

Linux hálózat beállítás

Varga Tibor

Hálózati kártya

- Minden hardver, így a hálózati kártyák is egy fájlra vannak leképezve a dev könyvtárban. Debian 9 előtt ezek elnevezése eth0, eth1, stb. A Debian 9 megjelenése óta az elnevezési rendszer megváltozott.
- A **net-tools** csomag Debian 9-ben alapértelmezetten már nincs telepítve, így az ifconfig és más parancsok, csak ezen csomag telepítésével használhatók. A net-tools csomagot az iproute2 csomag váltja, amely alapértelmezetten telepítve van.

Ethernet csatolók azonosítása

Az elérhető Ethernet csatolók gyors azonosítására az **ifconfig** parancsot használható, a következő módon:

- **# ifconfig -a | grep eth**

```
tibi@server:~/test$ ifconfig -a | grep eth
eth0 08:00:27:40:59:47 txqueuelen 1000 (Ethernet)
```

Az lspci parancs megjeleníti a kiszolgálón lévő összes PCI busz információit.

A buszra vonatkozó információk mellett a PCI és PCIe buszhoz csatlakoztatott összes hardver eszközről is információkat fog megjeleníteni. Például információkat jelenít meg az Ethernet kártyákról, RAID vezérlőkről, videokártyákról stb.

- Ethernet csatolók keresése: **\$ lspci -D | egrep -i "Network|Ethernet"**

```
tibi@server:~/test$ lspci -D | egrep -i "network|eth"
0000:00:03.0 Ethernet controller: Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
```

Ethernet csatolók azonosítása

A rendszeren elérhető összes hálózati csatoló azonosítására használható az `lshw` parancs is.

- **# `lshw -class network`**

```
tibi@server:~/test$ sudo lshw -class network
[sudo] tibi jelszava:
*-network
    description: Ethernet interface
    product: 82540EM Gigabit Ethernet Controller
    vendor: Intel Corporation
    physical id: 3
    bus info: pci@0000:00:03.0
    logical name: enp0s3
    version: 02
    serial: 08:00:27:40:59:47
    size: 1Gbit/s
    capacity: 1Gbit/s
    width: 32 bits
    clock: 66MHz
    capabilities: pm pcix bus_master cap_list ethernet physical tp 10bt 10bt-fd 100bt 100bt-fd 1000bt-fd autonegotiation
    configuration: autonegotiation=on broadcast=yes driver=e1000 driverversion=7.3.21-k8-NAPI duplex=full ip=192.168.0.108 latency=64 link=yes mingnt=255 multicast=yes port=twisted pair speed=1Gbit/s
    resources: irq:19 memory:f8200000-f821ffff ioport:d020(size=8)
```

Net-tools és Iproute2

- A net-tools csomag Debian 9-ben alapértelmezetten már nincs telepítve, így az ifconfig és más parancsok, csak ezen csomag telepítésével használhatók. A net-tools csomagot az iproute2 csomag váltja, amely alapértelmezetten telepítve van.

Alap parancsok

- **ifconfig -a**
- **ip a**

Az ifconfig

A hálózati kártyákról az **ifconfig** parancs is ad információt. De használata elavultnak számít, bár igen sokan használják.

