

Kurs administrowania systemem Linux 2023

Lista zadań na pracownię nr 11

Na zajęcia 13 i 22 maja 2024

Zadanie 1 (2 pkt). Zapoznaj się z programami `fdisk(8)` i GNU `parted(8)`. Przygotuj ich omówienie i porównanie. Za pomocą programu `truncate(1)` utwórz plik `disk.img` o rozmiarze 976580000 sektorów (we wszystkich zadaniach tej listy sektor oznacza 512 bajtów). Stwórz na dysku `disk.img` następujący układ partycji:

| Start | End | Rozmiar | Zawartość |
|-----------|-----------|------------|--|
| 0 | 0 | 1 s | MBR |
| 1 | 62 | 31 KiB | obszar nie przydzielony |
| 63 | 67108863 | ~32 GiB | partycja główna nr 1 typu 0x0c (FAT32 LBA) |
| 67108864 | 486539263 | 200 GiB | partycja główna nr 2 typu 0x83 (Linux) |
| 486539264 | 629145599 | 150 GiB | obszar nie przydzielony |
| 629145600 | 976579999 | ~165.7 GiB | partycja główna nr 4 typu 0x05 (rozszerzona) |

Wnętrze partycji rozszerzonej powinno być podzielone następująco:

| | | | |
|-----------|-----------|----------|---|
| 629145600 | 629145600 | 1 s | EBR partycji logicznej nr 5 |
| 629145601 | 629147647 | 2047 s | obszar nie przydzielony (<i>padding</i> EBR) |
| 629147648 | 713033727 | 40 GiB | partycja logiczna nr 5 typu 0x83 (Linux) |
| 713033728 | 975173631 | ~125 GiB | obszar nie przydzielony partycji rozszerzonej |
| 975173632 | 975173632 | 1 s | EBR partycji logicznej nr 6 |
| 975173633 | 975175679 | 2047 s | obszar nie przydzielony (<i>padding</i> EBR) |
| 975175680 | 976199679 | 500 MiB | partycja logiczna nr 6 typu 0x82 (Linux swap) |
| 976199680 | 976579999 | ~186 MiB | obszar nie przydzielony partycji rozszerzonej |

Program `fdisk(8)` powinien wypisać następującą listę partycji:

```
Disk disk4.img: 465.7 GiB, 500008960000 bytes, 976580000 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xbe9689af
```

| Device | Boot | Start | End | Sectors | Size | Id | Type |
|------------|------|-----------|-----------|-----------|--------|----|----------------------|
| disk4.img1 | | 63 | 67108863 | 67108801 | 32G | c | W95 FAT32 (LBA) |
| disk4.img2 | | 67108864 | 486539263 | 419430400 | 200G | 83 | Linux |
| disk4.img4 | | 629145600 | 976579999 | 347434400 | 165.7G | 5 | Extended |
| disk4.img5 | | 629147648 | 713033727 | 83886080 | 40G | 83 | Linux |
| disk4.img6 | | 975175680 | 976199679 | 1024000 | 500M | 82 | Linux swap / Solaris |

Zapoznaj się z poleceniami `losetup(8)` (w tym z jego opcjami `--show`, `--find` i `--partscan`) oraz `partx(8)`, `partprobe(8)` i `kpartx(8)`. Utwórz urządzenie blokowe skojarzone z plikiem `disk.img` i poproś jądro o utworzenie urządzeń odpowiadających partycjom tego urządzenia. Za pomocą polecenia `lsblk(1)` pokaż, że wszystkie partycje są dostępne w systemie jako urządzenia blokowe.

Zadanie 2 (3 pkt). Oto zrzut szesnastkowy zawartości pewnego dysku twardego:¹

```

0000000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | .....|
*
00000001b0 00 00 00 00 00 00 00 00 46 3f f9 21 00 00 00 82 | .....F?!.|
00000001c0 03 00 0c 2d 07 66 00 20 00 00 00 00 00 05 00 01 | ...-.f. ....|
00000001d0 e2 a0 83 b6 8a 2c 00 00 00 60 00 00 a0 00 00 2d | .....,...'-|
00000001e0 08 66 05 57 f3 64 00 20 00 05 00 00 80 0c 00 56 | .f.W.d. ....V|
00000001f0 42 15 83 01 e1 a0 00 00 00 40 00 00 00 20 55 aa |B.....@... U.|
0000000200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | .....|
*
0a004001b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 4d | .....M|
0a004001c0 28 66 83 d7 2f e8 00 08 00 00 00 00 20 00 00 62 | (f../......b|
0a004001d0 78 6b 05 98 a8 70 00 08 40 00 00 08 40 00 00 00 | xk...p..@...@...|
0a004001e0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | .....|
0a004001f0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 aa | .....U.|
0a00400200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | .....|
*
0a805001b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 83 | .....|
0a805001c0 59 6b 83 98 a8 70 00 08 00 00 00 00 40 00 00 00 | Yk...p.....@...|
0a805001d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | .....|
*
0a805001f0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 aa | .....U.|
0a80500200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | .....|
*
fffffffff0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | .....|

```

Znajdź identyfikator tego dysku oraz przygotuj tabele opisujące podział tego dysku na partycje, przypominające tabele z zadania 1. W dwóch pierwszych kolumnach powinny się znaleźć numery (LBA, dziesiętnie) pierwszego i ostatniego sektora opisywanego obszaru, przy czym w każdym wierszu, z wyjątkiem pierwszego, numer początkowy powinien być o 1 większy niż numer końcowy z wiersza poprzedniego (tzn. należy podzielić cały dysk na rozłączne obszary).

Utwórz za pomocą polecenia `truncate(1)` obraz dysku o takim samym rozmiarze i za pomocą polecenia `fdisk(8)` lub `parted(8)` utwórz na nim taki sam układ partycji. Za pomocą poleceń `dd(1)` i `hd(1)` sprawdź, że zawartość Twojego dysku jest identyczna z zawartością oryginalnego dysku (może się ewentualnie różnić kodem wykonywalnym w MBR, w zależności od użytego programu).

Zadanie 3 (Zrób sobie własny bootloader. 5 pkt). Przygotuj *pendrive* zawierający pojedynczą partycję MS-DOS rozciągającą się od 2048 sektora do końca urządzenia i zawierającą system plików FAT32 (tj. zupełnie zwyczajny *pendrive*). W pierwszych 446 bajtach MBR umieść program, który w razie próby uruchomienia przez BIOS systemu z tego *pendrive'a* wykona następujące czynności:

- wypisze na ekranie (przerwanie `int10h`) menu złożone z opcji:
 - License
 - Reboot
- wczyta z klawiatury znak;
- jeśli wybrano 1, to wypisze na ekran zawartość sektora 1 (bezpośrednio następującego po MBR); w sektorze tym należy umieścić test ASCII zawierający skróconą (żeby zmieściła się w 512B) treść licencji MIT i odczytywać go do pamięci za pomocą `int13h`;
- jeśli wybrano 2, to uruchomi ponownie BIOS (skacząc pod adres `0xffff0`);

¹Poniższy zrzut znajduje się też w pliku tekstowym `hexdump.txt` będącym załącznikiem do bieżącego pliku PDF.

5. powróci do punktu 1.

Przyda się w tym zadaniu jakiś dobry assembler, np. **nasm** i przyzwoity *hex editor*. Pomysłodawcą zadania jest Piotr Polesiuk.

Zadanie 4 (Zburzcie ten system, a ja go odbuduję w kilka sekund. 3 pkt). Zrób prostą instalację Linuksa na niewielkim dysku z partycjami MS-DOS, wykorzystującą cztery partycje: `rootfs`, `/var`, `/home` i `swap` (możesz np. użyć instalatora Debiana). Zapoznaj się z programem `sfdisk(8)` (w szczególności jego opcją `--dump`). Zrób za jego pomocą *backup* układu partycji na dysku. Wykonaj *backup* zawartości wszystkich partycji z wyjątkiem partycji `swap` w postaci zbioru skompresowanych archiwów TAR. Przygotuj skrypt powłoki, który na podanym urządzeniu blokowym automatycznie odtworzy zapamiętany układ partycji, założy odpowiednie systemy plików (pamiętaj o ustawieniu właściwych UUID-ów tych systemów, jeśli są wykorzystywane do montowania), zamontuje partycje we właściwych punktach montażowych (`rootfs` np. w katalogu `/mnt`), rozpakuje archiwa TAR, wykona `chroot(8)` do odtwarzanego systemu, a w nim `grub-install` w celu odtworzenia *bootloadera*. Skrypt powinien działać całkowicie automatycznie i w ciągu kilku sekund przywrócić kompletną instalację systemu. Zauważ, że `parted(8)`, a nawet `fdisk(8)` również mogą pracować wsadowo.