

# Hálózatok építése és üzemeltetése

ZH konzultáció

# ZH

---

- ▶ **Teszt feladatok lesznek**
- ▶ **GO / NO-GO**
- ▶ **Elektronikus rendszerben, Google Forms**
  - ▶ IB026, QBF13: kell notebook
  - ▶ IB.213: aki nem tud gépet hozni (de csak kb. 25 hely)
  
- ▶ **Nézzünk pár mintapéldát...**

# Linux alapok

---

▶ `ls -l` parancs eredményeként a következőt kapjuk

▶ `-rw-r--r-- 1 bela user 7 Oct 24 19:19 pelda.txt`

▶ Mit jelent ez?

- A. pelda.txt fájlt bela nevű felhasználó írhatja, olvashatja, futtathatja, mindenki más csak olvashatja
- B. pelda.txt fájlt user nevű felhasználó írhatja és olvashatja, a bela csoportba tartozó felhasználók csak olvashatják
- C. pelda.txt fájlt bela nevű felhasználó írhatja és olvashatja, mindenki más csak olvashatja
- D. pelda.txt fájlt bela nevű felhasználó írhatja és olvashatja, a user csoportba tartozó felhasználók csak olvashatják, más nem férhet a fájlhoz

# Linux alapok

---

▶ `ls -l` parancs eredményeként a következőt kapjuk

▶ `-rwxr----- 1 bela user 7 Oct 24 19:19 pelda`

▶ Mit jelent ez?

- A. pelda fájlt bela nevű felhasználó írhatja, olvashatja, futtathatja, mindenki más csak olvashatja
- B. pelda fájlt user nevű felhasználó írhatja, olvashatja, futtathatja, a bela csoportba tartozó felhasználók csak olvashatják, mások nem férhetnek a fájlhoz
- C. pelda fájlt bela nevű felhasználó írhatja, olvashatja, futtathatja, a user csoportba tartozó felhasználók csak olvashatják, mások nem férhetnek a fájlhoz
- D. pelda fájlt bela nevű felhasználó írhatja, olvashatja, futtathatja, a user csoportba tartozó felhasználók csak futtathatják, mások nem férhetnek a fájlhoz

# Linux alapok

---

- ▶ `ls -l` parancs eredményeként a következőt kapjuk
  - ▶ `-rwxr----- 1 bela user 7 Oct 24 19:19 pelda`
- ▶ Melyik parancs nem eredményezi azt, hogy a user csoport végrehajtási jogot kapjon a pelda fájlra?
  - `chmod a+rx pelda`
  - `chmod ug+x pelda`
  - `chmod o+rw pelda`
  - `chmod u+xwr pelda`

# Linux alapok

---

- ▶ A jelszó fájlt szeretnénk a harmadik oszlopa szerint numerikusan csökkenő sorrendbe rendezni. Melyik parancs oldja ezt meg?

- A. `echo /etc/passwd | sort -t: -n -k3 -r`
- B. `cat /etc/passwd | sort -t: -k3 -r`
- C. `cat /etc/passwd | sort -t: -k3`
- D. `cat /etc/passwd | sort -t: -n -k3 -r`

# Linux alapok

---

- ▶ Egy szűrőt szeretnénk írni, ami a standard bemenetét a kimenetre másolja úgy, hogy közben a nagybetűket kisbetűre, míg a kisbetűket nagybetűre cseréli. Melyik parancs oldja ezt meg?

- A. `tr a-z A-Z`
- B. `tr a-zA-Z A-Za-z`
- C. `tr A-Z a-z`
- D. `tr A-Za-z A-Za-z`

# Linux alapok

---

▶ Mit ír ki?

▶ `$ echo "bcacb" | grep "a.*b"`

- A. (semmit)
- B. `acb`
- C. `bcacb`
- D. `bcacbacb`



# Linux alapok

---

▶ Mit ír ki?

