{deny | permit}
icmp
forrás forrás_wildcard_maszk
cél cél_wildcard_maszk
[icmp-type [icmp-kód] | [icmp-üzenet]] [precedence precedencia]
[tos tos] [log | log-input] [time-range idő_intervallum
[fragments]
```

TCP esetén

```
access-list lista_száma [dynamic dynamic-név [timeout perc]]
{deny | permit} tcp
source forrás_wildcard_maszk [operator [port]]
destination cél_wildcard_maszk [operator [port]]
[established] [precedence precedencia] [tos tos]
[log | log-input] [time-range idő_intervallum_név][töredékek]
```

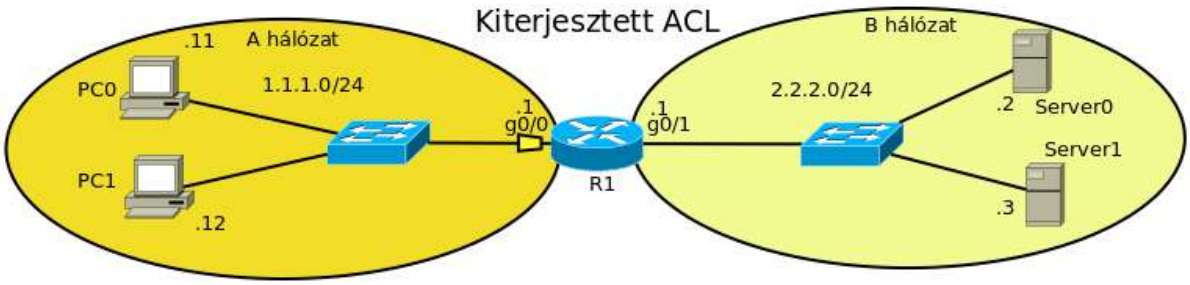
UDP

```
access-list lista_száma [dynamic dynamic-név [timeout perc]]
{deny | permit} udp
```

```
source forrás_wildcard_maszk [operator [port]]
destination cél_wildcard_maszk
[operator [port]] [precedence precedencia] [tos tos]
[log | log-input] [time-range idő_intervallum_név][töredékek]
```

Adott port tiltása

Szeretnénk egy adott szolgáltatás forgalmát letiltani. Legyen a példa kedvéért két hálózat, A és B. Mindként hálózatban két gép:



Szeretnénk tiltani a routeren a telnet elérését.

A szervereken a web tiltása a forgalomirányítón:

```
R1(config)#int g0/0
R1(config-if)#ip access-group 100 in
R1(config-if)#exit
R1(config)#access-list 100 deny tcp any any eq 80
R1(config)#access-list 100 permit ip any any
```

Elemezzük a tiltó parnacsot:

