

Dr. Mileff Péter

# UNIX/LINUX OPERÁCIÓS RENDSZER ÜZEMELTETÉSE

2. ELŐADÁS

Miskolci Egyetem  
Általános Informatikai Tanszék

## Felhasználók kezelése...

### Több felhasználós működés

- A Unix/Linux egy több felhasználós operációs rendszer
  - egyidejűleg több felhasználó is használhatja ugyanazt a rendszert, és mindegyikük akár több programot is futtathat
- Hogyan?
  - A Unix filozófia minden egyes bejelentkezett felhasználóhoz hozzárendel egy-egy úgynevezett terminált:
    - egy terminál pedig egy billentyűzet + megjelenítő egység együttesét jelenti
  - Az adott Unixos géphez legközvetlenebbül csatolt terminált (Linux esetén a gép saját billentyűzetét és monitorát) **konzol terminálnak** nevezzük.
    - Ez abból a szempontból kitérített, hogy bizonyos rendszeradminisztrációs feladatok csak innét hajthatók végre.

### Több felhasználós működés

- A hálózaton vagy grafikus felületen keresztül bejelentkezett felhasználókhöz
  - ún. **pszeudo-terminálok**at rendel a rendszer
    - Ahol a billentyűzet és a képernyő annak a gépnek a billentyűzetéhez és képernyőjéhez rendelődik, amely előtt a felhasználó ül.
- A terminálok megnevezése:
  - a szakzsargonban **tty**, illetve a pszeudo-termináloké **pty** vagy **ttyp**, a Linux ez utóbbi megnevezést használja.
  - Minden Unix rendszer többféle terminált képes kezelni.
  - Az egyes terminálokat gyártó és típusra utaló megnevezéssel azonosítják:
    - Amennyiben csak konzolról használjuk gépünket, elég annyit tudni, hogy a konzol terminál azonosítója „console”

## Felhasználók kezelése

- ◉ Minden felhasználó: egy **azonosító**, és egy **jelszó**.
- ◉ Cél: egy finomabb hozzáférés hierarchia kialakítása:
  - a felhasználókat **csoportok**ba oszthatjuk
  - minden felhasználónak van egy elsődleges csoportja
    - pl. student2009
  - ezen kívül tarthat még más csoportokhoz is.
  - A csoportneveket konvenció szerint kisbetűvel írják.
- ◉ Megvalósítás a háttérben:
  - A rendszer minden egyes felhasználóhoz egy numerikus felhasználó és (esetleg több) csoport-azonosítót rendel:
    - **UID** – felhasználói azonosító (user id)
    - **GID** – csoportazonosító (group id)

5

## Felhasználók kezelése

- ◉ Léteznek kitétetett felhasználónevek is.
  - egy, amelyik minden rendszeren megvan: ez a „**root**” felhasználó.
  - a rendszergazda azonosítója, aki felelős az adott rendszer karbantartásáért és üzemeltetéséért
  - aki a rendszerben „mindent megtehet”.
- ◉ Több, a root-tal ekvivalens felhasználót is létrehozhatunk.
  - Az összes olyan felhasználó, akinek UID-je 0 (felhasználó létrehozásakor ezt megadhatjuk) root-tal ekvivalens lesz.
  - Általában nem érdemes ezt tenni:
    - egy rendszeren éppen elég egy darab teljhatalmú felhasználó.

6

## A fájlrendszer és alapfogalmai...

7

## Alapfogalmak

- ◉ **A Unix/Linux-ban minden fájl.**
- ◉ „Definíció”: olyan „kommunikációs végpont”, ahova vagy byte-folyamot tudunk írni, vagy byte-folyamot tudunk onnét olvasni (esetleg mindkettőt).
- ◉ Fájlként kezelhető tehát:
  - a billentyűzet (csak olvasható), a szöveges képernyő (csak írható), a nyomtató, a CD írónk, stb, de még a fizikai memória tartalma is.
  - A hdd szektorait is egy speciális file olvasásával illetve írásával érhetjük el
  - ugyanígy az hálózati eszközöket is a file-kezelés szabályainak megfelelően használhatjuk.
- ◉ A megfelelő jogosultságokra persze szükség van!

