

# Hálózati architektúrák és Protokollok

## PTI – 6

Kocsis Gergely  
2018.04.11.

# Hálózati konfiguráció

---

## \$ ifconfig

Kapcsoló nélkül kiírja a csomópont aktuális hálózati interfész beállításait. Kapcsolókkal alkalmas azok beállítására.

**\$ ifconfig** interfész név ip\_cím **broadcast** broadcast\_cím **netmask** netmaszk

Feladat: Milyen módon lehetne bállítani az aktuális csomóponton, hogy az az **eth0** interfészen keresztül csatlakozzon a **191.168.0.0/16** hálózatba, míg **eth1** interfészen keresztül a **193.6.181.0/26** hálózatba. Mindkét esetben a legkisebb kisztható címet adjuk meg.

```
$ ifconfig eth0 191.168.0.1 broadcast 191.168.255.255 netmask 255.255.0.0
```

```
$ ifconfig eth1 193.6.181.1 broadcast 193.6.181.63 netmask 255.255.255.192
```



# Hálózati konfiguráció

## \$ route

Kapcsoló nélkül kiíratja a route táblát.

Kapcsolókkal alkalmas sorokat adhatunk a táblához.

```
$ route add -net hálózat_cím netmask netmaszk {dev interfész | gw ip_cím}
```

```
$ route add default gw ip_cím
```

Feladat: Milyen módon lehetne bállítani az aktuális csomóponton, hogy az az **eth0** interfészen keresztül csatlakozzon a **191.168.0.0/16** hálózatba, míg **eth1** interfészen keresztül a **193.6.181.0/26** hálózatba. Mindkét esetben a legkisebb kisztható címet adjuk meg.

Állítsuk be a fenti csomópont route tábláját úgy hogy az a csomagokat a megfelelő hálóba továbbítsa. Az alapértelmezett átjáró címe legyen **193.6.181.1**

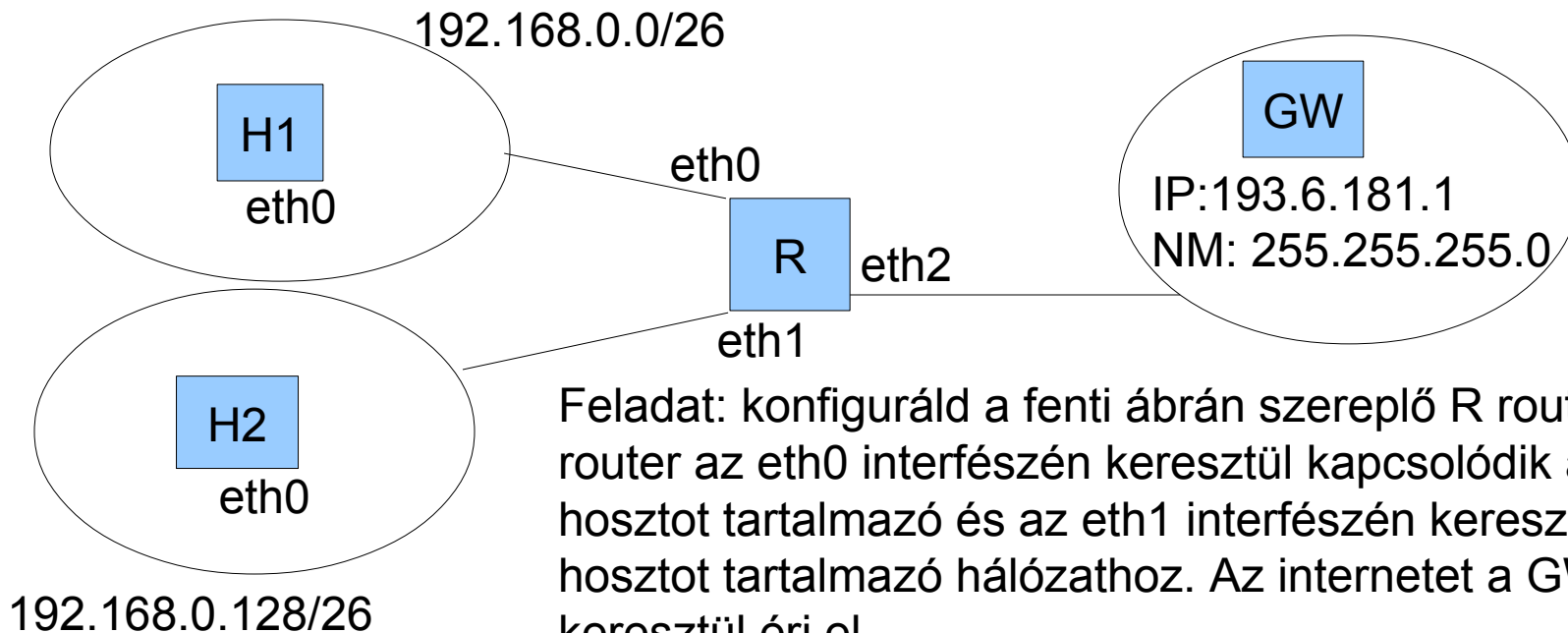
```
$ route add -net 191.168.0.0 netmask 255.255.0.0 dev eth0
```

```
$ route add -net 193.6.181.0 netmask 255.255.255.192 dev eth1
```

```
$ route add default gw 193.6.181.1
```



