

# GNU/Linux

## IV. kiadás

Ez a dokumentum szabad szoftver, szabadon terjeszthető és/vagy módosítható a **GNU Free Documentation License**-ben leírtak szerint.

Készítette: Fuszenecker Róbert <[hg8lhs@gmail.com](mailto:hg8lhs@gmail.com)> 2008-ban.

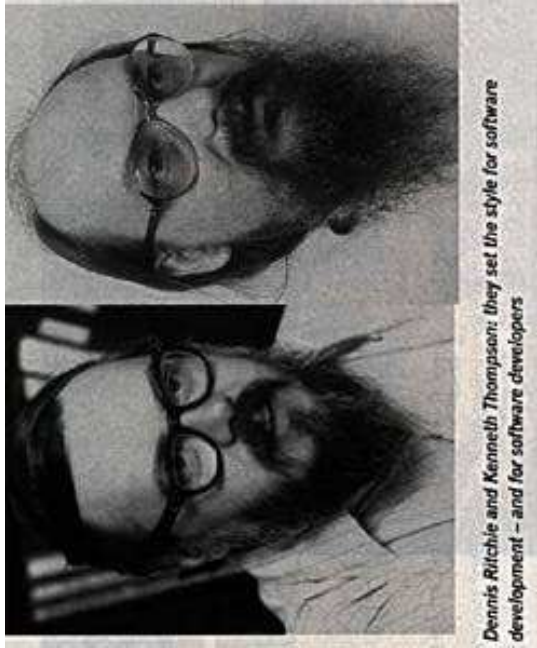
## Történeti áttekintés

1969-től napjainkig

- 1964, MIT intézet: Multics projekt GE-645 gépen
- többfelhasználós, időosztásos operációs rendszer
- B2 biztonsági minősítés (1985), ACL, SMP, relációs adatbázisok kezelése
- 1969, Ken Tompson: a „Space travel” nevű játék PDP-7-en
- nem multiuser, nem multitask, de megy
- UNICS (később UNIX)
- 1971 (?): PDP-11
- portolni szeretnék volna a UNICS-ot, de az assembly kód nem hordozható, ezért...
- megszületik a C nyelv: Dennis M. Ritchie, 1973



PDP-7



Dennis Ritchie és Ken Thompson

- 1974: a Berkeley egyetem oktatási célra másolatot kap
- 1977: kiadja a saját disztibúcióját **BSD** néven, ami
  - Berkeley Softver Distribution (régebben)
  - Berkeley Software Design (manapság)
- Ma is létező „kiadások”:
  - FreeBSD: modern technológiák, új fejlesztések
  - OpenBSD: nyíltság, szabványosság, biztonság
  - NetBSD: hordozhatóság (53 architektúra)
  - DesktopBSD, PC-BSD, DragonflyBSD: asztali felhasználásra
- A UNIX **de facto szabvány** lett, „ipari” implementációk
  - Apple: A/UX
  - IBM: AIX

- Berkeley Egyetem: BSD/OS
- Szovjetúnió: DEMOS (ДЕМОС)**
- Hewlett-Packard (HP): HP-UX
- Silicon Graphics, Inc. (SGI): IRIX
- Apple: MacOS X
- Waterloo Egyetem: QNX
- Sun Microsystems: Solaris, SunOS
- Microsoft: XENIX**
- SCO Group: SCO UNIX
- Novell: UnixWare
- Siemens: SINIX
- Andrew S. Tanenbaum: MINIX
- Linus Torvalds és a többiek: Linux

Történeti áttekintés

**1985: POSIX szabvány** – IEEE 1003 / ISO 9945

POSIX = **P**ortable **O**perating **S**ystem **I**nterface

1985, **Richard M. Stallman**: „Szabad szoftver kiáltvány”: készüljön egy teljesen szabadon használható operációs rendszert, ami olyan, mint a UNIX, de nem UNIX → a GNU nem UNIX („**GNU** is **not UNIX**”)

1987, **Andrew S. Tannenbaum**, Vrije Universiteit, Amsterdam: megszűntek a MINIX (**minimal UNIX**) 1990-es évek

**Linus Torvalds**, egy finn egyetemista kísérletezik az i386-os processzorral és a Minix-szel

Eredményeit közzéteszi az interneten: akkoriban még a comp.os.minix hírcsoportban (newsgroup)


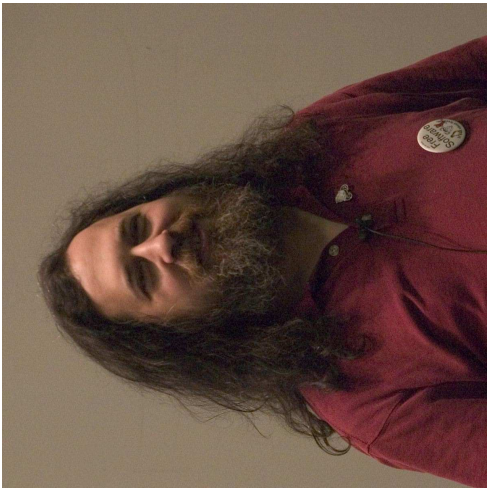
A GNU projekt és a Linux (Linus' UNIX) egymásra talál → megszűntek a (POSIX kompatibilis) **GNU/Linux**

Richard M. Stallman

Andrew S. Tannenbaum prof.

8 / 50

Történeti áttekintés





Richard M. Stallman

Andrew S. Tannenbaum prof.

8 / 50

Történeti áttekintés



Linus Torvalds

9 / 50

Történeti áttekintés

Az **internetnek** köszönhetően hihetetlenül gyorsan fejlődik és terjed. A linux-terjesztések (disztribúciók) száma közel 600 (nagy részüket már senki sem használja, pl. egyflopyps terjesztések)

Ma a UNIX jogok az **Open Group**nál vannak

a **POSIX szabvány** – IEEE 1003 / ISO 9945 – megvásárolható, letölthető

10 / 50

A UNIX alapfilozófiája

## A UNIX alapfilozófiája

Legyen **minden erőforrás fájl** (ID-vel, azonosítóval megcímezhető adatfolyam (stream))

eszközök (pl. hangkártya, nyomtató, merevlemez)

hálózati kapcsolat (socketek)

rendezett adathalmazok (képek, hangok, szöveges fájlok), stb.

egységesen kezelhetők legyenek az erőforrások (open(2), read(2), write(2), close(2), ioctl(2), seek(2), mmap(2))

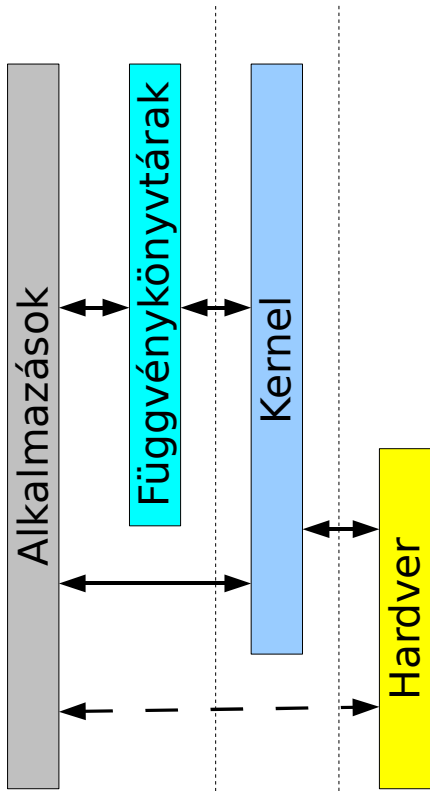
az operációs rendszer és a programok hordozhatósága (POSIX szabványnak (is) köszönhetően)

„villamosmérnökök írták saját maguknak”

11 / 50

Az operációs rendszer felépítése

## Az operációs rendszer felépítése



The diagram illustrates the layers of an operating system. At the top is 'Alkalmazások' (Applications). Below it is 'Függvénykönyvtárak' (Libraries), which is highlighted in cyan. Below that is the 'Kernel', highlighted in blue. At the bottom is 'Hardver' (Hardware), highlighted in yellow. Arrows indicate the flow of data and control between these layers: applications use libraries, libraries use the kernel, and the kernel interacts with the hardware. There are also vertical dashed lines separating the layers.

12 / 50

<div> <div>Az operációs rendszer felépítése</div> <div> <p><b>a hardver:</b> a számítógép fizikailag megérintható elemei</p> <p><b>kernel:</b> az operációs rendszer magja; feladatai (a teljesség igénye nélkül):</p> <ul style="list-style-type: none"> <li>elrejtteni a hardver sajátosságait; kezelni az eszközöket (definiált interfészeken keresztül)</li> <li>processzor- és memóriakezelés</li> <li> folyamatok irányítása: processzek indítása, ütemezése, leállítása, kivételek kezelése</li> <li>inter-process (folyamatok közötti) kommunikáció</li> <li>közös erőforrások kezelése (kölcsonös kizárás)</li> <li>„extra” szolgáltatások: <ul style="list-style-type: none"> <li>fájlrendszerek</li> <li>hálózat</li> <li>felhasználók, csoportok, jogosultságok, stb.</li> </ul> </li> </ul> </div> </div> <div> <div>13 / 50</div> </div>
--

<div> <div>Az operációs rendszer felépítése</div> <div> <p><b>függvénykönyvtárak:</b> magasabb szintű, összetett műveletek, melyeket a felhasználói programok gyakran használnak, például:</p> <ul style="list-style-type: none"> <li>magas szintű fájlkezelés: <ul style="list-style-type: none"> <li><i>printf(3)</i> vs. <i>write(2)</i>, <i>fscanf(3)</i> vs. <i>read(2)</i></li> </ul> </li> <li>sztringek, asszociatív tömbök</li> <li>matematikai műveletek, speciális algoritmusok (pl. FFT)</li> <li>speciális fájlformátumok (képek, hangok, tömörítési eljárások, stb.)</li> </ul> <p><b>a függvénykönyvtárak előnye,</b> hogy lehetővé teszik a kód-újrahasznosítást: amit más megírt, azt nem kell megírnom a programomban</p> </div> </div> <div> <div>14 / 50</div> </div>
---

<div> <div>Az operációs rendszer felépítése</div> <div> <p>memóriatakarékosság: ha egy függvénykönyvtár már be van töltve a memóriába, nem kell egy másik folyamat számára újra betölteni annak indításakor</p> <p><b>alkalmazások:</b></p> <p>definíció: folyamat (process): a lemezen tárolt program egy elindított változata (PID azonosítóval)</p> <p><b>előtérben futó folyamat:</b> általában használják a standard inputot, illetve outputot</p> <p><b>háttérben futó folyamat:</b> nem használják a standard inputot, illetve outputot</p> <p><b>démon:</b> olyan háttér folyamat, amely valamilyen erőforrás kezelésért (multiplexelés) felel</p> <p><b>zombi:</b> olyan folyamat, ami valami miatt „megdöglött”, időnként az operációs rendszer kitakarítja a memóriából</p> </div> </div> <div> <div>15 / 50</div> </div>
--

<div> <div>Inter-processz kommunikáció (IPC)</div> <div> <h2>Inter-processz kommunikáció (IPC)</h2> <pre> graph TD     A1[Alkalmazás 1] &lt;--&gt; L1[Lib-ek]     L1 &lt;--&gt; K[Kernel]     K &lt;--&gt; H[Hardver]     A2[Alkalmazás 2] &lt;--&gt; L2[Lib-ek]     L2 &lt;--&gt; K     A1 &lt;--&gt; IPC  A2     </pre> </div> </div> <div> <div>16 / 50</div> </div>
---

<div> <div>Inter-processz kommunikáció (IPC)</div> <div> <p><b>feladata:</b> a folyamatok közötti adatcsere megvalósítása; olyankor alkalmazzuk, ha egy erőforráshoz több folyamat szeretne hozzáférni → multiplexelés; például:</p> <ul style="list-style-type: none"> <li>nyomtatás, adatbázis-kezelés, hang, grafikus felhasználói felület</li> </ul> <p><b>megoldások:</b></p> <ul style="list-style-type: none"> <li><b>osztott memória, szemaforok/mutexek</b> <ul style="list-style-type: none"> <li>előnye: gyors</li> <li>tipikusan a grafikus felület (X szerver) használja</li> </ul> </li> <li><b>fifo-k</b> (pipe-ok és UNIX domain socketek) <ul style="list-style-type: none"> <li>előnye: jól illeszkedik a UNIX filozófiához → minden dolog fájlnak tekintendő; kicsit lassabb, mint az osztott memória</li> </ul> </li> <li><b>hálózat</b> (jellemzően TCP/IP) <ul style="list-style-type: none"> <li>előnye: a processzeknek nem kell ugyanazon a gépen futniuk → <b>kliens-szerver modell</b> alakítható ki</li> </ul> </li> </ul> </div> </div> <div> <div>17 / 50</div> </div>
--

<div> <div>IPC megvalósítása hálózaton</div> <div> <h2>IPC megvalósítása hálózaton</h2> <pre> graph TD     A1[Alkalmazás 1] &lt;--&gt; L1[Lib-ek 1]     L1 &lt;--&gt; K1[Kernel 1]     K1 &lt;--&gt; H1[Hardver 1]     A2[Alkalmazás 2] &lt;--&gt; L2[Lib-ek 2]     L2 &lt;--&gt; K2[Kernel 2]     K2 &lt;--&gt; H2[Hardver 2]     A1 &lt;--&gt; IPC  A2     </pre> </div> </div> <div> <div>18 / 50</div> </div>
---



## Hálózatok

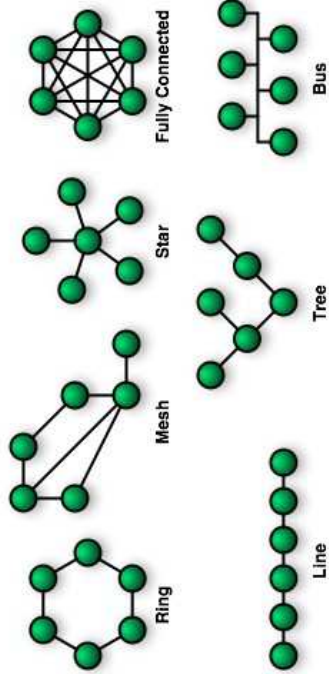
Számítógépek összekapcsolása információcsere céljából

Fogalmak (előző félévekből ismerősek):

- hálózati topológiák
- csomópont (node)
- ismétlő (repeater)
- hub
- switch
- híd (bridge)
- átjáró (gateway)
- útvonalválasztó (router)

Hálózati topológiák (rövid összefoglalás, ismétlés, példák):

- busz (pl. token-bus, 10Mbps ethernet)
- gyűrű (pl. token-ring)
- csillag (pl. 100/1000 Mbps Ethernet)
- ad-hoc (pl. wifi)