8

## Fájltípusok

- ◉ Legalább háromféle fájl típus:
  - egyszerű fájl, a jegyzékfájlok és a különleges fájl.
- ◉ **Egyszerű fájl:** bármilyen állomány, ami valamilyen adatot, szöveget tartalmaz.
  - pl.: egy forráskód, egy futtatható állomány, egy szöveges dokumentum, stb.
- ◉ **Jegyzékfájlok:** lehetővé teszik, hogy a fájljainkat (mindhárom típust) valamilyen logikai rendszerbe szervezzük.
  - pl.: /home
- ◉ **Különleges fájl:** ezek rendszerint olyan fájl, amelyek valamilyen eszközt képviselnek
  - pl.: szalagos meghajtót, a terminálunkat vagy a hangkártyánkat

9

## A fájlrendszer...

10

## A fájlrendszer

- ◉ A Unix operációs rendszer egyetlen fájlrendszerrel dolgozik.
  - alapvetően egy fa szerkezetű struktúra
  - **egy gyökere** van, a / nevű jegyzék.
- ◉ A gyökérben a korábban tárgyalt háromféle típusú bejegyzések találhatóak.
- ◉ Minden egyes jegyzék további bejegyzéseket és így további jegyzékeket tartalmazhat
  - továbbá biztosan tartalmazza a saját magára mutató .
  - és a szülőjére mutató .. bejegyzést.
  - A gyökérben a .. is saját magára mutat.

11

## A fájlrendszer

- ◉ Egy jegyzékben minden bejegyzésnek külön névvel kell rendelkeznie
  - akkor is, ha egyik sima fájl, mási újabb jegyzékbejegyzés.
- ◉ Nevek konvenciója:
  - bejegyzés neve az ASCII 0-ás és a / jel kivételével tetszőleges karaktert tartalmazhat.
  - Nem lehet 0 hosszúságú, és hosszának maximumát különböző Unix rendszerek másképp rögzítik.
  - Minden normális Unix-ban legalább 14 karakteres lehet a bejegyzés neve,
    - de többnyire 255 karakter megengedett.
    - Efelőtt a karaktereket szó nélkül lecsippenti, figyelmen kívül hagyja a rendszer.

12

## Alapfogalmak

- **Metaadatok:** a fájlrendszer belső adatstruktúrája, amely biztosítja az adatok megfelelő szervezését és elérését a lemezen.
  - Ezek alapvetően „adatokról szóló adatok”.
- Szinte minden fájlrendszernek saját metaadat-szerkezete van:
  - szerepet játszik abban, hogy a fájlrendszerek eltérő teljesítményjellemzőkkel rendelkeznek.
- Nagyon fontos:
  - a metaadatok nem sérülhetnek, mert ekkor a fájlrendszerben tárolt adatok elérhetetlenné válhatnak.

13

## Alapfogalmak

- **Inode:** az inode-ok az egyes fájlokkal kapcsolatos információt tartalmazzák:
  - a méretet,
  - a láncok számát,
  - a mutatókat azon lemezblokkokra, amelyek a fájl tartalmát ténylegesen tárolják,
  - valamint a létrehozás, módosítás és hozzáférés dátumát és idejét.

14

## Alapfogalmak

- **Napló:** naplónak nevezzük azt a belső struktúrát a lemezen, amelyben a fájlrendszer tárolja a fájlrendszer metaadatainak módosításait.
- Előnyei:
  - A naplózás (**journaling**) lényegesen csökkenti a rendszer összeomlását követő helyreállítás idejét.
- Miért?
  - Mert feleslegessé teszi a korábbi hosszú keresési folyamatot
    - amely a rendszerindításnál végigvizsgálta a teljes fájlrendszert.
  - Ehelyett csak a naplóban rögzített események kerülnek újra végrehajtásra.

15

## Standard jegyzékszerkezet

- A Unix/Linux-ok jegyzékszerkezete sok hasonlóságot mutat.
  - A szabadság megvan a jegyzékek rendszerezésében.
- A fájlrendszerek gerincváza:
  - **/bin:** Itt található a létfontosságú bináris programok.
  - **/dev:** A speciális eszközök leőhelye.
  - **/etc:** Mindenféle vegyes dolgot tartalmaz, például egyes programok konfigurációs fájljait, a jelszóállomány(oka)t, stb.
  - **/etc/rc.d:** A rendszer indulását és leállítását irányító szkriptek vannak itt.
  - **/etc/skel:** Az itt lévő fájlokat kapja meg minden új ember a home jegyzékébe.
  - **/home:** Rendszerint itt vannak a felhasználók saját jegyzékai.
  - **/lib:** A legtöbb program futásához nélkülözhetetlen dinamikusan linkelhető könyvtárak vannak itt.
  - **/proc:** Linuxokra jellemző, a processzekkel kapcsolatos információkat hordozó virtuális fájlrendszer.
  - **/root:** A rendszergazda home jegyzéke.

16

## Standard jegyzékszerkezet

- **/sbin**: A rendszergazda számára alapvető fontosságú bináris programok jegyzéke.
- **/tmp**: Ideiglenes fájlok tárolására szolgáló jegyzék. Mindenki írhat bele.
- **/usr**: Ez a jegyzék teszi ki a használt lemezterület nagyságrendileg 80-90%-át.
- **/usr/bin**: Bináris programok, melyek nélkül végsősükség esetén is létezni lehet.
- **/usr/doc**: Dokumentációk.
- **/usr/games**: Mi is lehet ez?
- **/usr/info**: Információs oldalak.
- **/usr/lib**: Mint a **/lib**.
- **/usr/local**: Az adott szerverre speciálisan jellemző dolgokat tartalmazza. Neki is van bin, lib, man, sbin, src és még sok-sok aljegyzéke.
- **/usr/man**: Kézikönyv oldalak.

17

## Standard jegyzékszerkezet

- **/usr/sbin**: Mint **/sbin**, csak kevésbé fontosak.
- **/usr/src**: Forráskódok.
- **/var**: Sűrűn változó dolgok otthona.
- **/var/catman**: Megformázott kézikönyv oldalak.
- **/var/log**: Bizonyos, főleg hálózattal kapcsolatos programok logfájljai (naplói).
- **/var/spool**: Várakozási sorok, például elküldendő levelek vagy elvégzendő nyomtatások feljegyzéseit tartalmazza.
  - Az érkező levelek is sok Unixban itt vannak.

18

## A /dev jegyzék

- **A rendszer-erőforrásokat reprezentáló speciális fájlok.**
  - A **/dev** (devices) aljegyzékben található.
- Ezek teremtik meg a kapcsolatot:
  - a kernel „device driver”-nek (eszközmeghajtó) nevezett, az egyes fizikai eszközök kezeléséért felelős komponensei és a rendszer egyéb részei között.
- Kétféle eszközmeghajtót különböztetünk meg:
  - a **karakteres** („c”) és a **blokkos** („b”) típusút, annak megfelelően, hogy az általa reprezentált eszköz milyen szervezésű.
  - A hdd, vagy floppy eszközmeghajtója blokkos,
  - soros vonal, vagy terminál karakteres pedig típusú.

19

## Főbb eszközmeghajtók

- **/dev/audio**: Ha valamilyen hangkártya vagy más zajkeltő szerkezet van a kernelbe konfigurálva, akkor a **.au** formátumú file-okat ide kiírva meghallgathatjuk őket.
  - Példa: `cat x.au >/dev/audio` Ha a hangkártya digitalizálásra is képes, ez a file olvasható is.
- **/dev/cdrom**: Ez általában egy link a bonyolultabb nevű, valódi CD-ROM speciális file-ra. Hasonlóan használható, mint egy winchester speciális file.
- **/dev/cua\***: A soros vonala(ka)t jelentő speciális file-ok. Írásuk vagy olvasásuk küldést/vételt jelent a megfelelő vonalon.
- **/dev/fd\***: A floppy diszketeket reprezentálják. A **fd0** kezdetű file-ok az A floppyra vonatkoznak, a **fd1** kezdetűek a B-re.
- **/dev/midi, mixer**: Hangkártyához tartozó file-ok, a **/dev/midi** értelemszerűen midi file-ok kezelésére.
- **/dev/mouse, modem**: Általában ezek linkek valamely soros portra.
- **/dev/pty\***: pseudo-terminál vonalakat reprezentáló speciális file-ok.

20

## Főbb eszközmeghajtók

- ◉ **/dev/hd\***: A rendszerben lévő AT buszos hdd-k:
  - a /dev/hda az első winchestert jelenti, /dev/hdb a másodikat.
  - Ha a gépben két IDE vezérlő vagy egy EIDE vezérlő van, akkor a többi diszkhez a /dev/hd1[ab] néven férhetünk hozzá.
  - Ha ezeket a file-neveket számokkal folytatjuk, az egyes diszkeken lévő partíciókhoz jutunk, például /dev/hda1, /dev/hda2.
- ◉ **/dev/sd\***: SCSI diszkek.
- ◉ **/dev/tty\***: A (virtuális) konzol terminálvonalai.
- ◉ **/dev/ttyS\***: Soros vonali terminálok
- ◉ **/dev/null**: Ez egy igen érdekes file: minden beleírt adatot elnyel, és olvasáskor mindig fájlvége-jelét ad.
  - Akkor hasznos, ha egy parancs kimenetét el akarjuk nyomni. Például, ha nem akarjuk a hibaüzeneteket látni:
  - Pl.: „ls -al 2>/dev/null”.

