Here are my notes about setup of various stuff on Eee PC to make it work better for me.

t=0x8d0f1f0 [0,0]

Contents: [Dobrica PavlinuÅ¡iÄ 's random unstructured stuff]

- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Startup)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Compressed root filesystem)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Update 2008-01-22)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Links)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Disk images)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Backup image from Eee using external USB disk)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Backup flash image from Eee PC using network)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Backup just part of image)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Emulation)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Example flash image)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Mount file-system)
  - Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Start emulation)
- Dobrica PavlinuÅ¡iÄ 's random unstructured stuff (Links)

Startup

Edit `/usr/bin/startsimple.sh` and insert something along following lines before exec icewm

```
sudo rm /tmp/nologin

xrdb -merge ~/.Xresources
setxkbmap hr us
xterm &

exec icewm
```

Compressed root filesystem

I don't really care much about Xandos on my Eee PC. However, I really do like idea about having read-only system filesystem (especially if your startup scripts are breakable as easy as ones on eee are). So, to improve this idea, I started to think how to compress read-only partition so I can at least save space.

As a first experiment, I copied whole flash from eee (about 3.6Gb used) and compressed it using `gzip -1` (lowest possible compression level). I was quite amazed to see that resulting archive was only 1.3Gb. So, I was up to something (and additional 2Gb of free space on 4Gb eee is also nice :-)

Update 2008-01-22

squashfs 3.3 can't compress comtent of eee's `/usr` without hanging on flock after about ~47000 files. This is quite annoying, but 3.2 works. Since it's compiled from upstream source it doesn't include lzma compression, but is saves 1.6Gb of disk space. More details is available, but in Croatian only.

Links

Here is collection of references about this issue:

fetchrss: http://del.icio.us/rss/dpavlin/debian+usb

- There was an error: 500 SSL negotiation failed:

Disk images

Backup image from Eee using external USB disk

```
dd if=/dev/sda of=/media/A/Partition1/flash4Gb.img
```

Path in `of` may be different depending on partition on your disk.

Backup flash image from Eee PC using network

Transfer somehow whole disk image to other computer. Good way might be to use netcat with something like this:

- on Eee

```
sudo nc -l -p 8888 < /dev/sda
```

- on other computer

```
nc name.of.eee.pc 8888 > hda
```

You might want to insert compression if your network connection is slower than flash read speed (which is according to `hdparm -tT /dev/hda` around 21MB/sec).

Alternative is to take `P701L.gz` from DVD which came with machine, but it has only one partition which is factory default one.

Backup just part of image

You can also copy just parts of flash filesystem if you want (this copies just disk after partition 2):

- on eee

```
dd if=/dev/sda bs=512 skip=4819500 | gzip | nc -w 3 other.computer
88882995524+0 records in
```

```
2995524+0 records out
1533708288 bytes (1.5 GB) copied, 279.348 seconds, 5.5 MB/s
```

- on other computer

```
nc -l -p 8888 | gzip -cd | > /rest/tmp/hda2-4
```

- now, ectract beginning of disk and first partition from P701L

```
dd if=P701L of=hda1 bs=512 count=4819500
```

- and merge partition together to create full disk image

```
cat hda1 hda2-4 > hda
```

Emulation

How to create virtual Eee PC?

Example flash image

```
# fdisk -l hda

Disk /backup/eee/hda: 3 GB, 3997486080 bytes
255 heads, 63 sectors/track, 486 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

          Device Boot      Start         End      Blocks   Id  System
/backup/eee/hda1               1         300     2409718   83  Linux
/backup/eee/hda2             301         484     1469947   83  Linux
/backup/eee/hda3             485         485           0    c  FAT32 LBA
/backup/eee/hda4             486         486           0   ef  EFI FAT
```

Mount file-system

We need first file system (factory defaults) to get access to kernel and initrd image

```
# fdisk -u -l hda

Disk /backup/eee/hda: 3 GB, 3997486080 bytes
255 heads, 63 sectors/track, 486 cylinders, total 7807590 sectors
Units = sectors of 1 * 512 = 512 bytes

          Device Boot      Start         End      Blocks   Id  System
/backup/eee/hda1              63     4803435     2409718   83  Linux
/backup/eee/hda2         4819563     7759395     1469947   83  Linux
/backup/eee/hda3         7775523     7775460           0    c  FAT32 LBA
```

```
/backup/eee/hda4         7791588     7791525           0   ef  EFI FAT
# mkdir 1
# mount hda 1 -o loop,offset=`expr 63 \* 512`
```

Start emulation

```
qemu -m 512 -hda hda -kernel boot/vmlinuz-2.6.21.4-eeepc -initrd boot/initramfs-eeepc.img -append
```

Links

fetchrss: http://del.icio.us/rss/dpavlin/eeepc

- There was an error: 500 SSL negotiation failed: