

FAT32 steg för steg



Lär dig att se innehållet på diskar,
på samma vis som Mouse ser innehållet i Matrix.

Wecksten, Mattias

2010



Utvinn en diskavbild

```
> sudo fdisk -l
Disk /dev/sdb: 91201 cylindrar, 255 huvuden, 63 sektorer/spår
Enheter = cylindrar med 8225280 byte, block med 1024 byte, räknat från 0

Enhet Start Början Slut Cyl. Block Id System
/dev/sdb1 0+ 12 13- 104391 7 HPFS/NTFS
/dev/sdb2 13 91200 91188 732467610 7 HPFS/NTFS
/dev/sdb3 0 - 0 0 0 Tom
/dev/sdb4 0 - 0 0 0 Tom
```



Först så tittar vi efter vilka enheter vi har i systemet. Jag hittar bland annat enheten /dev/sdb.

Utvinn en diskavbild

```
> sudo dd if=/dev/sdb of=disk_1.dd  
208782+0 poster in  
208782+0 poster ut  
106896384 byte (100 GB) kopierade, 6,944 s, 27,2 MB/s  
> _
```

Här utvinner vi hela disken med alla partitionerna.

Utvinn en diskavbild

```
> sudo dd if=/dev/sdb of=disk_1.dd  
208782+0 poster in  
208782+0 poster ut  
106896384 bytes (100 GB) kopierade, 6,944 s, 27,2 MB/s  
> _
```

Här utvinner vi hela disken med alla partitionerna.

Analys av partitionsblocken

```
> dd if=/dev/loop1 bs=1 skip=446 count=64 | xxd -c 16
000000: 0002 0300 0b0e 321e 8000 0000 00b8 0700 .....?.....
000010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
000030: 0000 0000 0000 0000 0000 0000 0000 0000 .....
> _
```

Wecksten, Mattias

2010



Nu tittar vi bara på de fyra partitionsblocken. Jag har markerat tre fält – partitionstypen (0x0b), offset till partitionens start (0x0000 0080) och partitionens längd (0x0007 b800).

0x0b = FAT32
0x0000 0080 = 128 sektorer
0x0007 b800 = 505 856 sektorer

Kontroll av partitionsinformationen

```
> mmls -t dos disk_1.dd
DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

  Slot   Start      End          Length      Description
00: ----  000000000  000000000  000000001  Primary Table (#0)
01: ----  000000001  000000127  000000127  Unallocated
02: 00:00  000000128  000000993  000000866  Win95 FAT12 (x00B)
03: ----  000000994  000011999  000000016  Unallocated
> _
```

Wecksten, Mattias

2010



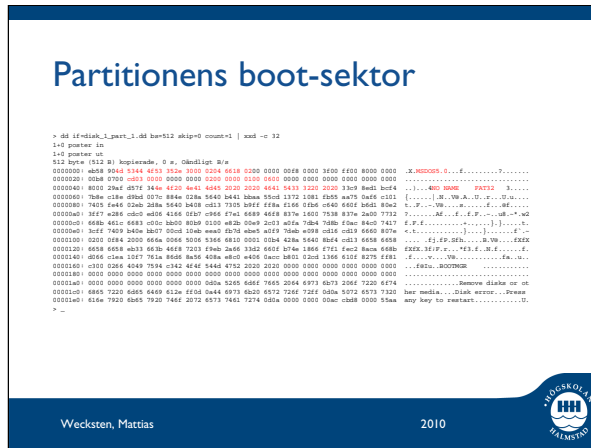
En snabb kontroll med mmls visar att vi räknat rätt.

Multiplicera partitionens längd i sektorer med sektorns storlek i byte (512) och vi får att storleken är ungefär 250 MB, eller helt exakt 258 998 272 bytes.

Utvinn en partition

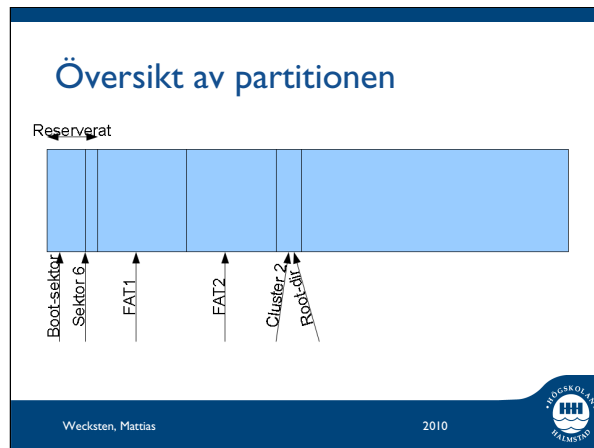
```
> dd if=disk_1.dd of=disk_1_part_1.dd bs=512 skip=128 count=505856
505856+0 poster in
505856+0 poster ut
258998272 byte (259 MB) kopierade, 5,924 s, 43,7 MB/s
> _
```

Här utvinns vi partition 1 från diskavbilden. Vi ber dd att sätta blocklängden till 512, hoppa över 128 block (vår offset) och sedan kopiera 505 856 block. Efter kopieringen rapporterar dd att filen blev exakt 258 998 272 bytes, precis samma som vi räknat fram.



