

DO-Linux Dokumentation

Florian Forster
Stefan Gress
Stefan Langner
Yovan Krtolitz

17. Januar 2004

Vorbesprechung

- Kontakt über E-Mail an alle
- Stefan G. und Yovan machen sich Gedanken über die Erstellung der Bootdisk und bereiten ein Skript vor
- Stefan G. erstellt ein Skript um eine `initrd` zu erzeugen, `BusyBox`¹ soll verwendet werden
- Yovan erstellt später ein Skript um eine `BootCD` zu erzeugen
- Stefan L. und Florian werden das Installationsskript entwickeln, sobald die Bootdisk steht
- Florian erstellt ein Skript um weitere Programme und deren Libraries der `Ramdisk` auf der `CD` zur Verfügung zu stellen
- Stefan L. wird die benötigten `rpm`-Pakete zusammensuchen
- Es wird sich getroffen, um die Skripten aufeinander abzustimmen, sich gegenseitig Hilfestellung zu geben und das weitere Vorgehen zu besprechen

1. Treffen

- Stefan G. und Yovan haben das `mkbootdisk.sh`-Script vorbereitet
- Stefan G. hat das `mkinitrd.sh`-Script vorbereitet

Gemeinsam wurden die beiden Scripte getestet und einige Fehler behoben. Ausserdem wurde der Build-Prozess verändert um die Boot-Diskette mit einem Loopback-Device zu bauen. So ist es möglich, das `CD-Image` auch auf Rechnern ohne Diskettenlaufwerk zu erzeugen. Um auf der Bootdisk (und damit auch später im Speicher) Platz zu sparen wird `BusyBox` als „All-In-One“-Lösung verwendet. Alle weiteren benötigten Programme und Libraries werden auf der `CD` in einem extra Verzeichnis angelegt und anschliessend in die `RAM-Disk` eingelinkt. So können auch Programme, welche die `glibc` Verwenden (und entsprechend groß sind), verwendet werden.

¹<http://www.busybox.net/>

2. Treffen

- Stefan L. und Florian haben das `install.sh`-Script weiterentwickelt

Wieder eine Fehlerüberprüfung. Anschliessend wurde besprochen, welche Pakete verwendet werden sollen (SuSE Minimalinstallation). Die Dateien für die Boot-CD wurden festgelegt. Das Script `collect-bins.sh` von Florian und die von Stefan L. zusammengesuchten RPM-Pakete stehen zur Verfügung und werden verwendet.

3. Treffen

- Yovan hat die `mkbootcd.sh` erstellt

Die Boot-CD wurde erstellt und probeweise auf zwei Rechnern² installiert. Dabei haben sich wieder Fehler bemerkbar gemacht welche behoben wurden.

Selbst gebaute RPM-Pakete

Folgende RPM-Pakete wurden selbst erstellt:

Yovan	<code>mutt-1.4.1-1.i586.rpm</code>	E-Mail-Client für die Kommandozeile
Stefan L.	<code>tleds-1.05b.i386.rpm</code>	Anzeige von Netzwerktraffik auf den Keyboard-LEDs
Stefan G.	<code>dolinux-2.4.19-1.i486.rpm</code>	Kernel für 486er
Florian	<code>freeciv-*.rpm</code>	Freier Klon des Klassikers „Civilization“ (mangels X wird nur der Server installiert)

Der Build-Prozess

Der Build-Prozess ist anhand der Skripte zum Erstellen der CD nachvollziehbar. Nochfolgend eine Übersicht über die wichtigsten Probleme die beim Erstellen der CD aufgetreten sind:

- Es wurde BusyBox verwendet um Platz auf der Bootdisk zu sparen.
- Da auch beim Erstellen und Testen der `mkbootdisk.sh` ein Rechner ohne Floppy verwendet wurde, wird die Bootdisk mithilfe eines Loopback-Devices erstellt.
- Da viele Programme für die Installation Libraries benötigen (und diese sehr viel Speicherplatz verbrauchen), wurden diese auf CD gelegt und nachträglich eingebunden.
- Das Partitionieren ist problematisch und wird dem Benutzer überlassen. `fdisk` ist so mächtig wie kryptisch, daher ist etwas Erfahrung angebracht.

²486, 20MB RAM, 500MB IDE-Festplatte und Pentium, 128MB RAM, 6GB IDE-Festplatte

- Bei der Installation der RPM-Pakete tritt das Henne-Ei-Problem auf. Um diesem Problem zu begegnen werden in einem ersten Durchlauf alle Pakete ohne Abhängigkeiten installiert und in einem zweiten Durchlauf alle Pakete noch einmal erneuert³.
- Die Abschlusskonfiguration wird dem Script `SuSEconf` ig größtenteils überlassen. Folgende Dateien werden selbst angelegt:
 - `/etc/inittab`
 - `/etc/lilo.conf`
 - `/etc/motd`
 - `/etc/fstab`
 - `/etc/issue`

³A la `rpm --upgrade --replacepkgs <pkg>`