- \$ ifconfig

```
tibi@server:~/test$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.0.108  netmask 255.255.255.0  broadcast 192.168.0.255
    inet6 2a01:36d:2800:61e8:a00:27ff:fe40:5947  prefixlen 64  scopeid 0x0<global
>
    inet6 fe80::a00:27ff:fe40:5947  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:40:59:47  txqueuelen 1000  (Ethernet)
    RX packets 50119  bytes 65988352 (65.9 MB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 6321  bytes 649555 (649.5 KB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 225  bytes 19736 (19.7 KB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 225  bytes 19736 (19.7 KB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

ifconfig

A kimenetben az alábbi táblázatból láthatunk, esetlegesen kulcsszavakat:

UP	a hálózati kártya aktiválva van
BROADCAST	képes broadcast üzeneteket küldeni
RUNNING	A szükséges erőforrások le vannak foglalva
ALLMULTICAST	az összes multicast üzenet elfogadása
PROMISC	az eszköz minden forgalmat elfogad
MULTICAST	küldhet és fogadhat multicast üzeneteket
MTU	Az maximálisan átvihető csomagméret

ifconfig parancs

Az ifconfig parancs segítségével megtudhatjuk a hálózati kártya IP címét is:

- **\$ ifconfig eth0**

Egy hálózati kártyára több IP címet is felhúzzhatunk a következő módon:

- **\$ ifconfig eth0:0 192.168.6.1 netmask 255.255.255.0**

Az IP cím állandósítása:

- **\$ iface eth0:0 inet static**
address 192.168.6.1
netmask 255.255.255.0

A net-tools és a iproute2

Az **net-tools** a régi Linux kernelhez készült hálózati eszköz gyűjtemény. Az újabb kernel újabb lehetőségeihez az **iproute2** eszközugyűjtemény használandó. Most összevetjük a két csomag utasításait.

Cél	Net-tools	Iproute2
cím és kapcsolat (link) beállítás	ifconfig	ip add, ip link
irányítótábla	route	ip route
szomszédok (neighbors)	arp	ip neigh
vlan	vconfig	ip link
alagút (tunnels)	iptunnel	ip tunnel
multicast	ipmaddr	ip maddr
statisztika	netstat	ss
segítség	ifconfig --help	ip help
interfészek státusza	ifconfig -s netstat -i	ip -s link

Példák a konfigurálás parancsokhoz

net-tools parancs

iproute2 parancs

Minden kapcsolt megjelenítése a hálózati interfészeken:

`ifconfig -a`

`ip link show`

Az interfész aktiválása

`ifconfig eth0 up`

`ip link set up eth0`

Az interfész deaktiválása:

`ifconfig eth0 down`

`ip link set down eth0`

IPv4-es cím interfészhez rendelése:

`ifconfig eth0 192.168.8.1/24`

`ip addr add 192.168.8.1/24 dev eth0`

Több IP cím:

`ifconfig eth0:0 192.168.8.2 up`

`ip addr add 192.168.8.2/24 dev eth0`

IPv4-es cím törlése interfészen:

`ifconfig eth0 0`

`ip addr del 192.168.8.2/24 dev eth0`

A route parancs és irányítótábla

net-tools parancs

iproute2 parancs

Irányítótábla megjelenítése:

`route -n ; route -rn`

`ip route show`

Alapértelmezett átjáró hozzáadása:

`route add default gw 192.168.10.1 eth0`

`ip route add default via 192.168.10.1 dev eth0`

Alapértelmezett átjáró törlése:

`route del default`

`ip route del default`

Az ip parancs itt kivételes, mert cserét is lehetővé tesz:

`ip route replace default via 192.168.10.1`

Statikusan hálózat hozzáadása:

`route add -net 192.168.20.0/24 gw 10.0.0.1 dev
eth0`

`ip route add 192.168.20.0/24 via 10.0.0.1 dev
eth0`

Statikusan megadott hálózat törlése:

`route del -net 192.168.20.0/24`

`ip route del 192.168.20.0/24`

A statisztika

net-tools parancs

iproute2 parancs

Foglalat/Sokcet statisztika megtekintése:

`netstat -l`

`ss -l`

ARP tábla megtekintése:

`arp -an`

`ip neigh`

Statikus ARP bejegyzés hozzáadása:

`arp -s 192.168.5.100 00:3c:54:29:ac:23`

`ip neigh add 192.168.5.100 lladdr
00:3c:54:29:ac:23 dev eth0`

Statikus ARP bejegyzés törlése:

`arp -d 192.168.5.100`

`ip neigh del 192.168.5.100 dev eth0`

IPv6-os cím megtekintése:

`ifconfig eth0`

`ip -6 addr show dev eth0`

Hálózat beállítások tartósan

DEBIAN esetén beállítások következő fájlokban tarolódnak amely szerkeszthető :

- **nano etc/network/interfaces** Kiíratás után:

```
allow-hotplug enp1s0
iface enp1s0 inet static
address 192.168.5.6
netmask 255.255.255.0
network 192.168.5.0
broadcast 192.168.5.255
gateway 192.168.5.1
```

Ezek után a felhúzzuk a hálózati kártyát: **\$ ifup enp1s0**

A hálózati kártya leállítása: **\$ ifdown enp1s0**

Hálózat beállítások tartósan

Az **Ubuntu Server 18.04** alapértelmezetten netplan-t használja.

A netplan beállítható Debian GNU/Linux rendszeren is:

- **apt install netplan.io**

Beállítás:

- Beállító fájlt generálni kell

\$ sudo generate netplan

- Ellenőrizzük létrejött a beállítást rögzítő fájl

\$ ls /etc/netplan/*.yaml

50-cloud-init.yaml

- Szerkesztés:

\$ sudo nano /etc/netplan/50-cloud-init.yaml

- Végül alkalmazzuk a beállításokat:

\$ netplan apply

```
network:
  ethernets:
    enp0s3:
      addresses: []
      dhcp: true
  version: 2
```

```
network:
  ethernets:
    enp0s3:
      addresses: [ip/maszk]
      gateway4: ip
      nameservers:
        addresses: [8.8.8.8,8.8.4.4]
      dhcp4: no
  version: 2
```

Hálózat beállítások tartósan

Az **Ubuntu Desktop 18.04** alapértelmezetten netplan-t használja.

A netplan beállítható Debian GNU/Linux rendszeren is:

- **apt install netplan.io**

Beállítás:

- Beállító fájlt generálni kell

\$ sudo generate netplan

- Ellenőrizzük létrejött a beállítást rögzítő fájl

\$ ls /etc/netplan/*.yaml

01-network-manager-all.yaml

vagy 01-netcfg.yaml

- Szerkesztés:

\$ sudo nano /etc/netplan/01-netcfg.yaml

- Végül alkalmazzuk a beállításokat:

\$ netplan apply