Här ser vi boot-sektorn av partition 1.

- MSDOS5.0 = OEM
- 0x200 = Byte/sektor (512)
- 0x04 = Sektorer/cluster (4)
- 0x1866 = Reserverade ytan (6246)
- 0x02 = Antal FATs (2)
- 0x0000 03cd = FAT i sektorer (973)
- 0x0000 0002 = Root directory-kuster (2)
- 0x0001 = FSINFO-sektor (1)
- 0x0006 = backup boot-block
- sektor (6)
- NO NAME = Volume label
- FAT32 = File system type label
- 0xAA55 = Signatur



En översikt av partitionens olika delar.

Kontrollera boot-sektorns backup

```
> dd if=disk_1_part_1.dd bs=512 skip=6 count=1 | xxd -c 32
1=0 pointer in
112 byte (512 B) kopierade, 0 % Öändligt B/A
000000: 4d58 5046 5146 4c31 3230 3203 0204 4618 0200 0000 0018 0000 3f00 ff00 8000 0000  X.MD5D58.0.f.....
000020: 0028 9700 c603 0000 0000 0000 0000 0100 0400 0000 0000 0000 0000 0000 0000 0000  .....
000040: 8020 204f 0a7f 1646 4c20 4a41 4640 3020 3020 4641 5413 3200 3200 3700 8a01 8a64  [.....]MD5D58  B.....
000060: 7826 c18a d8bd 0070 884e 028a 5640 8441 bbaa 5548 1172 1081 e505 aa75 0a66 c101  [.....]M.A.D.f...D.M.....
000080: 7451 1646 03a0 285a 8440 8488 0a13 7075 001a 1146 1146 0040 0400 8401 83a0  [.....]M.....M.....
000100: 3f47 2d85 ad0c 4d06 4166 0167 0866 27a1 6669 4628 8376 1600 7538 8376 2400 7712  [.....]E...f.f.f...uM--W
000120: 6680 441c 6843 020c 8000 8000 0210 4820 0049 20c3 a61a 7804 7804 f30c 8400 7417  [.....].....].....t.
000140: 3c7f 7816 8406 807f 0001 1048 4e4d 0170 404b 4073 78a0 0088 0c10 0c10 6500 8076  [.....].....f.....
000160: 0200 0384 2000 664a 0066 500c 5346 6810 0001 0364 434a 5640 804c 0c13 6658 6658  [.....]f.f.f.f.....B.W.....EEXX
000180: 6658 6658 4a13 6020 4661 7213 f3a0 8456 33a5 4641 874e 3866 771c f4c0 84a0 4f80  [.....]EEXX.f.f.f...T.S.f.....f.
000200: 0546 01a4 11e7 783a 8568 8436 408a 48c0 4400 84c0 020c 1366 610f 6278 f481  [.....]f.....f.....f.....f.
000220: c010 0264 4149 7836 c44c 4e4f 8446 8742 020c 2121 0000 0000 0000 0000 0000  [.....]EEXX.MD5D58.....
000240: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  [.....].....M.D58.....
000260: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  [.....].....M.D58.....
000280: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  [.....].....M.D58.....
000300: 6665 7220 6665 6469 612e f403 0a44 6973 6b20 6572 724f 724f 0d8a 5072 6573 7120  [.....]D58 error...Press
000320: 4166 7820 6665 7820 744f 2072 6a73 7461 7274 000a 0000 0000 000a c0a8 0000 000a  [.....]M any key to restart.....E
> _
```

Wecksten, Mattias

2010



Vi hoppar fram till sektor 6 för att kontrollera att vi hittar en identisk kopia av boot-sektorn.