▶ `$ echo "baaa" | grep "^a*"`

A. (semmit)

B. `baaa`

C. `aaa`

D. `^a*`

# Linux alapok

---

## ▶ Mit ír ki?

▶ `$ echo "a789b" | sed 's/a\([0-9]*\)b/x\1\1y/'`

- A. (semmit)
- B. **x789y**
- C. **x789789y**
- D. **ax789789yb**

# Linux alapok

---

## ▶ Mit csinál az alábbi shell script?

```
#!/bin/bash
for i in `find`; do
    if [ -h $i ]; then
        target=`ls -l $i | sed 's/.*->./'`
        rm $i
        cp $target $i
    fi
done
```

- A. az adott könyvtár alatt rekurzívan törli az összes szimbolikus linket
- B. csak az aktuális könyvtárban szereplő minden szimbolikus link helyére bemásolja azt a fájlt, amelyre az mutatott
- C. az aktuális könyvtárban rekurzívan minden szimbolikus link helyére bemásolja azt a fájlt, amelyre az mutatott
- D. az adott könyvtár alatt rekurzívan törli az összes szimbolikus linket és ha talál azonos nevű fájlt, akkor azt bemásolja a helyére

# Szoftver szerszámok

---

▶ Mire használható a következő parancs?

▶ `$ traceroute www.bme.hu`

- A. célhozrt elérhetőségének tesztelésére
- B. küldő és fogadó gépek közti egyirányú késleltetés mérésére
- C. küldő és fogadó gépek közti körülfordulási idő mérésére
- D. küldő és fogadó gépek közti útvonal felderítésére

# Szoftver szerszámok

---

▶ Mire használható a következő parancs?

▶ `$ ping www.bme.hu`

- A. célhoszt elérhetőségének tesztelésére
- B. küldő és fogadó gépek közti egyirányú késleltetés mérésére
- C. célhoszt nevének IP címre történő feloldására
- D. küldő és fogadó gépek közti útvonal felderítésére

# Szoftver szerszámok

- ▶ “netstat -aputne” parancs eredményeként a következőt kaptuk:

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State	User	Inode	PID/Program name
tcp	0	0	0.0.0.0:22	0.0.0.0:*	LISTEN	0	30979609	1248/sshd
tcp	0	0	127.0.0.1:80	0.0.0.0:*	LISTEN	0	29630331	25210/apache2

- ▶ Mit jelent?

- A. két kívülről elérhető szolgáltatás fut a gépen: egy secure shell szerver és egy webszerver
- B. egy lokálisan elérhető secure shell szerver és egy kívülről nem elérhető webszerver fut a gépen
- C. egy kívülről elérhető secure shell szerver fut a gépen és egy lokálisan futó alkalmazás kapcsolódik egy távoli webszerverhez
- D. egy kívülről nem elérhető secure shell szerver fut a gépen és egy lokálisan futó alkalmazás kapcsolódik egy távoli webszerverhez

# Szoftver szerszámok

---

▶ Mire való a következő parancs?

▶ `$ tcpdump -i eth0 -ne ip`

- A. eth0 interfészen bejövő minden csomag rögzítése
- B. eth0 interfészen bejövő csak az interfésznek címzett csomagok rögzítése
- C. eth0 interfészen bejövő és kimenő IP csomagok rögzítése
- D. eth0 interfészen bejövő és kimenő nem IP csomagok rögzítése

# Szoftver szerszámok

- ▶ Mit jelent a következő “ifconfig eth1” parancsra adott válasz?

```
eth1      Link encap:Ethernet  HWaddr c4:34:6b:25:58:77
          inet addr:152.66.244.35  Bcast:152.66.244.255  Mask:255.255.255.0
          inet6 addr: fe80::c634:6bff:fe25:5877/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:16969664  errors:0  dropped:56  overruns:0  frame:0
          TX packets:14331565  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:18852095741 (17.5 GiB)  TX bytes:12559977231 (11.6 GiB)
          Interrupt:20 Memory:d0700000-d0720000
```

- A. eth1 interfész inaktív állapotban van
- B. eth1 interfész a 152.66.244.35/24 IPv4 címmel van felkonfigurálva és inaktív állapotban van
- C. eth1 interfész a 152.66.244.35/24 IPv4 címmel van felkonfigurálva és aktív állapotban van
- D. eth1 interfész a 152.66.244.35/16 IPv4 címmel van felkonfigurálva és aktív állapotban van



# Szoftver szerszámok

---

▶ Mire való a következő parancs?

