

Einführung, Geschichte, Einsatzzwecke

Linux-Kurs der Unix-AG

Benjamin Eberle

06. November 2014



UNIX
AG
TU Kaiserslautern

RH Regionales
Hochschul-
Rechenzentrum **RK**
Kaiserslautern

Linux' Anfang

Hello everybody out there using minix - I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. [...] I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. [...]

Linus (torvalds@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable [sic] (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-).

– Linus Torvalds in `comp.os.minix`, 26. August 1991

Entstehungsgeschichte – Kurzfassung I

- ▶ Mitte der 60er-Jahre beginnt eine Gruppe von Firmen um Bell Labs die Entwicklung eines Betriebssystems für Mainframes
- ▶ Ende der 60er steigt Bell Labs aus
- ▶ einige Bell-Labs-Angestellte verfolgen die Ziele des Projekts weiter
- ▶ Ergebnis 1969: Unics (**U**niplexed **I**nformation and **C**omputing **S**ervice)
- ▶ später Unix

Entstehungsgeschichte – Kurzfassung II

- ▶ ab Mitte der 70er-Jahre wird Unix mit Quellcode an Interessierte verteilt
- ▶ es entwickeln sich diverse Derivate
- ▶ in den 80er-Jahren wird Unix kommerziell (kein Zugang zu Quellcode, hohe Kosten)
- ▶ 1983 wird das GNU-Projekt gegründet (**G**NU's **n**ot **U**nix)
- ▶ Ziel: Ein freies Betriebssystem und freie Anwendungen entwickeln
- ▶ das GNU-Projekt entwickelte viele freie Anwendungen, aber kein Betriebssystem
- ▶ 1991 wird Linux erstmals als Hobby-Projekt veröffentlicht

Linux heute

- ▶ Linux wird von einer Vielzahl von Entwicklern weiter verbessert
- ▶ viele Entwickler sind Freiwillige, viele andere werden von Firmen für die Mitarbeit an Linux bezahlt
- ▶ Linux ist nur ein Betriebssystemkern (Kernel)
- ▶ viele der Anwendungen stammen vom GNU-Projekt: GNU/Linux
- ▶ Linux wurde auf fast jede Computer-Plattform portiert

Einsatzzwecke von Linux

- ▶ da der Quellcode offen liegt und verändert werden darf, wurde Linux an eine Vielzahl an Plattformen angepasst
- ▶ Mobiltelefone (v. a. Android)
- ▶ Eingebettete Systeme (z. B. Steuerungsanlagen, Multimedia-Systeme in Fahrzeugen)
- ▶ Thin-Clients
- ▶ Netbooks, Tablets, Laptops, Desktops
- ▶ Server, Großrechner, Hochleistungs-/Hochverfügbarkeits-Cluster

Distributionen

- ▶ ein Linux-System besteht aus vielen unabhängigen Programmen
- ▶ diese alle herunterzuladen, zu compilieren, zu installieren und aktuell zu halten ist nahezu unmöglich
- ▶ daher gibt es verschiedene Linux-Distributionen, die den Linux-Kernel und viele Anwendungen bündeln
- ▶ zur Erstinstallation gibt es einen Installer
- ▶ weitere Programme können über das Paketsystem nachinstalliert und aktualisiert werden

rpm-basierte Distributionen

- ▶ Red Hat Enterprise Linux (RHEL): Kostenpflichtige Distribution für den Einsatz in Firmen, selten neue Versionen, lange Produktunterstützung
- ▶ Fedora: Kostenlose, häufig aktualisierte „Community-Version“ von RHEL
- ▶ CentOS: Kostenloser RHEL-Nachbau
- ▶ SUSE Linux Enterprise Server/Desktop (SLES/SLED): Konkurrenz zu Red Hat
- ▶ openSUSE: SUSE für Privatleute, relativ häufig neue Versionen

Debian-basierte Distributionen

Debian

- ▶ ausschließlich von Freiwilligen entwickelte, komplett freie Distribution
- ▶ selten neue Versionen, relativ lange Produktunterstützung, sehr stabil

Ubuntu

- ▶ Debian-Ableger, wird von Canonical entwickelt
- ▶ alle 6 Monate eine neue Version, enthält aktuellere Software als Debian
- ▶ Long-Term-Support-Version (LTS) wird 5 Jahre unterstützt

Arbeiten mit Linux

- ▶ lange wurden Linux-Rechner ausschließlich über eine Textkonsole (Shell) bedient
- ▶ moderne Linux-Distributionen unterstützen auch grafische Oberflächen (X Window, X11)
- ▶ für viele Aufgaben ist die Text-Shell aber effizienter
- ▶ insbesondere für repetitive Aufgaben: hunderte Bilder skalieren, MP3-Dateien nach ID3-Tags benennen, ...
- ▶ auf Servern ist meistens keine grafische Oberfläche installiert: Textkonsole ist Pflicht

Window Manager

- ▶ fast nur Fensterverwaltung (vergrößern, verkleinern, ...)
- ▶ benötigen kaum Systemressourcen (wie Arbeitsspeicher)
- ▶ nach Einübungszeit effizientes Arbeiten durch Hotkeys, etc.
- ▶ Beispiele: fvwm, fluxbox

Desktop Environments

- ▶ viele Funktionen: Menü, Systemleiste, ...
- ▶ integrierte Zusatzprogramme: Dateimanager, Browser, ...
- ▶ hoher Ressourcenverbrauch
- ▶ Beispiele: Gnome, KDE, (Unity)

Anmeldevorgang

- ▶ Anmeldemanager fragt nach Benutzername und Passwort
- ▶ Auswahl eines Window Managers/Desktop Enviroments möglich
- ▶ nach Anmeldung: Verwendung der installierten Anwendungen (Browser, Textverarbeitung, etc.)
- ▶ alternativ: Wechsel zu einer Text-Konsole und nicht-grafische Anmeldung
- ▶ Konsole: üblicherweise Strg-Alt-F1 bis F6, F7: grafische Oberfläche
- ▶ Text-Konsole (Terminal) kann auch als Programm unter der grafischen Oberfläche gestartet werden

Text-Konsole (Shell)

- ▶ Prompt (häufig: `benutzer@rechner:~$`) zeigt an, dass die Shell auf Befehle wartet
- ▶ Befehl eingeben, mit Enter ausführen
- ▶ Befehle bestehen aus einem Programmnamen (z. B. `ls`: Dateien auflisten) und (manchmal optionalen) Parametern
- ▶ Parameter sind entweder Optionen (z. B. `-a`: auch versteckte Dateien anzeigen) oder Argumente (z. B. `/home/linux-kurs`: Dateien in `/home/linux-kurs` anzeigen)
- ▶ → `ls -a /home/linux-kurs` zeigt alle Dateien in `/home/linux-kurs` an

Navigieren in der Eingabe

- ▶ mit Pfeil-Rechts und -Links kann in der Eingabe navigiert werden
- ▶ mit Pos1 und Ende kann an den Anfang oder das Ende der Eingabe gesprungen werden
- ▶ Cursor zeigt die aktuelle Position an
- ▶ nützlich bei Tippfehlern

History

- ▶ History: speichert ausgeführte Kommandos
- ▶ mit Pfeil-Hoch und -Runter können Kommandos erneut ausgeführt werden, ohne sie wieder einzugeben
- ▶ Einträge in der History können auch verändert werden: nützlich bei Tippfehlern
- ▶ history zeigt alle Einträge der History an
- ▶ mit Strg-R kann man die History rückwärts durchsuchen

Tab-Completion

- ▶ durch Drücken der Tabulator-Taste (links neben Q) werden Befehle/Dateinamen automatisch vervollständigt
- ▶ nur möglich, wenn die bisherige Eingabe eindeutig ist
- ▶ gibt es mehrere Möglichkeiten, können diese durch doppeltes Drücken der Tab-Taste aufgelistet werden

Wichtige Steuerungszeichen

- ▶ Strg-W löscht das Wort vor dem Cursor
- ▶ Strg-U löscht alles vor dem Cursor
- ▶ Strg-D **beendet die Shell/Ende der Eingabe bei manchen Programmen**
- ▶ Strg-C **beendet lang laufende Programme**
- ▶ Strg-V ermöglicht es, ein Steuerungszeichen oder Tab einzugeben