```
R1(config)#access-list 100 deny tcp any any eq 80
```

- 100 - az ACL bejegyzés száma - ACE (Access Control Entry)
- deny - tiltás
- Illeszkednie kell a csomagnak a következőkre:
    - tcp - TCP forgalom legyen
    - any - (első any) bárhonnan jön - forrás
    - any - (második any) bárhova megy - cél
    - eq 80 - célport 80 (webszerver)

Láthatjuk, hogy csak a webes elérés van tiltva. Bármelyik PC-ből ha böngészőben megpróbáljuk megnézni a bármelyik szerver weblapját, az nem elérhető.

Ha viszont, bármely PC-ről a ping paranccsal megvizsgálom a kapcsolatot a szerverekhez, a kapcsolat működni fog. Működnie is kell mivel a ping ICMP protokollt használ.

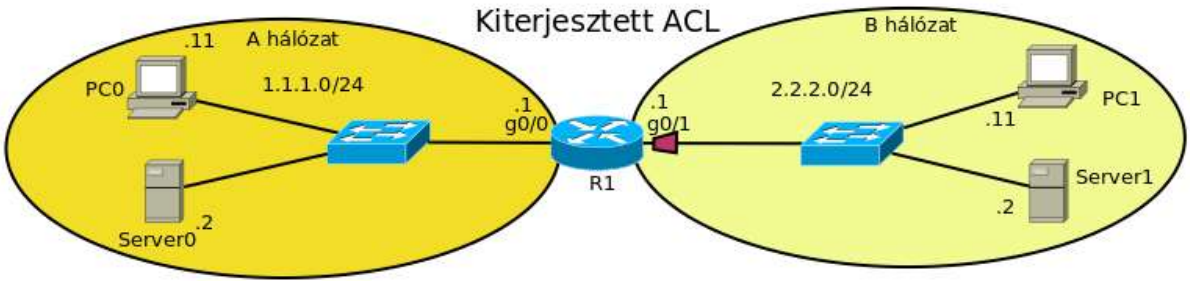
Munkamenet csak a belső hálózatról

Ha TCP kapcsolatot használ egy hálózati eszköz, akkor kiépíti vagyis stabilizálja a kapcsolatot. A kiépített kapcsolatot munkamenetnek hívjuk. A munkamenetben a kapcsolat kétirányú. A kapcsolatkezdeményezőtől van egy kapcsolat a cél felé, és céltól is mennek

válasz csomagok a kezdeményező felé. Utóbbi csomagok a már kiépített, stabil munkamenethez tartoznak.

A már kiépített kapcsolatokra beállíthatók szűrési szabályok.

Tegyük fel, hogy van A és B hálózat. Szeretnénk, ha az A hálózatról elérhető lenne a B hálózat, de a B hálózatról ne lehessen elérni az A hálózatot.



A szándékaink tehát a következők:

- A hálózatról kezdeményezhető munkamenet a B hálózat felé
- B hálózatról nem kezdeményezhető munkamenet az A hálózat felé

A konfiguráció megengedi, hogy ha egy csomag az GigabitEthernet 0/1 interfészen tart befele, akkor azt elfogadjuk ha:

- nincs beállítva a TCP fejlécben a SYN jelző
- a célcím nagyobb mint 1023

Magyarázat:

- Ha be van állítva a SYN jelző a TCP fejlécben, akkor éppen egy új kapcsolat kiépítése történik.
- Ha cél nagyobb mint 1023, csak akkor lehet a válasz forgalom.

Beállítás:

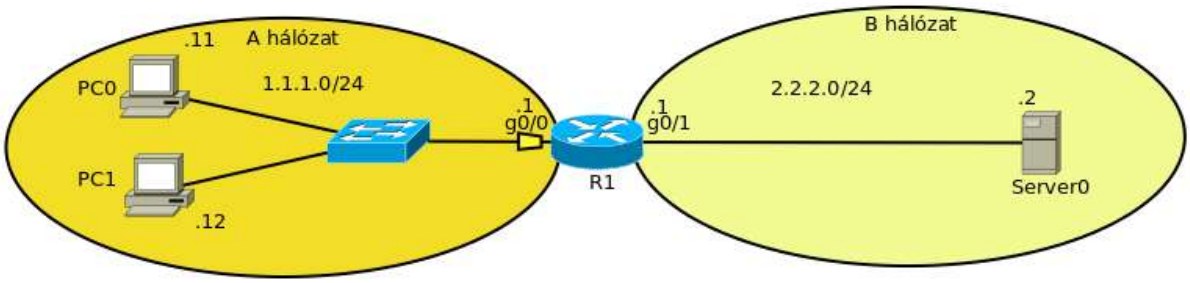
```
R1(config)#access-list 101 permit tcp any any gt 1023 established
R1(config)#int g0/1
R1(config-if)#ip access-group 101 in
```

FTP tiltása

Ha FTP-t használunk mindig két csatorna épül ki egy kapcsolathoz. Egy **vezérlő** és egy **adat** csatorna. Bármelyiket tiltjuk a a kapcsolatot megakadályoztuk.

A következő példában az FTP kapcsolat vezérlő és adat csatornáját tiltjuk.

Topológia:



Beállítások:

```
R1(config)#int g0/0
R1(config-if)#ip access-group 100 in
R1(config-if)#exit
R1(config)#access-list 100 deny tcp any any eq ftp
R1(config)#access-list 100 deny tcp any any eq ftp-data
R1(config)#access-list 100 permit ip any any
```

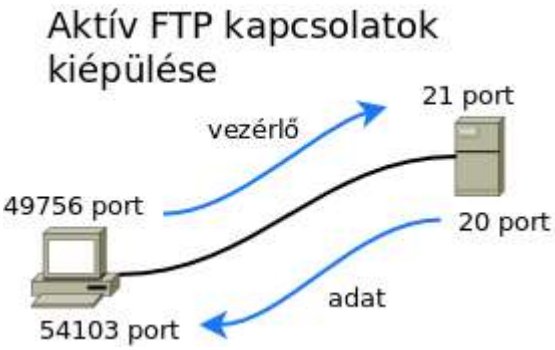
Lista ürítése:

```
R1(config)#no access-list 100
```

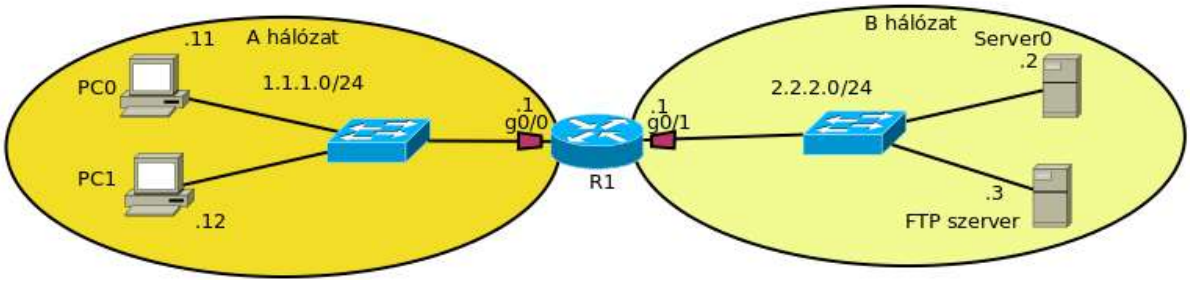
Aktív FTP engedése

Az FTP kapcsolat lehet aktív és passzív.

Aktív kapcsolat esetén a kliens kiépít egy kapcsolatot a szerver 21-es portjára. A szerver is kiépít saját 20-as portjáról egy adatkapcsolati csatornát a kliens felé.



Aktív FTP kapcsolat engedélyezése:



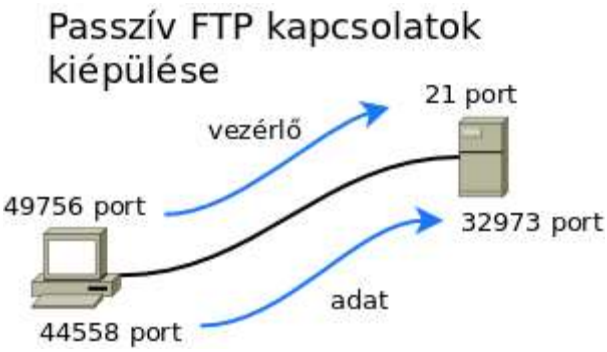

```
R1(config)#int g0/0
R1(config-if)#ip access-group 100 in
```

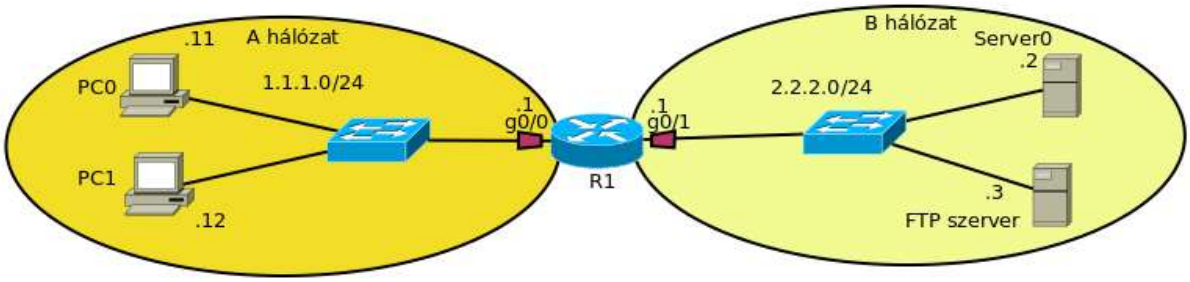
```
R1(config-if)#exit
R1(config)#access-list 100 permit tcp any host 2.2.2.3 eq ftp
R1(config)#access-list 100 permit tcp any host 2.2.2.3 eq 20 established

R1(config)#int g0/1
R1(config-if)#ip access-group 101 in
R1(config)#exit
R1(config)#access-list 101 permit host 2.2.2.3 eq ftp any established
R1(config)#access-list 101 permit host 2.2.2.3 eq 20 any
```

Passzív FTP engedése

Passzív kapcsolat esetén a kliens kezdeményez egy kapcsolatot a szerver 21-s portjára, majd kezdeményez egy másik kapcsolatot a szerver 1023 feletti portjára.

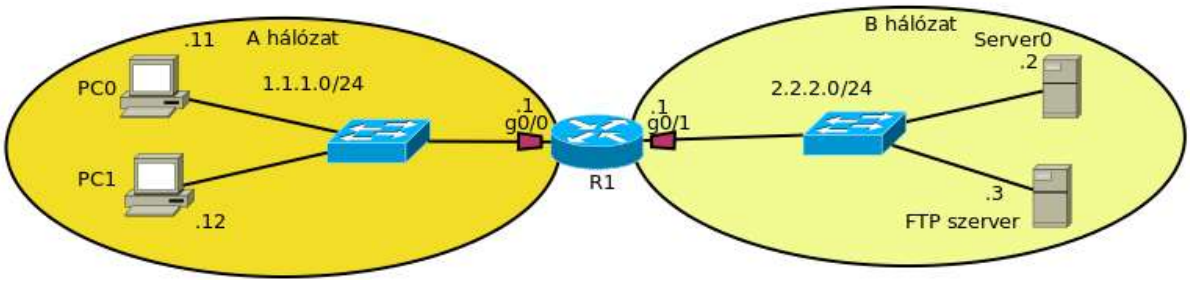
Az következő ábra egy lehetséges portkiosztást mutat be:






```
R1(config)#int g0/0
R1(config-if)#ip access-group 100 in
R1(config-if)#exit
R1(config)#access-list 100 permit tcp any host 2.2.2.3 eq ftp
R1(config)#access-list 100 permit tcp any host 2.2.2.3 gt 1024

R1(config)#int g0/1
R1(config-if)#ip access-group 101 in
R1(config-if)#exit
R1(config)#access-list 101 permit tcp host 2.2.2.3 eq ftp any established
R1(config)#access-list 101 permit tcp host 2.2.2.3 gt 1024 any established
```

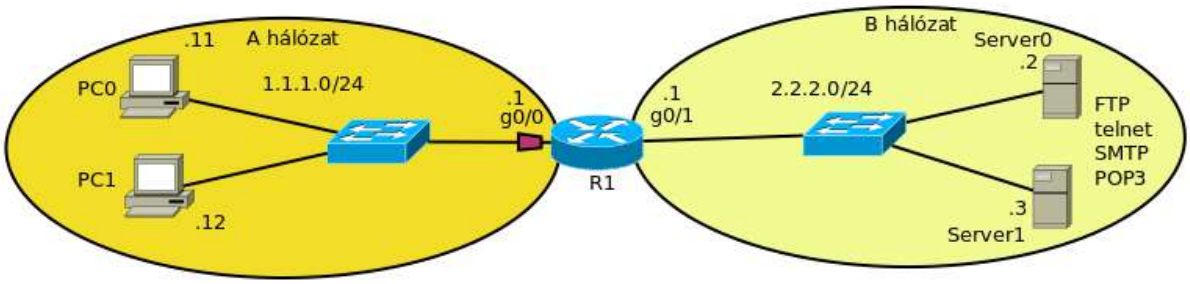
ICMP engedése




```
R1(config)#int g0/0
R1(config-if)#ip access-group 100 in
R1(config-if)#exit
R1(config)#access-list 100 permit icmp any any echo

R1(config)#int g0/1
R1(config-if)#ip access-group 101 in
R1(config-if)#exit
R1(config)#access-list 101 permit icmp any any echo-reply
```

Több szolgáltatás engedése

A www, telnet, smtp, pop3 és ftp engedése. Tegyük fel, hogy az előbbi szolgáltatásokat szeretnénk elérni a B hálózat egyik szerverén:



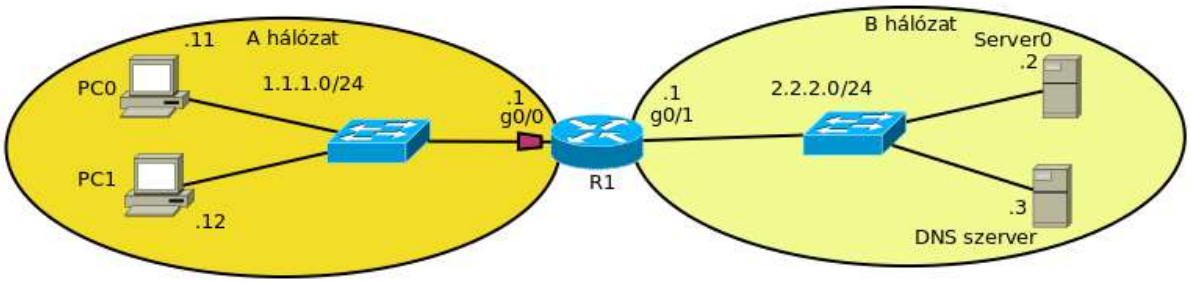
Megvalósítás a következő módon:

```
R1(config)#int g0/0
R1(config-if)#ip access-group 100 in
R1(config-if)#exit
R1(config)#access-list 100 permit tcp any any eq www
R1(config)#access-list 100 permit tcp any any eq telnet
R1(config)#access-list 100 permit tcp any any eq smtp
R1(config)#access-list 100 permit tcp any any eq pop3
R1(config)#access-list 100 permit tcp any any eq 21
R1(config)#access-list 100 permit tcp any any eq 20
```

DNS engedése

A DNS szervert UDP és TCP-én keresztül is engedni kell. UDP csatornán megy a szimpla névlekérdezés. TCP-én keresztül történik a zónafájlok átvitele az elsődlees szerverről a másodlagos szerverre.

Legyen A és egy B hálózat:



A B hálózatban van egy DNS szerver, amit az A hálózat gépei használnak. Ennek engedélyezése:

```
R1(config)#int g0/0
R1(config-if)#ip access-group 100 in
R1(config-if)#exit
R1(config)#access-list 100 permit udp any any eq domain
R1(config)#access-list 100 permit tcp any any eq 53
```

A tcp esetén nem használható a domain kulcsszó. Helyette 53-as port.

A DNS szerver tesztelése:

```
nslookup valami.hu 2.2.2.3
```

A valami.hu a kereset tartománynév, az IP cím a DNS szerver IP címe.

Nevesített szabványos ACL

Nevesített szabványos ACL:

```
R1(config)# ip access-list standard S1
R1(config-std-nacl)# deny 192.168.10.11 0.0.0.0
R1(config-std-nacl)# end
```

Nevesített szabványos ACL módosítása:

```
R1# show access-lists
Standard IP access list TILT
    10 deny  192.168.11.10
    20 permit 192.168.10.0, wildcard bits 0.0.0.255
R1#conf t
R1(config)# ip access-list standard TILT
R1(config-std-nacl)# 15 deny host 192.168.11.11
R1(config-std-nacl)# end
R1# show access-lists
Standard IP access list TILT
    10 deny  192.168.11.10
    15 deny  192.168.11.11
    20 permit 192.1678.10.0, wildcard bits 0.0.0.255
R1#
```

Nevesített kiterjesztett ACL

Használat:

```
R1(config)# ip access-list extended KER
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 NEZ
R1(config-ext-nacl)# permit tcp any 192.168.10.0 0.0.0.255 established
R1(config-ext-nacl)# exit
R1(config)# int g0/0
R1(config-if)# ip access-group KER in
R1(config-if)# ip access-group NEZ out
```

Ellenőrzés:

```
R1# show access-lists
R1# show ip int g0/0
```

VTY vonalak védelme

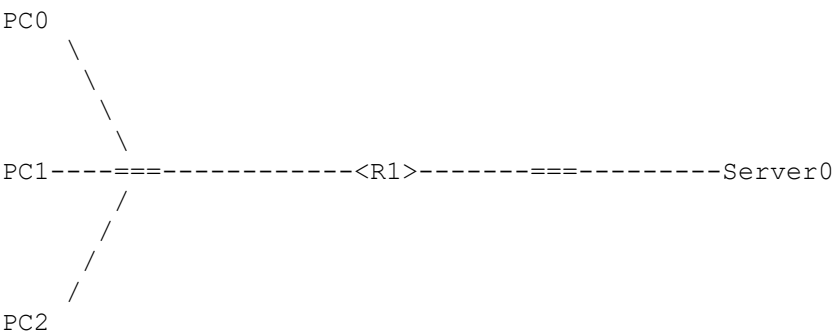
```
R1(config)# line vty 0 4
R1(config-line)# login local
R1(config-line)# transport input ssh
R1(config-line)# access-class 2 in
R1(config-line)# exit
R1(config)# access-list 2 permit 192.168.10.0 0.0.0.255
R1(config)# access-list 2 deny any
```

Függelék

Példák 001

Adott 3 PC:

- 192.168.5.10
- 192.168.5.11
- 192.168.5.12



Csak a 192.168.5.10 érheti el router mögötti szervert:

```
R1(config)#access-list 1 permit 192.168.5.10
R1(config)#int g0/0
R1(config-if)#ip access-group 1 in
```

```
R1(config-if)#end
```

Az „1”-es az ACL neve. Az in befele jövő csomagokra vonatkozik a beállítás. Az in helyet írhatunk out-t.

Engedélyezés egy célhálózathoz:

```
R1(config)#access-list 1 permit 195.100.100.0 0.0.0.255
```

Tiltás egy célgéphez:

```
R1(config)#access-list 1 deny host 180.100.100.3
```

Illeszkedés forrásra és célra is:

```
R1(config)#access-list 100 deny
  tcp 195.100.100.0 0.0.0.255 0.0.0.0 0.0.0.0 eq 80
```

A példában ha bármely webszerverhez jön egy kapcsolat a 195.100.100.0 hálózatból, akkor azt eldobjuk.