## Hálózati protokollok

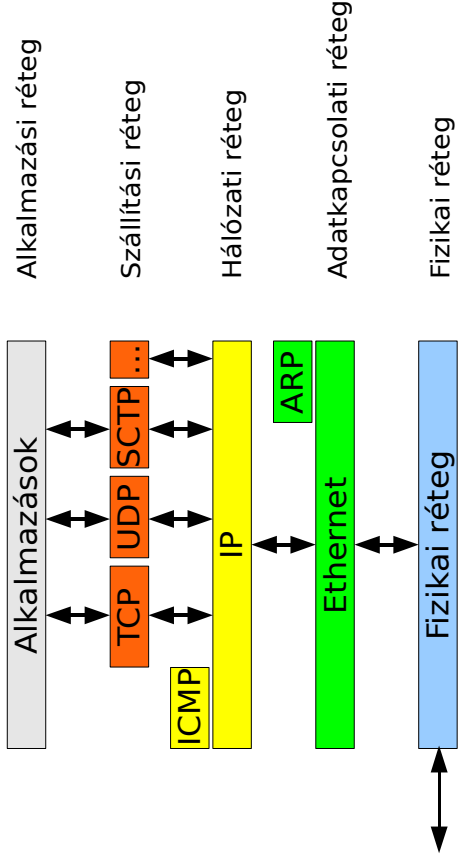
„Protokoll: az OSI modell **azonos szinten** elhelyezkedő rétegei közötti kommunikáció szabályrendszerét írja le.”

(Sándor Tamás)

Ebben a részben a következő protokollokkal ismerkedünk meg:

- Ethernet (IEEE 802.3)
- ARP (RFC 826)
- IP v4 és v6 (RFC 791 és 2460)
- ICMP (RFC 792)
- TCP (RFC 793)
- UDP (RFC 768)

## Hálózati rétegek



## Hálózati rétegek feladatai

**Fizikai réteg:** megteremti a „bitszintű” kapcsolatot a hálózat elemei (csomópontok) között

**Adatkapcsolati réteg** (jelen esetben Ethernet):

- keretszintű kommunikációt tesz lehetővé, a node-okat fizikai címmel (MAC-címmel) azonosítja
- az ARP (Address Resolution Protocol) segítségével az IP-címeket MAC-címeké konvertálja

**Hálózati réteg:** a csomagok egyik csomóponttól a másik csomópontig juttatását irányítja

- az IP csomag tartalmazza a forrás gép és a cél gép IP-címét, ami alapján az útvonalválasztók eljuttatják a csomagot a megfelelő hálótatba

az ICMP (Internet Control Message Protocol) üzenetek a hálózat állapotáról adnak információt

- ICMP echo request → ping
- csomópont nem található (host not found)
- hálózat nem található (network not found), stb.

### Szállítási réteg:

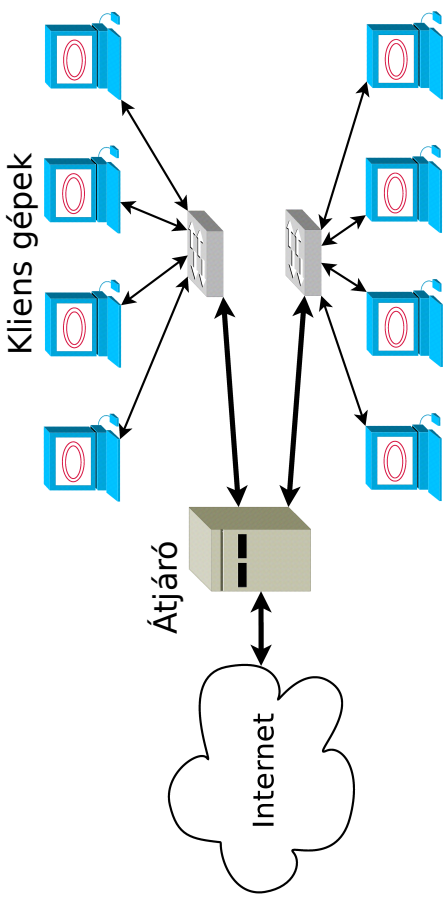
**TCP (Transmission Control Protocol):** az adatfolyam darabokra tördelését végzi, majd a célgépen (sorrendhelyesen és hibamentesen) visszaállítja az adatfolyamot, így egyfajta „soros bájtvitel” tesz lehetővé (stream)

megbízható, sorrendhelyes, nyugtázott átvitel

a **TCP portcím** segítségével megkülönböztethetők,

Hálózati rétegek feladatai
<p>megcímezhetők azon folyamatok (gyakran démonok, rendszerfolyamatok), melyek „regisztrálták” magukat, vagyis megnyitottak egy TCP portot</p> <p><b>UDP (User Datagram Protocol)</b>: lehetővé teszi, hogy a folyamatok üzeneteket küldjenek egymásnak; az üzenetek épsége és sorrendje nem garantált, mert nincs nyugtázás</p> <p>az UDP is használ portcímeket, melyek függetlenek a TCP portoktól</p> <p>tipikusan real-time alkalmazásokban (Voice-over-IP)</p> <p><b>SCTP (Stream Control Transport Protocol)</b>: multimédia tartalmak blokkos átvitelét (SCTP üzenetek) teszik lehetővé; egy SCTP üzenet több megabájt méretű is lehet</p> <p>előnye a TCP-hez képest, hogy torlódás esetén előbbja a régóta várakozó üzeneteket, így utat enged</p>
25 / 50

