LINUX USER GROUP AUSTRIA
LUGA

Linux
Kernel
Basics

28. Februar 1996

Linux Kernel Basics

- ***Geschichte* — Wer, Warum und Wann**
- ***Standards* — POSIX, BSD, iBCS2**
- ***Generelle Kernel Architektur* — Micro- oder Macrokernel**
- ***Besonderheiten* — Module, Memory Management, Treiber**

Geschichte – Wer, Warum, Wann, Wie

Wer

Linus Torvalds, Student in Helsinki.

Warum

- MINIX zu beschränkt. MINIX wurde als Entwicklungs Platform verwendet, aber nicht als Basis
- Proof of Concept
  Ist die Entwicklung eines Betriebssystemes wirklich nur in großen Firmen möglich?

Wann

Erste lauffähige Version (0.01) wurde September 1991 freigegeben. Zum Installieren und Übersetzen mußte noch MINIX verwendet werden.

Erste Self Hosting Version (0.12) wurde im Jänner 1992 freigegeben.

Freigabe Version 1.0 im März 1994

Freigabe Version 1.2 im März 1995

Freigabe Version 2.0 für März 1996 geplant

Wie

- Verteilte Entwicklung:
  - Entwicklergruppen
  - Bug-Fixes
- Zentrale Versionsverwaltung und Integration durch Linus Torvalds

Standards – POSIX, iBCS2, BSD

POSIX

POSIX Standards P1003.1 und P1003.2 werden eingehalten.

POSIX ist immer das gewählte Verhalten bei Entscheidungen.

BSD

BSD ist vor allem für das TCP/IP gewählt worden. Wo BSD von den RFCs abweicht wurde BSD-Verhalten implementiert.

iBCS2

iBCS2 ist als Modul implementiert. Erlaubt das Ausführen von iBCS2 kompliant Programmen (z.B. SCO Unix).

Macrokernel

⊕ Performance

⊕ Unified Buffer Cache

⊕ `/proc`

⊖ Reboot nach Änderung

⊖ Debugging schwieriger

⊖ Abstürze häufiger

⊖ Abhängigkeiten Treiber/Kernel

Module

- Können zur Laufzeit in den Kernel eingefügt und wieder herausgenommen werden.
- Interface = exportierte Funktionen
  - Kernel
  - Module
- Kombinierbar

Memory-Management

Overcommitment — Prozesse können mehr Speicher anfordern, als vorhanden ist.

Viele Prozesse nützen Speicher nicht aus:

- Copy-On-Write (fork-exec, shared-libraries)
- FORTRAN (keine dynamische Speicherverwaltung)

Nachteile:

- Programmanpassung an knappen Speicherplatz schwierig.
- Abstürze

Abhilfe:

- Resource Limits

Kerneld

- Erlaubt load und unload von Modulen on Demand

  z.B. Laden eines Treibers beim ersten Ansprechen des Devices.

- Für alle Kernel-Module möglich. Die Zuordnung von Kernel-internen Modul Bezeichnungen und den externen Modul-Namen ist konfigurierbar.
- Spart Platz im RAM. Distributionen können bis auf die primäre Festplatte (SCSI, IDE Treiber) vereinfacht werden.

Vorschau auf v2.0

- ISDN
  ISDN Treiber für aktive (ICN) und passive (TELES) ISDN Karten. X75 und synchrones PPP werden unterstützt.
- Software RAID
  RAID0/1 und RAID5 werden im Kernel emuliert.
- IP Besonderheiten
  Firewall Funktionalitäten, Interface Aliasing, IP Accounting, Masquerading, IP Tunneling und Load Balancing machen Linux zum idealen Rechner für Netzwerkbetreiber und Internet Service Provider.
- Watchdog
  Hard- und SOftware Watchdogs sind möglich. Bei Fehler in Applikationen oder dem Kernel erfolgt ein automatischer reboot.

Ports

- 68k — Amiga, Atari TT
- Alpha — *Red Hat* Distribution
- ARM
- SPARC
- MIPS — Diskless
- PowerPC — Mach-Personality