21

## Főbb eszközmeghajtók

- ◉ **/dev/zero**: Az előzőhöz hasonló file, azzal a különbséggel, hogy olvasáskor végtelen sok 0 értékű byte-ot ad vissza.
- ◉ Hol használják?
- ◉ Pl.: fájllok létrehozása:
  - # dd 10kfile bs=1k count=10"
    - létrehoz egy 10Kbyte hosszú, 10kfile nevű csupa 0-ból álló file-t.

22

## Főbb fájlrendszerek...

23

## Főbb fájlrendszerek

- ◉ Korábbi Unix/Linux rendszerek esetében a fájlrendszerek támogatottsága nem volt túl nagy.
  - A 2.4 és újabb Linux kernelék esetén azonban már egész sokféle fájlrendszer közül lehet választani.
- ◉ **Fontos**: nincs olyan fájlrendszer, amely tökéletesen megfelelné mindenféle alkalmazáshoz.
- ◉ Minden fájlrendszernek vannak erősségei és gyengéi, amelyeket figyelembe kell venni.
- ◉ Alapvetően kétféle fájlrendszerről beszélhetünk:
  - **naplózó (journaling)**
  - **nem naplózó** fájlrendszer

24

## ReiserFS...

25

## ReiserFS jellemzői

- Hans Reiser és a Namesys fejlesztőcsapat tervezte.
- Legfontosabb előnyei:
  - **1. Jobb lemezterület-kihhasználás:**
    - az összes adat egy kiegyensúlyozott B\*-fastruktúrába van szervezve.
    - A fastruktúra jobban ki tudja használni a lemezterületet:
      - mivel a kis fájlok közvetlenül a B\*-fa levélcsoportjaiban kerülnek tárolásra,
      - nem pedig egy másik helyen
      - és csak egy mutató mutat a tényleges tárolási helyre.
    - a tárterület nem 1 vagy 4 kilobájtos egységekben kerül lefoglalásra,
      - hanem az adatok pontosan a szükséges méretet foglalják el

26

## ReiserFS jellemzői

- További előnye: **az inode-ok dinamikus lefoglalása.**
- Így a rendszer rendkívül rugalmas, szemben az Ext2-vel:
  - ahol az inode-ok sűrűségét a fájlrendszer létrehozásakor kell megadnunk.
- **2. Jobb lemezhozzáférési teljesítmény:**
  - kis fájlok esetén az adatok és a „stat\_data” (inode) információ általában egymás mellett kerül tárolásra.
  - Ez az információ egyetlen lemez I/O-művelettel kiolvasható,
    - tehát csak egy lemezhozzáférés szükséges a kívánt információ lekéréséhez

27

## ReiserFS jellemzői

- **3. Gyorsabb helyreállítás összeomlás után:**
- a fájlrendszer ellenőrzése nagyon nagy fájlrendszerek esetén is csak néhány másodpercet vesz igénybe.
- a legutolsó metaadat-módosításokat nyomonkövető napló segítségével támogatják.

28

## EXT2...

29

## Ext2 jellemzői

- ◉ Az **Ext2** eredete a Linux történetének első napjaira nyúlik vissza.
  - Az eredeti **Extended File System** 1992 áprilisában készült el és lett beépítve a Linux 0.96c-be.
- ◉ Azóta számos módosítás.
  - Ext2 néven évekig a legnépszerűbb Linux-fájrendszer volt.
- ◉ Mára sokat veszített fontosságából. Oka:
  - A rendkívül rövid helyreállítási idejű naplózó fájlrendszerek megszületése.
- ◉ Legfőbb erénye a **megbízhatóság**.

30

## Ext2 megbízhatósága

- ◉ Számos javításon és komoly tesztelésen ment keresztül.
  - Ezért nevezik sziklaszilárdnak.
- ◉ Rendszerkimaradás után, amikor a fájlrendszer nem szabályosan lett lecsatolva:
  - az e2fsck elkezdli elemezni a fájlrendszer adatait. A metaadatok konzisztens állapotba kerülnek és a függőben lévő fájlok vagy adatblokkok egy kijelölt jegyzékbe íródnak (lost+found).
- ◉ A naplózó fájlrendszerrel ellentétben az e2fsck a teljes fájlrendszert végigvizsgálja
  - Nemcsak az utoljára módosított metaadatbitekét.

31

## Ext2 megbízhatósága

- Hátránya:** lényegesen tovább tart, mint a naplózó fájlrendszer naplóadatainak ellenőrzése.
- ◉ Magas rendelkezésre állást igénylő kiszolgálóhoz nem ajánlatos Ext2 fájlrendszert választani.
- Előnye:**
- ◉ Néha gyorsabb a többi fájlrendszernél:
    - Mert az Ext2 nem tart karban naplót és lényegesen kevesebb memóriát használ.

32



## XFS...

33

## Az XFS fájlrendszer

- ◉ **Fejlesztő:** SGI, az 1990-es évek eleje. Az IRIX operációs rendszerhez.
- ◉ **Az XFS mögötti elképzelés:**
  - ◉ egy olyan nagy teljesítményű 64 bites naplózó fájlrendszer létrehozása volt, amely a mai extrém feldolgozási igényeknek is megfelel.
  - ◉ Az XFS kiválóan kezeli a nagy fájlokat és jól működik csúcsmínőségű hardveren is.
- ◉ **Hátránya:**
  - ◉ a ReiserFS-hez hasonlóan az XFS is nagy gondot fordít a metaadatok integritására,
  - ◉ de az adatok integritására már kevesebbet.

34

## XFS jellemzői

- ◉ **Kiváló méretezhetőség allokációs csoportok használatával:**
  - a fájlrendszer alapjául szolgáló blokkeszköz nyolc vagy több egyenlő méretű lineáris részre van osztva.
    - ◉ Ezeket allokációs csoportoknak hívjuk.
  - Minden allokációs csoport maga kezeli a saját inode-jait és szabad lemezterületét.
  - Mivel az allokációs csoportok egymástól függetlenek, a kernel egyszerre többet is megcímezhet.
  - Ez a funkció a lelke az XFS jó méretezhetőségének

35

## XFS jellemzői

- ◉ **Nagy teljesítmény a lemezterület hatékony kezelésével:**
  - a szabad területet és inode-okat az allokációs csoportokon belül B+ -fák kezelik.
  - A B+ -fák használata nagyban hozzájárul az XFS jó teljesítményéhez és méretezhetőségéhez.
  - Az XFS késleltetett lefoglalást használ.

36

## Támogatott fájlrendszerek

Fájlrendszer	Jellemző
<b>cramfs</b>	Tömörített ROM fájlrendszer: Tömörített, csak olvasható fájlrendszer ROM-okhoz.
<b>hpfs</b>	Nagy teljesítményű fájlrendszer: Az IBM OS/2 szabványos fájlrendszere – csak írásvédett módban támogatott.
<b>iso9660</b>	A CD-ROM-okon használt szabványos fájlrendszer.
<b>minix</b>	Ez a fájlrendszer elvileg egyetemi operációsrendszer-projektekből származik, és ez volt a Linuxon használt első fájlrendszer. Manapság hajlékonylemezek fájlrendszereként használják.
<b>msdos</b>	A fat fájlrendszert eredetileg DOS alatt használták. Ma már más operációs rendszerek is alkalmazzák.
<b>ncpfs</b>	A Novell-kötetek hálózaton keresztüli felkapcsolására szolgáló fájlrendszer.
<b>nfs</b>	Hálózati fájlrendszer (Network File System): Az adatok a hálózat bármely gépén tárolhatók és a hálózaton keresztül elérhetővé tehetők.
<b>smbfs</b>	A Server Message Block protokollt elsősorban a Windows különböző verziói használják a hálózaton keresztüli fájlérés megvalósításához.
<b>sysv</b>	SCO UNIX, Xenix és Coherent (kereskedelmi UNIX- rendszerek PC-khez) alatt használt fájlrendszer.
<b>ufs</b>	BSD, SunOS és NeXTstep alatt használt fájlrendszer. Csak írásvédett módban támogatott.
<b>umsdos</b>	UNIX MSDOS-os: (UNIX on MSDOS): A szokványos fat fájlrendszert bővíti ki: speciális fájlok létrehozásával eléri a UNIX funkcionalitását (jogosultságok, láncok, hosszú fájlnevek).
<b>vfat</b>	Virtuális FAT (Virtual FAT): A fat fájlrendszer kiterjesztése (támogatja a hosszú fájlneveket).
<b>ntfs</b>	Windows NT fájlrendszer, csak olvasható

37

Köszönöm a figyelmet!

38