Kopieren und Einfügen

- ▶ Strg-C und Strg-V haben in der Shell eine Sonderfunktion
- ▶ Kopieren von Text durch Markieren mit der Maus
- ▶ Einfügen mit der mittleren Maustaste, Mausrad oder gleichzeitiges Drücken von rechter und linker Maustaste

Wichtige Unterschiede zu Windows

- ▶ schon besprochen: verschiedenen Distributionen und grafische Oberflächen
- ▶ Installation von Zusatzprogrammen über das Paketsystem (Windows: Kauf von CDs/DVDs oder Download vom Hersteller)
- ▶ Updates auch über das Paketsystem (Windows: Windows-Updates und für jedes installierte Programm einzeln)
- ▶ Hardware-Treiber schon integriert (Windows: Treiber vom Gerätehersteller)
- ▶ Dateisystem-Hierarchie, keine Laufwerksbuchstaben

Dateisystem-Hierarchie

- ▶ Wurzel-Verzeichnis /
- ▶ darunter Unterverzeichnisse
- ▶ Bsp: /home/ enthält Daten der Benutzer
- ▶ /home/linux-kurs enthält Daten des Benutzers linux-kurs
- ▶ USB-Sticks, CDs, etc. werden als Verzeichnisse eingebunden
- ▶ Verzeichnisse wechseln mit
cd /pfad/zum/neuen/verzeichnis
- ▶ Zurück ins Home mit cd ohne Argument

Manpages

- ▶ Dokumentation zu Programmen, speziellen Dateien, etc. steht in Manpages
- ▶ Aufruf mit `man <programm>`
- ▶ Manpage wird mit dem Standard-Pager (meistens `less` angezeigt)
- ▶ Navigieren (hoch/runter scrollen) mit den Pfeiltasten
- ▶ Suchen mit `/suchbegriff`
- ▶ Weitersuchen mit `n`
- ▶ Beenden mit `q`
- ▶ mehr zu `less` in „Arbeiten mit der Shell Teil 2“

Gliederung

- ▶ Manpages sind in verschiedene Abschnitte gegliedert
- ▶ wichtige Abschnitte:
 - ▶ NAME: Programmname und Kurzbeschreibung
 - ▶ SYNOPSIS: Aufruf (Optionen und Argumente)
 - ▶ DESCRIPTION: Beschreibung des Programms
 - ▶ OPTIONS: Beschreibung der Optionen (manchmal unter DESCRIPTION)
 - ▶ ARGUMENTS: Beschreibung der Argumente (auch manchmal unter DESCRIPTION)
 - ▶ AUTHOR/COPYRIGHT: Angaben zum Autor/den Autoren
 - ▶ SEE ALSO: Weiterführende Verweise
- ▶ Gliederung nicht fest vorgegeben, Autor kann davon abweichen

Kapitel

- ▶ Manpages sind nach Kapiteln geordnet
- ▶ 10 Kapitel existieren
- ▶ wichtige Kapitel:
 - ▶ 1: Benutzerkommandos (z. B. `ls`, `less`)
 - ▶ 4: Gerätedateien (z. B. `/dev/null`)
 - ▶ 5: Konfigurationsdateien (z. B. `/etc/passwd`)
 - ▶ 7: Verschiedenes (z. B. Lizenzen, Internet-Protokolle)
 - ▶ 8: Kommandos für die Systemverwaltung (z. B. `useradd`)
- ▶ manchmal Manpages mit gleichem Namen in verschiedenen Kapiteln: explizite Auswahl mit `man <kapitel> <manpage>` (Beispiel: `passwd`)

apropos

- ▶ mit `apropos <stichwort>` oder `man -k <stichwort>` kann man nach Stichwörtern in Manpages suchen
- ▶ durchsucht den NAME-Abschnitt der Manpages

Weitere Dokumentationsquellen

- ▶ manche Programme (v. a. GNU) sind zusätzlich zu den Manpages auch über Info-Seiten dokumentiert:
`info <programm>`
- ▶ unter `/usr/share/doc` liegt die Dokumentation zu den installierten Paketen
- ▶ fast jedes Programm hat eine Option `-h`, `-help` oder `--help`, die eine kurze Hilfe ausgibt

Starten des Übungsystems

1. Stecke den USB-Stick in den Rechner.
2. Schalte den Bildschirm ein.
3. Schalte den Rechner ein.
4. Wenn die Installations-Meldung kommt, drücke Esc.