▶ `$ ip addr del 152.66.244.35/16 dev eth1`

- A. eth1 interfész inaktív állapotba kapcsolása
- B. eth1 interfész IPv4 címének beállítása és aktív állapotba kapcsolása
- C. eth1 interfész IPv4 címének törlése
- D. eth1 interfész IPv4 címének törlése és inaktív állapotba kapcsolása

# Hálózati funkciók

---

- ▶ Az alábbiak közül mire használható az iptables filter táblája?
  - A. címfordítás beállítására
  - B. routing beállításokra
  - C. csomagszűrés beállítására
  - D. bejövő forgalom szűrési szabályainak beállítására

# Hálózati funkciók

---

## ▶ Mire való a következő parancs?

```
▶ $ iptables -t nat -A POSTROUTING -s 10.0.0.0/8 -o eth2  
-j SNAT --to-source 192.168.1.10
```

- A. port forwarding beállítása, kívülről hozzáférhetővé válik a 10.0.0.0/8-as tartomány
- B. egy új címfordítási szabály hozzáadása a nat tábla elejéhez, melynek segítségével a 10.0.0.0/8-as belső hálózatról kijutunk a külső hálózatra
- C. egy új címfordítási szabály hozzáadása a nat táblához, ami a 10.0.0.0/8-as címeket cseréli le, ha a csomag az eth2 interfészen érkezett
- D. egy új címfordítási szabály hozzáadása a nat táblához, ami a 10.0.0.0/8-as címeket cseréli le, ha a csomag az eth2 interfészen kerül majd kiküldésre

# Hálózati funkciók

---

## ▶ Mire való a következő parancs?

▶ `$ iptables -A INPUT -p udp --sport 53 -j ACCEPT`

- A. engedélyezi az átmenő UDP forgalmat, ha a forrás port 53
- B. engedélyezi a bejövő UDP forgalmat, ha a forrás vagy cél port 53
- C. engedélyezi a bejövő UDP forgalmat, ha az DNS lekérésre vonatkozik (forrás port 53)
- D. engedélyez bármilyen forgalmat, ha az DNS lekérésre vonatkozik (forrás port 53)

# Hálózati funkciók

- ▶ Mit eredményez a következő konfigurációs beállítás (isc-dhcp-server: dhcpd.conf)?

```
# HeEpUz internal subnet.  
subnet 10.0.0.0 netmask 255.255.255.0 {  
    range 10.0.0.101 10.0.0.150;  
    option domain-name-servers 152.66.115.1, 8.8.8.8;  
    option domain-name "haepuz.hu";  
    option routers 10.0.0.1;  
    option broadcast-address 10.0.0.255;  
    default-lease-time 600;  
    max-lease-time 7200;  
}
```

- A. a kliens gépen a default gateway 10.0.0.1-re lesz beállítva
- B. a kliens gépen az /etc/resolv.conf fájlba vagy a 152.66.115.1 vagy a 8.8.8.8 (csak az egyik) nameserver paraméter íródik be
- C. a kliens gép a 10.0.0.0/24-es címtartományból kap egy tetszőleges dinamikus címet
- D. a kliens gép dinamikusan kap egy IPv4 címet, ehhez a netmask paramétert 255.0.0.0 értékre állítja

# Hálózati funkciók

---

## ▶ Mire való a következő parancs?

▶ `$ dig -t A @8.8.8.8 tmit.bme.hu +norecurse`

- A. lekéri a Google névszerverétől a tmit.bme.hu névhez tartozó IPv4 rekordot; ha nincs információja, üres választ ad
- B. lekéri a Google névszerverétől a tmit.bme.hu névhez tartozó IPv4 rekordot; ha nincs információja, root DNS szerverhez fordul
- C. lekéri a Google névszerverétől a tmit.bme.hu névhez tartozó összes rekordot; ha nincs információja, root DNS szerverhez fordul
- D. lekéri a Google névszerverétől a tmit.bme.hu névhez tartozó összes rekordot; ha nincs információja, üres választ ad

# Routing

---

- ▶ **Mi a különbség a forwarding és routing között? Melyik állítás helyes?**
  
- A. nincs különbség, egymás szinonimái
- B. a forwarding mechanizmus felelős a forgalomirányítási táblák dinamikus feltöltéséért, míg a routing algoritmusok a legrövidebb utak számításáért
- C. a routing algoritmusok útvonalakat számolnak és dinamikusan konfigurálják a forgalomirányítási táblákat, míg a forwarding algoritmusok ezek alapján hoznak döntéseket
- D. a routing algoritmusok dinamikusan konfigurálják a forgalomirányítási táblákat, míg a forwarding algoritmusok valamilyen előre beállított policy szerint útvonalakat számolnak végpontok között

# Routing

---

- ▶ Mi a különbség a link state és distance vector alapú routing protokollok között? Melyik állítás helyes?
  