Hálózati rétegek feladatai
<p>az újabbaknak („legyen inkább kis hiba a hangban/képben, de ne késsen, és élvezhető maradjon”)</p> <p>üzenetalapú, megbízható, sorrendhelyes, nyugtázott</p> <p><b>Egyéb</b>: tipikusan routing protokollok, melyek a hálózat útvonalválasztóinak működését befolyásolják</p> <p><b>Alkalmazási réteg</b>: az alkalmazások egymás között megvalósuló protokolljai, pl.:</p> <p>FTP (File Transfer Protocol): fájlok átvitelére</p> <p>TFTP (Trivial File Transfer Protocol): boot-olás során, fájlok átvitelére (egyszerűsített ftp)</p> <p>POP (Post Office Protocol): levelek letöltése a szerverről</p> <p>stb.</p>
26 / 50

Hálózatok kialakítása
<h2>Hálózatok kialakítása</h2> <p>A következő ábra példát mutat egy helyi hálózat kialakítására:</p> 
27 / 50

Hálózatok kialakítása
<p>A fenti hálózat a következő összetevőket tartalmazza:</p> <p>kliens számítógépek, két alhálózatba szervezve (a hálózati ábrájó más-más interfészére „csatlakoznak”)</p> <p>switch-ek: ezek biztosítják a fizikai kapcsolatot a hálózati csomópontok között</p> <p>átjáró (gateway)</p> <p>külső, publikus hálózat, „Internet”</p>
28 / 50

A hálózat elemeinek paraméterei
<h2>A hálózat elemeinek paraméterei</h2> <p><b>MAC-cím</b>: 48 bit hosszú, a hálózati interfész egyedi azonosítója, elvileg nem változtatható meg, a gyártó programozza a hardverbe</p> <p>Létezik broadcast (mindenkit megszólító) MAC-cím is: FF:FF:FF:FF:FF:FF</p> <p>A hálózati kártyák „egymás között” Ethernet protokollt (Layer 2) használnak, ezért az IP-címeket MAC címmé kell konvertálni. Erre szolgál az ARP protokoll.</p> <p><b>IP-cím</b>: a hálózati interfész Layer-3 címe, ez alapján történik az IP keretek célba juttatása két végpont (pl. számítógép, intelligens nyomtató, router) között</p>
29 / 50

A hálózat elemeinek paraméterei
<p>IPv4 esetén 32 bit hosszú:</p> <p>4x8 bites szakaszok decimális formátumban, ponttal („.”) elválasztva, pl.: 192.168.0.34</p> <p>IPv6 esetén 128 bit: 8x16 bites szakaszok hexadecimális formátumban, kettősponttal („:”) elválasztva</p> <p>pl.</p> <p>fe80:0000:0000:0000:0000:0000:0000:0001</p> <p>IPv6 címek tömörebb írásmódja:</p> <p>a bevezető „0”-kat nem kell kiírni:</p> <p>pl: fe80::0012 → fe80::12</p> <p>azon szakaszok, melyek csak 0-t tartalmaznak, rövidíthetők „::”-tal:</p>
30 / 50

A hálózat elemeinek paraméterei													
<p>pl. fe80:0000:0000:0000:0000:0000:0000:0001  → fe80:0:0:0:0:0:0:1 → fe80::1</p> <p><b>hálózati maszk</b> (IPv4) vagy <b>prefix-hossz</b> (IPv6):  azt határozza meg, hogy az IP-cím elején hány bit  azonosítja a hálózatot, és az IP-cím végén hány bit  azonosítja a hálózati eszközt</p> <p>Példa #1:</p> <table> <tr> <td>Maszk:</td><td>255.255.255.0 <small>(első 24 bit 1-es értékű)</small></td></tr> <tr> <td>Hálózat címe:</td><td>10.0.3.0</td></tr> <tr> <td>Node 1:</td><td>10.0.3.1</td></tr> <tr> <td>Node 2:</td><td>10.0.3.2</td></tr> <tr> <td>...</td><td></td></tr> <tr> <td>Node 254:</td><td>10.0.3.254</td></tr> </table>		Maszk:	255.255.255.0 <small>(első 24 bit 1-es értékű)</small>	Hálózat címe:	10.0.3.0	Node 1:	10.0.3.1	Node 2:	10.0.3.2	...		Node 254:	10.0.3.254
Maszk:	255.255.255.0 <small>(első 24 bit 1-es értékű)</small>												
Hálózat címe:	10.0.3.0												
Node 1:	10.0.3.1												
Node 2:	10.0.3.2												
...													
Node 254:	10.0.3.254												
31 / 50													

