

Operációs rendszerek

1. gyakorlat: UNIX rendszergazdai ismeretek 1

A UNIX rendszergazdai ismereteket tanszékünkön a hallgatók virtuális gépeken sajátíthatják el. Ezen virtuális gépeknek a használata a 104-es labor LINUX operációs rendszerrel működő PC-in van lehetőség.

Ha nem áll rendelkezésre a terem (pl. Gyakorlatok alkalmával), akkor használható távoli bejelentkezéssel is. Ilyenkor ügyeljünk arra, hogy egy gépen lehetőség szerint csak egy virtuális gép induljon el (több virtuális gép indítása az erőforrások kimerüléséhez és a gép újraindulásához vezethet).

User mode Linux (UmL)

A használt virtuális gép a User mode Linux (röv.: UmL) néven ismert szoftver. Ennek használatához előbb le kell azt másolni. Terjedelme miatt a felhasználói kvóta ezt nem teszi lehetővé, így a felhasználók közös, ideiglenes helyre másolhatják a szoftvert. Az alábbi példa mutatja hogyan:

```
$ mkdir /tmp/$USER
$ cp ~szkovacs/um/* /tmp/$USER
$ cd /tmp/$USER
```

Mivel közös diszk területre másoltuk ezért fontos, hogy munkánk végeztével onnan töröljük:

```
$ cd
$ rm -rf /tmp/$USER
```

Az ideiglenes munkajegyzékünk ezután két fájlt tartalmaz: `linux` futtatható fájl és `root_fs_debian2.2_small`

Az UmL ezután a következő utasítással indítható:

```
$ ./linux root=/dev/ubd0 ubd0=root_fs_debian2.2_small mem=32M
```

Ha elírtunk valamit, akkor az UmL nem indul el és `Kernel panic` hiba üzenettel megáll. Ilyenkor azt tehetjük, hogy egy másik terminál ablakból (ugyanazon a gépen) kiadjuk a következő utasítást:

```
$ killall -TERM linux
```

Ennek hatására a hibásan indult UmL leáll. Esetleg megadható `-KILL` szignál is.

Ha X-Window felületről indítjuk az UmL-t, akkor a indítás során további két ablak nyílik, ami a virtuális gépünk termináljait tartalmazza. Ha Windows-os gépről jelentkezünk be, akkor ezeket az ablakokat nem sikerül megnyitni és ezért kapunk néhány hibaüzenetet. Ennek ellenére az UmL így is használható.

Ha sikeresen elindult az UmL, akkor be lehet jelentkezni, mint egy rendes linux-os gépre. Az egyetlen felhasználó, amivel be lehet jelentkezni a rendszergazda, aminek UNIX rendszereken a neve *root* és az UmL-ben a jelszava *root*.

Az első amivel meg kell ismerkednünk, az operációs rendszer indítása és leállítása. Ehhez tisztában kell lennünk azzal a mechanizmussal, ami a UNIX operációs rendszer működését szabályozza. Ezt SVR-init mechanizmusnak hívjuk.

Az SVR-init mechanizmus

Az AT&T UNIX System V Release óta öröklük a UNIX rendszerek ezt a mechanizmust. A lényege, hogy a kernel betöltődése után egyetlen processzt indít el, ez az *init* (PID=1 PPID=0) processz. Az összes többi processzt a *init* processz indítja. Hogy milyen processzeket kell indítania, az a rendszer futási szintjétől, runlevel-jétől függ. A runlevel egy séma, amely meghatározza, hogy az operációs rendszer adott üzemmódjában milyen processzeknek kell futnia. Néhány szabványos runlevel:

0 – halt: ez egy speciális üzemmód, amelyben az operációs rendszer szabályos leállítása történik.

1 – Single user mode: ebben az üzemmódban a UNIX rendszert csak egyetlen felhasználó, a rendszergazda használhatja. Csak alapvető szolgáltatások működnek, nincs hálózat sem.

2 – Multi user mode: ez az üzemmód az, amelyben minden olyan szolgáltatás rendelkezésünkre áll, amit egy operációs rendszertől elvárunk.

3,4,5 – ezek az üzemmódok rendelkezésre állnak, de nincsen szabványos szerepül. A rendszergazda úgy definiálja ezeket ahogy akarja.

6 – újraindítás: szabályosan leállítja a rendszert, majd újra betölti a kernelt és indítja az operációs rendszert.

Az OS előző és pillanatnyi üzemmódját le is kérdezhetjük. Erre szolgál a *runlevel* utasítás.

Pl.

```
% runlevel
N      2
```

A látható kiírás azt jelenti, hogy a pillanatnyi üzemmód a 2-es, az előző pedig „N”, ami annyit jelent, hogy nem volt előző üzemmód, vagyis a rendszer indítása óta ebben az üzemmódban van.

Egy UNIX rendszer üzemmódját csak a rendszergazda változtathatja meg. Erre szolgál az *init* utasítás. Pl.

```
% init 0
```

Ennek hatására az operációs rendszer szabályosan leáll, majd (ha erre képes a hardver) kikapcsolja a gépet.