- A. azonos nézetben dolgoznak, hatékonyságbeli különbség van köztük
- B. a link state alapú megoldások lokális nézetben dolgoznak, ezért jobban skálázódnak, nagyobb hálózatban jobban használhatók
- C. a link state alapú megoldások globális nézetben dolgoznak, így képesek meghatározni a legrövidebb utat bármely két csomópont között, míg a distance vector alapú megoldások lokális nézetben dolgoznak, így nem feltétlenül az optimális útvonalat határozzák meg egyes csomópontok között
- D. a distance vector alapú algoritmusok lokális információk alapján dolgoznak, elosztottan, míg a link state alapú algoritmusok teljes képpel rendelkeznek a hálózatról



# Routing

---

- ▶ Egy hoszt eth0 interfészn a default gateway-t szeretnénk beállítani. Azt tudjuk, hogy a gateway a 192.168.0.0/24-es hálózat első használható IP címén található. Melyik parancs végzi el helyesen a konfigurációt?

- A. `sudo route add default gw 192.168.0.1`
- B. `sudo route add default gw 192.168.0.0`
- C. `sudo route add default gw 192.168.0.1 netmask 255.255.255.0`
- D. `sudo route add -net 0.0.0.0 netmask 0.0.0.0 gw 192.168.0.1 eth0`

# Routing

- ▶ Melyik állítások igazak, ha egy (pl. Quagga) routerben a következő eredményt kapjuk a `show ip route` parancsra?

```
bb1# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route
O    10.0.0.0/24 [110/10] is directly connected, eth0, 00:38:17
C>* 10.0.0.0/24 is directly connected, eth0
O>* 10.0.1.0/24 [110/20] via 10.0.0.2, eth0, 00:37:27
O>* 10.0.2.0/24 [110/30] via 10.0.0.2, eth0, 00:37:23
O    10.0.3.0/24 [110/40] via 10.0.0.2, eth0, 00:37:19
C>* 10.0.3.0/24 is directly connected, eth1
C>* 127.0.0.0/8 is directly connected, lo
```

- A. 10.0.3.0/24 hálózatot eth0 és eth1 interfészen keresztül is eléri, jelenleg a közvetlen kapcsolatot használja
- B. 10.0.3.0/24 hálózatot 40-es költségű OSPF úton ér(het)í el
- C. 10.0.2.0/24 hálózatot nem éri el közvetlenül
- D. 10.0.1.0/24 hálózatot a 10.0.1.1 gateway-en keresztül éri el

# Routing

- ▶ Egy hoszt routing táblájában az alábbi három bejegyzés szerepel. Ezek alapján mely állítások igazak?

```
$ route -n
```

```
Kernel IP routing table
```

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	192.168.77.1	0.0.0.0	UG	600	0	0	eth0
192.168.77.0	0.0.0.0	255.255.255.0	U	600	0	0	eth0
192.168.0.0	0.0.0.0	255.255.255.128	U	600	0	0	eth1

- A. A hoszt minden forgalmat a 192.168.77.1 felé küld.
- B. A hoszt a 192.168.77.16-nak címzett forgalmat az eth0 interfészén küldi ki.
- C. A hoszt a 192.168.0.200-nak címzett forgalmat az eth1 interfészén küldi ki.
- D. A hoszt a 192.168.0.200-nak címzett forgalmat a default gateway felé küldi.

# Python

---

- ▶ Mit ír ki az alábbiak közül a Python interpreter a következő utasításra?

```
2 * "11" + "2"
```

- A. `TypeError: cannot concatenate 'str' and 'int' objects`
- B. `24`
- C. `1113`
- D. `"11112"`

# Python

---

- ▶ Érvényes-e a következő utasítás a Python nyelvben?

```
["12" , 12]
```

- ▶ Nem, az interpreter kivételt dob: `SyntaxError: invalid syntax`
- ▶ Nem, Python nyelvben csak azonos típusú értékek tárolhatóak egy listában
- ▶ Igen, Python nyelvben a lista bármilyen típusú értékeket tárolhat
- ▶ Igen, de az azonos értékeket csak egyszer tárolódnak le: `["12"]`

# Python

---

- ▶ Mit ír ki a következő Python függvény az alábbi paraméter listával: `magic_function(0, c=1, b=2)`?

```
def magic_function(a, b=3, *args, **kwargs):  
    print a + b
```

- A. 1
- B. 2
- C. 3
- D. **NameError: name 'c' is not defined**