

Die Architektur von UNIX/Linux-Betriebssystemen

Mag. Lukas Feiler, SSCP

lukas.feiler@lukasfeiler.com

http://www.lukasfeiler.com/lectures_brg9

1977: Berkeley Software Distribution/BSD (U.C. Berkeley)



1986: AIX (IBM)

1991: GNU/Linux (Open Source-Community)



1992: Solaris (früher Sun Microsystems, jetzt Oracle)

1993: FreeBSD

1996: OpenBSD

2001: MacOS X



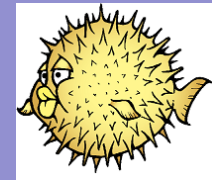
Linux-Distributionen

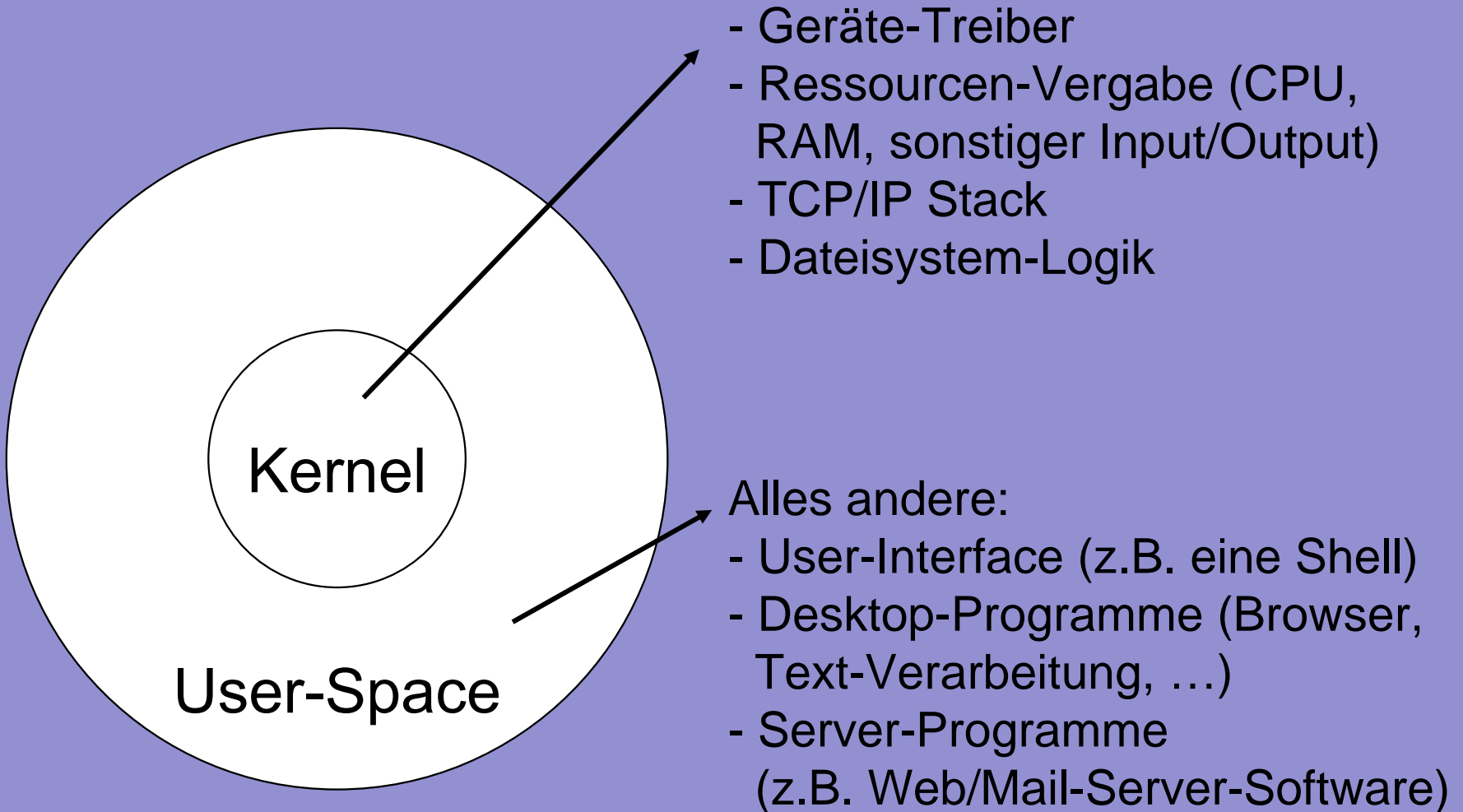
1994: SUSE Linux (jetzt Novell)

1994: RedHat Linux (RedHat)

1994: Debian (Open Source-Community)

2004: CentOS (=RedHat Linux aber gratis!)





→ GNU/Linux: GNU im User-Space, Linux im Kernel

Der Kern des Betriebssystems a.k.a. „**Kernel**“

Shell vs. Graphical User Interface (GUI)

Warum verwendet man noch eine Shell?

- Aus Sicherheitsgründen
- Aus Effizienzgründen
- Aus Performance-Gründen

GUI unter UNIX/Linux

- X Window System ("X")
- Auf X aufbauend: Desktop-Umgebungen wie GNOME od. KDE

Eine Test-Shell

- Putty herunterladen & ausführen
- Host Name: *lukas.empoweredmail.com*
- Port: 2233
- Username: *user1, user2, ..., user67, ... oder user100*
- Passwort: *******

Shell-Basics

Grundlegende Commands

`ls` Den Inhalt eines Directory auflisten
`cd` Das aktuelle Directory wechseln
`pwd` Das aktuelle Directory anzeigen
`less` sich den Inhalt einer Datei ansehen
`cat` sich den GANZEN Inhalt einer Datei ansehen
`echo` etwas ausgeben
`mkdir` Ein Directory erstellen (make directory)

Mit Optionen

`ls -l`
`ls -la` (= `ls -l -a`)

Mit Argumenten

`ls -l /home`
`less /tmp/test.txt`

User-Interfaces

Die Grundsätze der UNIX/Linux-Architektur

- 1) Datei-basiert
- 2) Multi-User
- 3) Multi-Tasking

Die zentralen Eigenschaften von UNIX

UNIX/Linux ist ein ...

1) Datei-basiertes Betriebssystem

→ *alles* ist eine Datei

z.B. ein Gerät wie eine Festplatte (`ls -l /dev/sda`)

z.B. eine Soundkarte (`ls -l /dev/audio`)

z.B. ein Verzeichnis! (Directory)

→ Datei-basierte Systemsicherheit: jede Datei hat „permissions“
read (r), write (w), execute (x)

```
cat /tmp/secret-even.txt
```

```
cat /tmp/secret-uneven.txt
```

Das UNIX-Dateisystem:

/ „root“ (nicht C:\, D:\ etc.)

/var/log logfiles

/bin, /usr/bin/ binaries (ausführbare Commands)

/tmp Temporäre Dateien

/dev devices (Geräte)

Die zentralen Eigenschaften von UNIX

UNIX/Linux ist ein ...

2) Multi-User Betriebssystem

→ mehrere User können gleichzeitig eingeloggt sein

Im Unterschied zu z.B. DOS, Windows 95, Mac OS 9

→ User-Rechte/File Permissions für Owner & Group

```
ls -l /tmp/test.txt
```

```
-rwx rw- r-- 1 root sys 81 Oct 19 17:08 /tmp/test.txt
```

Diagram illustrating the components of the command output:

- `-rwx` → owner
- `rw-` → group
- `r--` → others
- `1` → owner
- `root` → owner
- `sys` → group
- `81` → bytes
- `Oct 19 17:08` → last modified

Berechtigungen ändern

```
chgrp your_group test.txt (Gruppe ändern)
```

```
chmod g+r test.txt (Gruppe read-Rechte geben)
```

```
chmod g-r test.txt (Gruppe read-Rechte nehmen)
```

```
chmod o+r test.txt (Allen read-Rechte geben)
```

→ Es gibt einen „Superuser“: **root**

Die zentralen Eigenschaften von UNIX

UNIX/Linux ist ein ...

3) Multi-Tasking Betriebssystem

→ mehrere Programme können gleichzeitig laufen

Im Unterschied zu z.B. DOS, Windows 95, Mac OS 9

→ Ressourcen-Vergabe durch den Kernel

in Action:

```
top
```

```
ps -aux
```

```
kill    (um das evil_spammer.sh „Problem“ zu lösen 😊)
```

Die zentralen Eigenschaften von UNIX

Erweiterte Shell-Kenntnisse

Was ist eigentlich ein Command?

→ eine Datei!

Aber wo sind Commands gespeichert? Wie findet sie die Shell?

→ `echo $PATH`

→ Commands selbst finden:

`which ls`

Wie lerne ich mehr über ein Command

→ `ls --help`

→ `man ls` (Linux/UNIX kommt mit man-uals!)

Suchen mittels “/Suchbegriff“

Erweiterte Shell-Kenntnisse #2

Output-Redirection – The Power of the Shell!

- Zeilen filtern: `cat /etc/passwd | grep lukas`
- Zeilen sortieren: `cat /etc/passwd | sort`
- Nur ersten 5 Zeilen zeigen: `cat /etc/passwd | head -n 5`
- Nur letzten 5 Zeilen zeigen: `cat /etc/passwd | tail -n 5`

Output in eine Datei umleiten:

```
echo This is a test > test.txt
```

Zeilen numerieren:

```
cat -n /etc/passwd
```

Weitere wichtige commands

`who` Wer ist eingeloggt

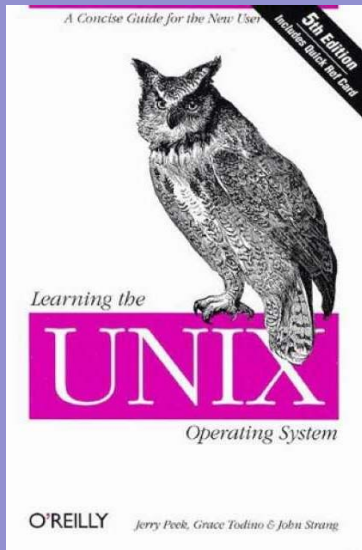
`find` Dateien finden;

z.B. `find /home/lukas -name '*.txt'`

UNIX-Rätsel

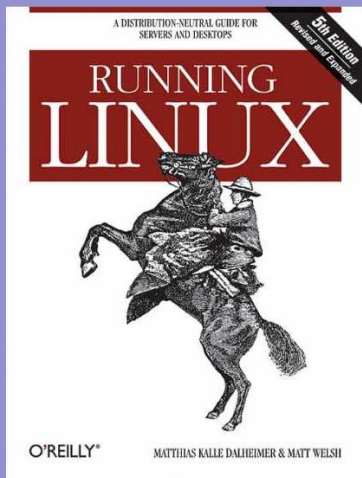
- 1) Mit welchem Befehl schreibt man in eine Datei (/tmp/\$USER) das aktuelle Datum?
Hinweis: `date` und `>`
- 2) Wie viele *.txt Dateien enthält das Verzeichnis /tmp/count?
Hinweis: `ls`, `cat` (für Fortgeschrittene: geht das auch mit `wc`?)
- 3) Wie lautet eine Befehlszeile, die alle existierenden User-Accounts des Systems auflistet, die "mail" im Namen enthalten?
Hinweis: `cat`, `grep`, /etc/passwd (f. Fortgeschr.: nur 1. Spalte: `cut`)
- 4) Wie lautet eine Befehlszeile, die nur die erste *.txt Datei aus /tmp/count auflistet? (für Fortgeschrittene: auch mit `find`?)
Hinweis: `head`
- 5) Wieviele *.txt Dateien enthält /tmp/count einschließlich der Sub-Directories?

Wer mehr will:



Learning the UNIX Operating System
(O'Reilly, 5th edition 2002)

UNIX, Ein praktischer Einstieg
(O'Reilly, 1. Auflage 2002)



Running Linux
(O'Reilly, 5th edition 2005)

Linux Praxishandbuch
(O'Reilly, 1. Auflage 2005)

online unter:

<http://www.oreilly.de/german/freebooks/runux5ger/>