A hálózat elemeinek paraméterei	
<p>Megjegyzés: ha a hálózati maszk első 24 bitje „1” értékű, akkor a maradék 8 biten található a hálózati node-ok címe, tehát elvileg 256 darab node lehetséges, gyakorlatilag a <b>0. node a hálózatot azonosítja</b>, a 255. node pedig a broadcast cím (lásd később).</p>	
32 / 50	

A hálózat elemeinek paraméterei															
<p>Példa #2:</p> <table> <tr> <td>Prefix-hossz:</td><td>48 bit</td></tr> <tr> <td>Maszk:</td><td>ffff:ffff:ffff:ffff:ffff:ffff:0:0</td></tr> <tr> <td>Hálózat címe:</td><td>fe80::0</td></tr> <tr> <td>Node 1:</td><td>fe80::1 <small>(azaz fe80:0:0:0:0:0:0:1)</small></td></tr> <tr> <td>Node 2:</td><td>fe80::2</td></tr> <tr> <td>...</td><td></td></tr> <tr> <td>Node N:</td><td>fe80::ffff:fffe</td></tr> </table> <p>Megjegyzés: ha a hálózati maszk első 48 bitje „1” értékű, akkor a maradék 16 biten található a hálózati node-ok címe, tehát elvileg 65536 darab node lehetséges, gyakorlatilag a <b>0. node a hálózatot azonosítja</b>, a 65535. node pedig a broadcast cím (lásd később).</p>		Prefix-hossz:	48 bit	Maszk:	ffff:ffff:ffff:ffff:ffff:ffff:0:0	Hálózat címe:	fe80::0	Node 1:	fe80::1 <small>(azaz fe80:0:0:0:0:0:0:1)</small>	Node 2:	fe80::2	...		Node N:	fe80::ffff:fffe
Prefix-hossz:	48 bit														
Maszk:	ffff:ffff:ffff:ffff:ffff:ffff:0:0														
Hálózat címe:	fe80::0														
Node 1:	fe80::1 <small>(azaz fe80:0:0:0:0:0:0:1)</small>														
Node 2:	fe80::2														
...															
Node N:	fe80::ffff:fffe														
33 / 50															

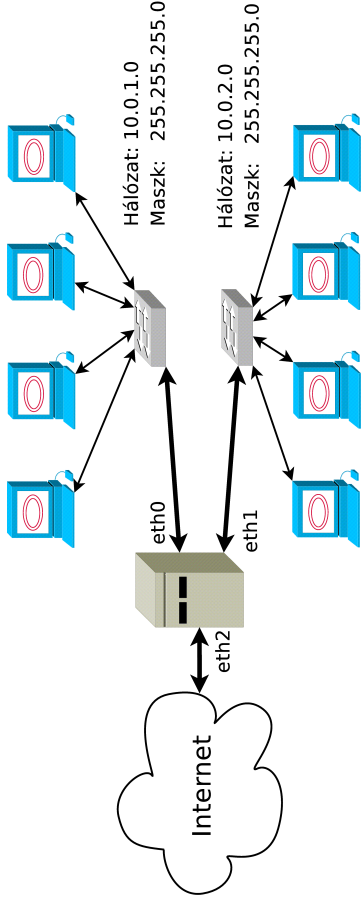
A hálózat elemeinek paraméterei									
<p><b>hálózat címe:</b> ha az IP-cím node része 0, akkor a cím a hálózatot azonosítja (bármely node címe és a maszk közötti bitenkénti „ÉS” kapcsolattal számítható ki).</p> <p>A hálózati címet az átjárók és útvonalválasztók használják.</p> <p>Példák:</p> <table> <tr> <td>IPv4:</td><td>10.0.3.255</td></tr> <tr> <td>IPv6:</td><td>fe80:: <small>(azaz fe80:0:0:0:0:0:0:0)</small></td></tr> </table> <p><b>broadcast cím:</b> ha az IP-cím node részének minden bitje „1” értékű, az IP-cím a hálózat broadcast címe. Ha egy IP-csomag „cél” mezejében a broadcast cím található, akkor azt a csomagot a hálózat minden node-ja fogadja</p> <p>Példák:</p> <table> <tr> <td>IPv4:</td><td>10.0.3.255</td></tr> <tr> <td>IPv6:</td><td>fe80::ffff:ffff <small>(azaz fe80:0:0:0:0:0:0:ffff:ffff)</small></td></tr> </table>		IPv4:	10.0.3.255	IPv6:	fe80:: <small>(azaz fe80:0:0:0:0:0:0:0)</small>	IPv4:	10.0.3.255	IPv6:	fe80::ffff:ffff <small>(azaz fe80:0:0:0:0:0:0:ffff:ffff)</small>
IPv4:	10.0.3.255								
IPv6:	fe80:: <small>(azaz fe80:0:0:0:0:0:0:0)</small>								
IPv4:	10.0.3.255								
IPv6:	fe80::ffff:ffff <small>(azaz fe80:0:0:0:0:0:0:ffff:ffff)</small>								
34 / 50									

Speciális IP-tartományok	
<h2>Speciális IP-tartományok</h2> <p>Bizonyos IP-tartományokat az IETF (Internet Engineering Task Force) speciális célokra tart fenn:</p> <ul style="list-style-type: none"> <li>loopback (localhost) <ul style="list-style-type: none"> <li>127.0.0.0/8</li> <li>::1/128</li> </ul> </li> <li>link local IP-k (IP autokonfigurációra): <ul style="list-style-type: none"> <li>169.254.0.0/16</li> <li>fe80::/10</li> </ul> </li> <li>site local (csak az adott szervezeten belül route-olják) <ul style="list-style-type: none"> <li>fc00::/7</li> </ul> </li> </ul>	
35 / 50	

Speciális IP-tartományok	
<p>multicast IP-címek:</p> <ul style="list-style-type: none"> <li>224.0.0.0/4</li> <li>FF00::/8</li> </ul> <p>privát hálózatok, szabadon felhasználhatók (RFC 1918):</p> <ul style="list-style-type: none"> <li>10.0.0.0/8</li> <li>172.16.0.0/12</li> <li>192.168.0.0/16</li> </ul>	
36 / 50	

## Útvonalválasztás

Az alábbi hálózat arra mutat példát, hogy hogyan kell egy átjáró útvonalválasztását (routing) beállítani.



A **kliens node-ok útvonalválasztása** egyszerű, mert minden, nem nekik szóló IP-csomagot az egyetlen hálózati interfészük felé kell továbbítaniuk.

Az **átjáró** beállításának szabályai:

ha az ábrájába/útvonalválasztóba érkező csomag cél mezeje a 10.0.1.0/255.255.255.0 hálózatot címzi, akkor a csomagot az „eth0” interfészre kell juttatni (az „eth0” interfészen keresztül eljut a „felső” hálózatba, ahol – szerencsés esetben – a cél node található)

Például:

a 10.0.1.23 című node-nak szóló csomagot az „eth0” felé kell routolni, mert

$$10.0.1.23 \text{ \& } 255.255.255.0 = 10.0.1.0 \rightarrow \text{eth0}$$

ha az ábrájába/útvonalválasztóba érkező csomag cél mezeje a 10.0.2.0/255.255.255.0 hálózatot címzi, akkor a csomagot az „eth1” interfészre kell juttatni

**minden mást** az „eth2” interfészen keresztül kell elküldeni (ez a default gateway, azaz az alapértelmezett átjáró az Internet felé)

A route-olási szabályokat a routing tábla tartalmazza (példa):

```
hg8lhs@sanacitta:~$ route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
10.64.64.64 0.0.0.0 255.255.255.255 UH 0 0 0 ppp0
10.0.3.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
0.0.0.0 0.0.0.0 0.0.0.0 U 0 0 0 ppp0
```

A fenti routing tábla jelentése:

a 10.64.64.64 című node-nak (netmask 255.255.255.255, vagyis 1 db IP van az adott hálózatban → 1 db node) küldendő csomagokat a „ppp0” interfészen keresztül kell elküldeni

a **10.0.3.0** című hálózat az „eth0” hálózati kártyára „vannak kötve”

minden más cél-című csomagot a „ppp0” eszközön át kell küldeni

## Hálózati címfordítás – NAT

A NAT a „Network Address Translation” kifejezés rövidítése.

Típusai:

**SNAT:** Source Network Address Translation: az átjáró a belső (privát) hálózatból érkező IP csomag **forrás IP-címét** kicseréli a saját, külső (publikus) IP-címével, a válaszként érkező csomagban pedig „visszacseréli” az IP-címeket.

Tehát a privát hálózatban levő node úgy látja, mintha közvetlenül a cél node-dal kommunikálna, a külső (cél) node pedig azt hiszi, hogy az átjáróval kommunikál. Ezzel a módszerrel elrejthetők a privát hálózat node-jai, növelhető a biztonság.

Gyakran az IP-címek cseréje a TCP ill. UDP portok cseréjével is jár.

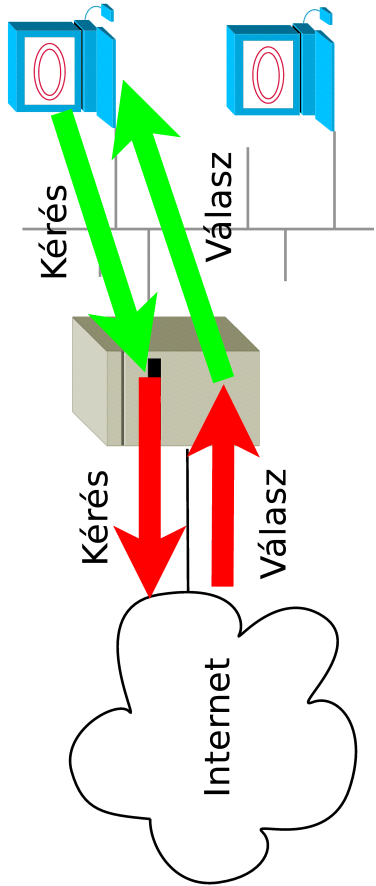
**DNAT:** Destination Network Address Translation: az átjáró a külső hálózatból (pl. Internet) érkező IP-csomagban a **cél IP-címét** (= átjáró publikus IP-címe) kicseréli a privát hálózat egyik gépének IP-címével, majd az privát hálózat gépétől érkező válasz csomagban visszacseréli a címeket.