A UNIX directory struktúrája

Tekintsük egy kicsit át a UNIX rendszerek szabványos és szokásos jegyzék struktúráját. A gyökér jegyzék / (root) amiben fájlok csak ritkán találhatók, Linux rendszereken rendszerint egy szimbolikus link van itt, ami a kernelre mutat. A gyökérből nyílnak a következő jegyzékek:

bin – ebben a jegyzékben találjuk azokat a bináris fájlokat, programokat, melyek az operációs rendszer külső utasításait valósítják meg.

dev – mivel „UNIX alatt minden fájl”, így kell, hogy legyenek olyan speciális fájlok, melyek pl. hardver eszközöket szimbolizálnak. Ezeket a speciális fájlokat ebben jegyzékben találhatjuk. Az eszközökhöz kötött spec. fájlok típusa kétféle lehet: karakteres (c), ill. blokk (b). A karakteres eszközök jellemzője, hogy nem értelmeznek pozicionáló műveleteket (fseek(), fpos()). Ezzel szemben a blokk eszközök ezeket műveleteket is támogatják.

etc – ebben a jegyzékben találhatók az operációs rendszer konfigurációs fájlljai.

lib – a futtatható programok számára vannak itt dinamikus és statikus library-k.

proc – ez egy speciális jegyzék, aminek a tartalma mindig az operációs rendszer pillanatnyi állapotától függ.

sbin – az operációs rendszer gazdája által használt külső utasítások bináris fájlljai vannak itt.

tmp – ideiglenes fájlok számára fenntartott jegyzék

usr – felhasználói (nem az operációs rendszer részét képző) programok, alkalmazások telepíthetők ebbe a jegyzékbe.

var – ebben a jegyzékben vannak egyes programok állapotinformációi, az operációs rendszer naplója, stb.

root - a rendszergazda home directory-ja

home – a felhasználók home directory-jai

mnt – mountolásra fenntartott directory

A UNIX rendszerek jegyzékstruktúrája nem egyetlen fájlrendszert takar, hanem rendszerint többet. Így az egyes jegyzékek tartalma fizikailag különböző diszkeken helyezkedhet el. Egy eszköz fájlrendszerének a jegyzék struktúra egy jegyzékéhez való kötését *mount*-olásnak nevezzük.

Új fájlrendszer létrehozása

Az UmL virtuális gép, így virtuális diszkeket használ. Ezek a virtuális diszkek a gazdagépen fájlként valósulnak meg. Ha a virtuális géphez új diszket akarunk kapcsolni, akkor az UmL leállítása után a következőt kell tennünk. Létre kell hozni egy megfelelő méretű üres fájlt:

```
$ dd if=/dev/zero of=banyek bs=1024 count=10240
```

Ezután újraindítjuk az UmL-t az alábbi parancssorral:

```
$ ./linux \
```

- `root=/dev/ubd0 \`
- `ubd0=root_fs_debian2.2_small \`
- `ubd1=banyek \`
- `mem=32M`

Az újonnan létrehozott eszköz a `/dev/ubd/1` amit az imént létrehozott *banyek* nevű fájl tartalmaz. Hogy használhassunk ezt az eszközt először fájlrendszert kell létrehozni rajta. A fájlrendszer létrehozását a UNIX *mkfs* utasítása teszi lehetővé. Mivel Linux rendszereken a legelterjedtebb fájlrendszer típus a 2nd extended file system (röv. ext2fs), ezért a fejlesztők rendelkezésünkre bocsátottak egy *mkfs.ext2* programot is. Ennek a programnak egyetlen argumentuma az eszköz amin létre szeretnénk hozni a fájlrendszert.

```
% mkfs.ext2 /dev/ubd/1
```

Ha ez sikeresen végbement, akkor elkészült a diszken a fájlrendszer. Az utasítás kiadásának feltétele, hogy a megadott eszköz blokk eszköz legyen. Hogy lássuk a diszk tartalmát, be kell azt kötni a directory struktúrába. Erre a *mount* utasítást fogjuk használni. Az utasítás kiadása argumentum nélkül az éppen mount-olt fájlrendszereket listázza. Ha mount-olni szeretnénk egy fájlrendszert, akkor két argumentumot kell megadni: mit és hová.

```
% mount /dev/ubd/1 /mnt
```

Az utasítás hatására a `/dev/ubd/1` diszk tartalma megjelenik a */mnt* jegyzékben. Az utasítás kiadásának a feltétele, hogy a megadott eszközön létezzen fájlrendszer és a megadott jegyzék üres legyen.

Most használhatjuk a diszk fájlrendszerét, úgy mint egy tetszőleges jegyzéket a directory struktúrában.

Ha ki szeretnénk kötni a fájlrendszer a jegyzékből, akkor a *mount* utasítást kell visszavonni. Erre szolgál az *umount* utasítás, aminek egyetlen argumentuma az eszköz, vagy a jegyzék lehet.

```
% umount /mnt
```

ugyanazt a hatást érzük el, ha

```
% umount /dev/ubd/1
```

utasítást használva. Az utasítás kiadásának a feltétele, hogy senki ne használja az umount-olandó fájlrendszert.