```
# This file describes the network interfaces
# For more information, see netplan(5).
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: yes
    enp0s8:
      dhcp4: no
      dhcp6: no
      addresses: [192.168.56.110/24, ]
      gateway4: 192.168.56.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
```

Hosznév

A hosznév tulajdonképpen a **számítógép neve**.

A gépnév lekérdezése: **\$ hostnamectl**

A gépnév beállítása: **\$ hostnamectl set-hostname gepnev**

```
tibi@server:/etc/netplan$ hostnamectl
  Static hostname: server
        Icon name: computer-vm
        Chassis: vm
        Machine ID: 323531196e5b4e37b8ed64933da18c9c
        Boot ID: f7bea6395c0d4e239ec6162851a058f5
  Virtualization: oracle
  Operating System: Ubuntu 18.04.4 LTS
        Kernel: Linux 4.15.0-76-generic
  Architecture: x86-64
tibi@server:/etc/netplan$ hostnamectl set-hostname tibi-server
tibi@server:/etc/netplan$ hostnamectl
  Static hostname: tibi-server
        Icon name: computer-vm
        Chassis: vm
        Machine ID: 323531196e5b4e37b8ed64933da18c9c
        Boot ID: f7bea6395c0d4e239ec6162851a058f5
  Virtualization: oracle
  Operating System: Ubuntu 18.04.4 LTS
        Kernel: Linux 4.15.0-76-generic
  Architecture: x86-64
```

Ez a beállítás tartós, az `/etc/hostname` fájl tartalmát is cseréli.

A traceroute

A traceroute egy számítógép-hálózati diagnosztikai eszköz, az internetprotokoll (IP) hálózaton áthaladó Csomagok útvonalának (path) meghatározására, és az átviteli késleltetés mérésére. Linux esetén telepíteni kell **\$ sudo apt install traceroute**

A traceroute a legtöbb operációs rendszeren elérhető.(A Microsoft Windows operációs rendszereken tracert-nek nevezik.)

```
tibi@server:/etc/netplan$ traceroute -F google.com 100
traceroute to google.com (172.217.19.110), 30 hops max, 100 byte packets
 1  _gateway (192.168.0.1)  1.700 ms  1.519 ms  1.371 ms
 2  pppoe.peer.digicable.hu (10.0.0.1)  1.957 ms  1.790 ms  1.551 ms
 3  * * *
 4  et-10.bb01.dunaujvaros.digicable.hu (94.21.3.48)  6.561 ms  6.382 ms  6.190 ms
 5  * * *
 6  * * *
 7  te-0-2-0-4.xr01.budapest.digicable.hu (94.21.3.89)  7.920 ms  te-0-3-0-1.xr01.budapest.
  digicable.hu (78.131.3.113)  7.419 ms  6.930 ms
 8  as15169.peering.digicable.hu (94.21.255.2)  13.186 ms  12.547 ms  12.117 ms
 9  74.125.242.241 (74.125.242.241)  11.634 ms  7.219 ms  6.160 ms
10  209.85.255.243 (209.85.255.243)  7.362 ms  6.761 ms  6.796 ms
11  bud02s27-in-f14.1e100.net (172.217.19.110)  6.004 ms  6.203 ms  5.957 ms
```

Telnet

- A Telnet lényege, hogy a saját számítógépéről be tud jelentkezni egy másik (mindegy, hogy a világ melyik részén lévő) számítógépre. Az FTP-vel és a Gopher-rel csak az ott lévő adatokat érte el, Telnet esetében programokat is futtathat a távoli (remote) gépen.
- **A Telnet protokoll célja egy általánosan elérhető, kétirányú, nyolcbites byte-alapú kommunikációs rendszer biztosítása. Egyaránt használható két terminál közötti (linking), illetve processzek közötti kommunikációra. TCP alapon működik.**
- Ma már a telnet-alapú terminálhasználat meglehetősen kevésbé elterjedt, lévén a **telnetben nincs semmiféle titkosítás**, ezért általában az SSH-t használják helyette. Ugyanakkor szinte az összes Linux-disztribúció alapból telepíti a telnet-klienst, hiszen rengeteg egyéb protokollt lehet vele kényelmesen debuggolni, illetve "kézzel" irányítani: például HTTP, POP3, SMTP.

SSH

- A Secure Shell (röviden: SSH) egy szabványcsalád, és egyben egy protokoll is, amit egy **helyi és egy távoli számítógép közötti biztonságos csatorna kiépítésére** fejlesztettek ki. Nyilvános kulcsú titkosítást használ a távoli számítógép hitelesítésére, és opcionálisan a távoli számítógép is hitelesítheti a felhasználót.
- Az SSH-t leggyakrabban arra használják, hogy egy távoli gépre belépjenek vele és parancsokat adjanak ki, de támogatja a tunnelinget, azaz tetszőleges TCP portok és X11 kapcsolatok továbbítását; fájlok biztonságos átvitelére is használható a kapcsolódó SFTP (Secure FTP) és SCP (Secure Copy) protokollok segítségével.
- Az SSH szerverek alapértelmezésben a 22-es TCP portot figyelik.