# Hálózati konfiguráció



```
$ ifconfig eth0 192.168.0.1 broadcast 192.168.0.63 netmask 255.255.255.192  
$ ifconfig eth1 192.168.0.129 broadcast 192.168.0.191 netmask 255.255.255.192  
$ ifconfig eth2 193.6.181.2 broadcast 193.6.181.255 netmask 255.255.255.0  
$ route add -net 192.168.0.0 netmask 255.255.255.192 dev eth0  
$ route add -net 192.168.0.128 netmask 255.255.255.192 dev eth1  
$ route add -net 193.6.181.0 netmask 255.255.255.0 dev eth2  
$ route add default gw 193.6.181.1
```



# Knoppix első lépések

---

Parancssori manuál:

**\$ man {parancsnév}**

Pl. \$ man ifconfig

Hálózati beállításokat kizárólag rendszergazdai jogosultsággal módosíthatunk!

Rendszergazdai jogosultsági szint (root shell) terminálablakban:

**\$ su -**

Billentyűzetkiosztás átváltása terminálablakban:

**\$ setxkbmap {hu|us}**

Pl. **\$ setxkbmap hu**



# Hálózati interfész státusza

---

Interfész státuszának módosítása

```
$ ifconfig eth0 {up|down}
```

Pl. Helyezzük az interfészt passzív, majd ismét aktív státuszba:

```
$ ifconfig eth0 down
```

```
$ ifconfig eth0 up
```

Mindkét parancs kiadása után ellenőrizzük az interfész beállításait az `ifconfig eth0` paranccsal. Hogyan állapítható meg az `ifconfig` kimenet alapján az interfész státusza?



# IP cím konfiguráció

---

## Feladat:

Konfiguráljuk fel a 10.0.1.x/24 IP címet számítógépünk eth0-s interfészen!

```
$ ifconfig eth0 10.0.1.x netmask 255.255.255.0 broadcast 10.0.1.255 up
```

Kompakt forma:

```
$ ifconfig eth0 10.0.1.x/24 broadcast 10.0.1.255 up
```

Töröljük az interfész IP címét:

```
$ ifconfig eth0 0.0.0.0
```

Dinamikus IP cím konfiguráció:

```
$ dhclient eth0
```



# IP kommunikáció ellenőrzése (ping)

A ping parancs az IP összeköttetés ellenőrzésére használható. ICMP (Internet Control Message Protocol) ECHO REQUEST csomagokat küld ki, melyre a megszólított csomópont ICMP ECHO REPLY üzenettel válaszol.

```
ping [opciók] {ip_cím|domain_név}
ping 193.6.128.5
ping -c 4 www.c3.hu
ping -s 1000 193.6.128.25
ping -w 3 10.0.1.1
```

## Opciók:

- -b: broadcast cím pingelése
- -c {szám}: adott számú ECHO REQUEST kérés küldése után leáll a program
- -f : elárasztásos (folyamatos) ping
- -s {méret}: a kiküldött ICMP ECHO REQUEST csomagok mérete bájtban
- -w {időtartam}: időtúllépés megadása másodpercben



# A route parancs

Statikus útvonal bejegyzése:

```
$ route add -net 10.0.1.0/24 dev eth0
```

```
[root@erlang ~]# route add -net 10.0.1.0/24 dev eth0
[root@erlang ~]# route -n
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
10.0.1.0         0.0.0.0         255.255.255.0   U        0      0      0 eth0
193.6.135.0     0.0.0.0         255.255.255.0   U        0      0      0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U       1002    0      0 eth0
0.0.0.0         193.6.135.1    0.0.0.0         UG        0      0      0 eth0
```

Statikus útvonal bejegyzése:

```
$ route add -net 10.0.2.0/24 gw 10.0.0.1 dev eth0
```

```
[root@erlang ~]# ifconfig eth0:0 10.0.0.2/24
[root@erlang ~]# route add -net 10.0.2.0/24 gw 10.0.0.1 dev eth0
[root@erlang ~]# route -n
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
10.0.0.0         0.0.0.0         255.255.255.0   U        0      0      0 eth0
10.0.1.0         0.0.0.0         255.255.255.0   U        0      0      0 eth0
10.0.2.0         10.0.0.1        255.255.255.0   UG        0      0      0 eth0
193.6.135.0     0.0.0.0         255.255.255.0   U        0      0      0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U       1002    0      0 eth0
0.0.0.0         193.6.135.1    0.0.0.0         UG        0      0      0 eth0
[root@erlang ~]# █
```