Ezzel azt érhetjük el, hogy egy szolgáltatást több szerver láthat el: a „kívülről” érkező kéréseket a privát hálózat gépeihez továbbítjuk, azok pedig párhuzamosan tudják elvégezni a szerver-funkciókat. A megfelelő IP-cím kiválasztásának algoritmusá többnyire egyszerű körül-forgó (round-robin, pl. NetBSD-ben) vagy terhelésfüggő (egy túlterhelt szerverre nem küld újabb kérést)

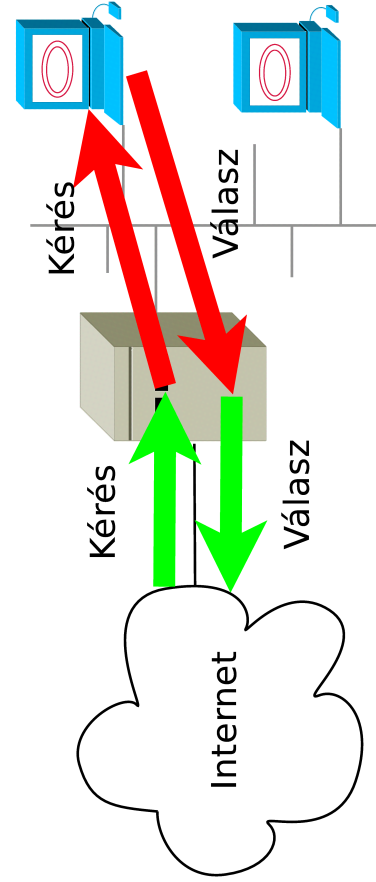
**Port forwarding:** a DNAT speciális esete: nem csak IP-cím alapján szelektál, hanem TCP/UDP port alapján is. Ezzel elérhető, hogy bizonyos szolgáltatásokat (HTTP, FTP, SMTP) más-más szerver szolgáljon ki.



## SNAT - Source NAT



## DNAT - Destination NAT



## Hálózati kiszolgálók

Definíció: a számítógép-hálózat azon elemei, amelyek meghatározott szolgáltatást nyújtanak a többiek számára

A szolgáltatások a szerver gép rögzített TCP és/vagy UDP portjain érhetők el. Ezeket a szolgáltatás-portcím párokat részben szabványosították (), részben „kialakultak”, de facto szabványok lettek

A szolgáltatás-portcím hozzárendelések a /etc/services fájlban találhatóak. Például:

echo - 7/tcp - 7/udp: válaszként visszaküldi azt, amit a kliens üzenetként küldött

daytime - 13/tcp - 13/udp: pontos időt szolgáltat

ftp - 21/tcp: fájlok feltöltése, letöltése (nem biztonságos)

smtp - 25/tcp: levelek küldése  
domain - 53/tcp - 53/udp: gépnév→IP-cím konverzió  
www - 80/tcp: World Wired Web, világháló  
pop3 - 110/tcp: Post Office Protocol, levelek letöltése  
ntp - 123/tcp - 123/udp: Network Time Protocol, pontos idő lekérdezése  
netbios-ns - 137/tcp - 137/udp: Windows fájl megosztás :: név feloldás  
netbios-ssn - 139/tcp - 139/udp: Windows fájl megosztás, session layer (viszony réteg)

Az én /etc/services fájlom 557 összerendelést tartalmaz.

## Az NGW100

Az NGW100 név a „Network GateWay” kifejezésből ered. Képes hálózati átjáróként működni, mert a gyártó cég (ATMEL) két Ethernet csatlakozóval szerelte fel.

Főbb tulajdonságai:

- AP7000 (AVR32B rev. 1) processzor, 130 MHz-es órajellel
- 32 Mbájt SDRAM
- 8 Mbájt NAND FLASH (BOOT memória), JFFS2 fájlrendszerrel
- 8 Mbájt soros FLASH, JFFS2 fájlrendszerrel
- 2 db Ethernet MAC
- USB csatlakozó
- MMC/SD kártya foglalat

2 db SPI csatlakozó  
számos GPIO kivezetés  
soros konzol, JTAG csatlakozó debugolás céljából  
u-BOOT, előtelepített Linux (kernel, függvénykönyvtárak és felhasználói programok)

Szerver funkciói:

- WEB szerver
- Fájl szerver (Windows megosztás)
- ssh és telnet (hálózati bejelentkezés)

<EOF>



# Köszönöm a figyelmet!

Ez a dokumentum a <http://hg8lhs.ham.hu/tdk> oldalról tölthető le.

## Felhasznált szoftverek

Ubuntu Linux 7.10 (kernel: 2.6.22-14\_amd64)  
OpenOffice Writer 2.3.0  
The GIMP 2.4.2  
Dia 0.96.1  
Acrobat Reader for Linux 8.1.1