Hitta och kontrollera FAT I

```
> dd if=diak_1_parr_1.dd bs=512 skip=6244 count=1 | xxd -c 32
1=0 pointer in
1=0 pointer in
512 byte (512 B) kopierade, 0 % öändligt B/s
000000:  ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff .....
000020:  0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 .....
000040:  1200 0000 ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff .....
000060:  ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff .....
000080:  ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff ffff .....
0000a0:  2000 0000 2a00 0000 2b00 0000 2c00 0000 2d00 0000 2e00 0000 2f00 0000 3000 0000 .....
0000c0:  3100 0000 3200 0000 3300 0000 3400 0000 3500 0000 3600 0000 3700 0000 3800 0000 .....
0000e0:  3900 0000 3a00 0000 3b00 0000 3c00 0000 3d00 0000 3e00 0000 3f00 0000 4000 0000 .....
000100:  4100 0000 4200 0000 4300 0000 4400 0000 4500 0000 4600 0000 4700 0000 4800 0000 .....
000120:  4900 0000 4a00 0000 4b00 0000 4c00 0000 4d00 0000 4e00 0000 4f00 0000 5000 0000 .....
000140:  5100 0000 5200 0000 5300 0000 5400 0000 5500 0000 5600 0000 5700 0000 5800 0000 .....
000160:  5900 0000 5a00 0000 5b00 0000 5c00 0000 5d00 0000 5e00 0000 5f00 0000 6000 0000 .....
000180:  6100 0000 6200 0000 6300 0000 6400 0000 6500 0000 6600 0000 6700 0000 6800 0000 .....
0001a0:  6900 0000 6a00 0000 6b00 0000 6c00 0000 6d00 0000 6e00 0000 6f00 0000 7000 0000 .....
0001c0:  7100 0000 7200 0000 7300 0000 7400 0000 7500 0000 7600 0000 7700 0000 7800 0000 .....
0001e0:  7900 0000 7a00 0000 7b00 0000 7c00 0000 7d00 0000 7e00 0000 7f00 0000 8000 0000 .....
> _
```

Wecksten, Mattias

2010



Vi hoppar förbi den reserverade ytan (6246 sektorer) och hamnar i början av FAT1. 0x0fff fff8 står för "end-of-file", troligtvis filer mindre än ett kluster. 0 står för oallokerat utrymme. Övrigt anger nästa kluster i kedjan. Om disken inte är fragmenterad kommer klustren troligtvis i ordning, därav mönstret (1 2 3 4 ...).

Hitta och kontrollera FAT2

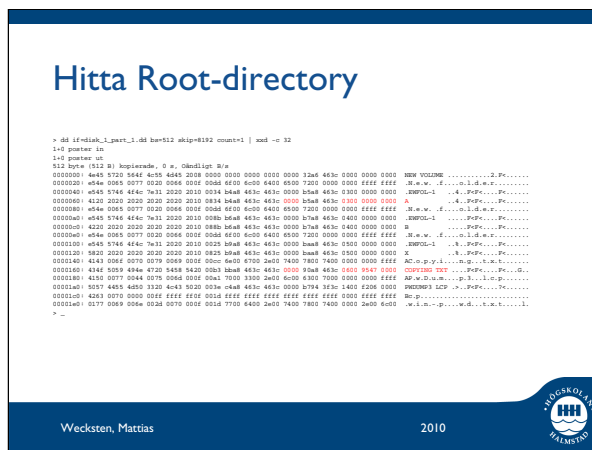
```
> dd if=diak_1_parr_1.dd bs=512 skip=7219 count=1 | xxd -c 32
000000:  ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef .....
000020:  0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 .....
000040:  1100 0000 ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef .....
000060:  ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef .....
000080:  ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef ffef .....
0000a0:  2000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 .....
0000c0:  1100 0000 3200 0000 3300 0000 3400 0000 3500 0000 3600 0000 3700 0000 3800 0000 .....
0000e0:  3900 0000 3a00 0000 3b00 0000 3c00 0000 3d00 0000 3e00 0000 3f00 0000 4000 0000 .....
000100:  4100 0000 4200 0000 4300 0000 4400 0000 4500 0000 4600 0000 4700 0000 4800 0000 .....
000120:  4900 0000 4a00 0000 4b00 0000 4c00 0000 4d00 0000 4e00 0000 4f00 0000 5000 0000 .....
000140:  5100 0000 5200 0000 5300 0000 5400 0000 5500 0000 5600 0000 5700 0000 5800 0000 .....
000160:  5900 0000 5a00 0000 5b00 0000 5c00 0000 5d00 0000 5e00 0000 5f00 0000 6000 0000 .....
000180:  6100 0000 6200 0000 6300 0000 6400 0000 6500 0000 6600 0000 6700 0000 6800 0000 .....
0001a0:  6900 0000 6a00 0000 6b00 0000 6c00 0000 6d00 0000 6e00 0000 6f00 0000 7000 0000 .....
0001c0:  7100 0000 7200 0000 7300 0000 7400 0000 7500 0000 7600 0000 7700 0000 7800 0000 .....
0001e0:  7900 0000 7a00 0000 7b00 0000 7c00 0000 7d00 0000 7e00 0000 7f00 0000 8000 0000 .....
140 pointer to
140 pointer to
112 byte (112 x) kopierade, 0 s., öändligt B/s
> _
```

Wecksten, Mattias

2010



För att hitta FAT2 så hoppar vi förbi reserverade ytan och längden av FAT1 (973).