# A route parancs

---

Default útvonal bejegyzése:

```
$ route add default gw 10.0.0.1
```

## Feladat:

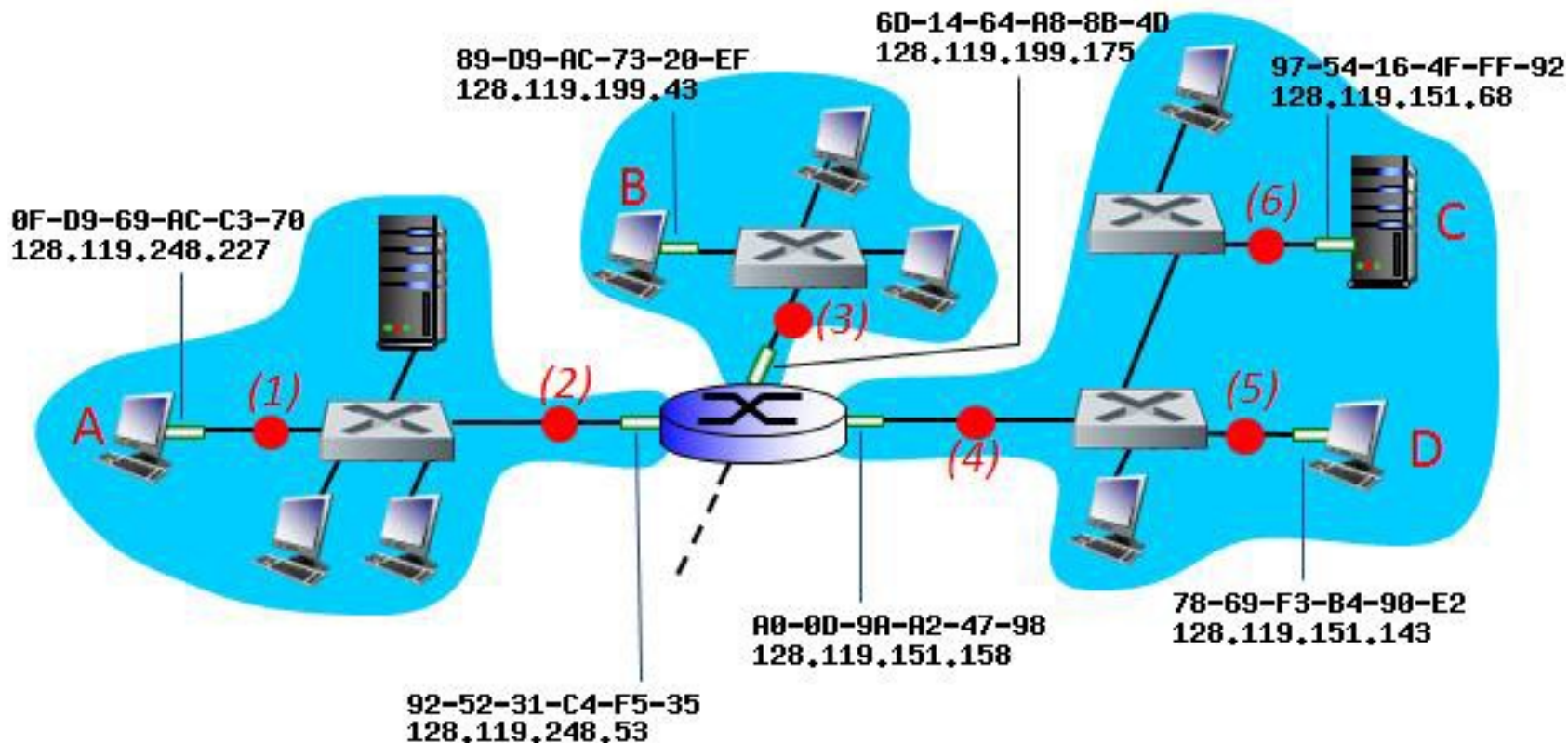
- A tanteremben alakítsunk ki 4 db **/26**-os alhálózatot a **10.0.1.0/24** kiinduló hálózatból.

A hallgatói gépek oszloponként egy IP hálózatba tartoznak, mindenki konfiguráljon fel egy szabad címet az eth0 interfészén a megfelelő alhálózatból.

- Pingeljünk meg az oszlopunkban (saját hálózatunkban) egy másik csomópontot.
- Pingeljünk meg egy másik oszlopbeli csomópontot. Kapunk választ? Magyarázzuk meg a jelenséget!
- Módosítsuk úgy a routing táblánkat, hogy a terem minden alhálózatát el tudjuk érni!



# IP, MAC, ARP



A B csomópontból az A-ba küldünk egy datagramot. Mik lesznek az Ethernet keretben található forrás és a cél címek (MAC cím és IP cím) az ábra szerinti (3), (2), és (1) pontokban? (a négyzetek switch-et, a henger routert jelöl)

Milyen címeket kapunk az (1), (2), (4) és (6) pontokon, ha az A csomópontból a C csomópontba küldünk datagramot?



# Hálózati konfiguráció