Root directory kunde hittas i kluster 2. Kluster 2 ligger precis efter FAT2. Hoppa över reserverade ytan + 2 gånger storleken på FAT.

Folder A hittar vi i kluster 0x0000 0003. Varje kluster är 4 sektorer, det vill säga, hoppa 4 sektorer från kluster 2. Foldrar har storlek 0.

Filen COPYING.TXT hittar vi i kluster 0x0000 0006. Hoppa $(6-2)*4=16$ sektorer. Storleken på filen är $0x0000 4795=18325$ bytes. Detta betyder att filen kommer att ockupera 9 kluster. Vilken är den största filstorleken du teoretiskt kan ha i FAT32?

Foldern A

```
> dd if=diak_1_parr_1.dd bs=512 skip=8196 count=1 | xxd -c 32
1=0 pointer in
1=0 pointer in
312 byte (512 B) kopierade, 0 s, ömslagigt B/A
00000000 2a2d 2020 2020 2020 2020 2010 0014 6a48 463c 463c 0000 65a8 463c 0300 0000 0000  .4.PdP.....Pc.....
00000020 2a2d 2020 2020 2020 2020 2010 0014 6a48 463c 463c 0000 65a8 463c 0000 0000 0000  .4.PdP.....Pc.....
00000040 4120 2077 0044 0076 0068 0060 0060 0060 0060 0060 0060 0060 0060 0060 0060 0060  AB w.D.m.....p.2..i.m.p.....
00000060 5057 4455 4450 3220 4c43 5020 0037 c0a8 463c 0000 5a44 3f3c 1300 9706 0000  PWDENMP LCP .7..PdP.....Lc.....
00000080 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
000000a0 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
000000c0 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
000000e0 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
00000100 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
00000120 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
00000140 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
00000160 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
00000180 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
000001a0 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
000001c0 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
000001e0 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
000001f0 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000  .....
> _
```



Foldern A innehåller filer.

Hitta COPYING.TXT i FAT1

```
> dd if=disk_1_part_1.dd bs=512 skip=6246 count=1 | xxd -c 4
1+0 poster in
1+0 poster ut
512 byte (512 B) kopierade, 0 s., 0ändligr B/s
000000: ffff ffff ....
000004: ffff ffff ....
000008: ffff ffff ....
00000c: ffff ffff ....
000010: ffff ffff ....
000014: ffff ffff ....
000018: 0700 0000 ....
00001c: 0000 0000 ....
000020: 0000 0000 ....
000024: 0a00 0000 ....
000028: 0000 0000 ....
00002c: 0000 0000 ....
000030: 0000 0000 ....
000034: 0000 0000 ....
000038: ffff ffff ....
00003c: 1000 0000 ....
> _
```

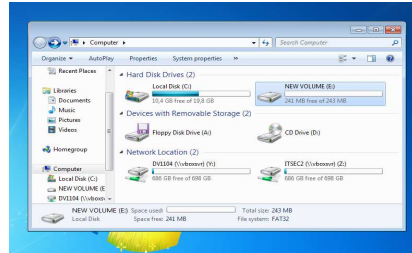
Wecksten, Mattias

2010



Jag dumpar FAT1 formaterat som kluster. FAT börjar med kluster 0 och går uppåt. I kluster 6 finns en pekare till kluster 7. I kluster 7 en pekare till kluster 8... I kluster 14 finner vi slutligen end-of-file markering. Jag hävdade tidigare att denna fil skulle ockupera exakt 9 kluster. Vid en snabb kontroll visar det sig stämma exakt.

Vad vi har tittat på

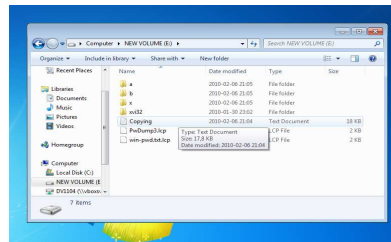


Wecksten, Mattias

2010



Vad vi har tittat på

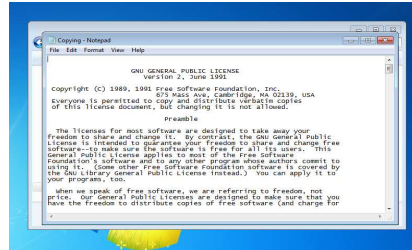


Wecksten, Mattias

2010



Vad vi har tittat på



Wecksten, Mattias

2010



Vad vi har tittat på

