FOSS-Lizenzen und -Anwendungen, ICT, Hardware

Linux-Kurs der Unix-AG

Andreas Teuchert

22. Juli 2014

Hardwarekomponenten

- Mainboard
- CPU
- RAM
- Grafikkarte
- Festplatte
- Optische und Wechsel-Datenträger
- Peripherie

Mainboard

- verbindet verschiedene Komponenten mit der CPU
- besitzt Sockel/Slots/Ports für
  - CPU
  - RAM
  - PCI-/AGP-/PCIe-Karten
  - IDE (PATA), SATA
  - USB
  - Parallel-/Seriell-Port
- enthält das BIOS (Basic Input/Output System)

CPU

- **C**entral **P**rocessing **U**nit
- für die Datenverarbeitung zuständig
- Desktop- und Notebook-Systeme: üblicherweise eine CPU
- Server-Systeme: meist mehrere CPUs
- verschiedene Architekturen
  - x86 - 32-Bit-Architektur
  - x64 - 64-Bit-Architektur
  - ARM-Architektur

RAM

- **R**andom **A**ccess **M**emory (beliebig zugreifbarer Speicher)
- flüchtiger Zwischenspeicher für Datenverarbeitung
- Desktop- und Notebook-Systeme: üblicherweise 4 bis 8 GB
- Server-Systeme: unterschiedlich, je nach Anforderung

Grafikkarte

- auf Mainboard bzw. in CPU integriert
- dedizierte Grafikkarten per PCI-, AGP- oder PCIe-Slot
- übernehmen für die Rechenoperationen für Grafikausgabe
- haben meistens mehrere Monitoranschlüsse (VGA, DVI, HDMI, DisplayPort)
- HDMI auch zur Audioübertragung

Festplatte

- nicht-flüchtiger Speicher für Daten und Programme
- meist SATA-Anschluss, ältere Festplatten IDE
- Magnetspeicher:
  - hohe Kapazitäten (heutzutage bis zu 4TB pro Festplatte)
  - rotierende Magnetscheibe, beweglicher Schreib-Lese Kopf
- Flashbasierte Speicher (**S**olid **S**tate **D**isk, SSD):
  - keine beweglichen Teile, geringer Energieverbrauch
  - kleine Kapazitäten, begrenzte Schreibzyklen

Partitionierung

- Festplatten werden in Partitionen aufgeteilt
- bis zur vier primäre Partitionen
- eine primäre Partition kann als erweiterte Partition beliebig viele logische Partitionen beinhalten
- Festplatten unter `/dev` als `hdX` (IDE) oder `sdX` (SATA)
- `X`: fortlaufender Buchstabe
- Partitionsnummer wird an den Festplattennamen angehangen (1-4: primär, ab 5: logisch)
- Beispiel: `/dev/sdd3`: dritte primäre Partition auf der vierten SATA-Festplatte

Optische und Wechsel-Datenträger

Optische Datenträger

- Compact Disk (CD), üblich 700MB
- Digital Versatile Disc (DVD), bis zu 9,4GB
- Blu-ray Disc, unterschiedliche hohe Kapazitäten

Wechsel-Datenträger

- Diskette
- Flashspeicher, USB-Sticks oder Speicherkarten,
- Externe Festplatten (mittels eSATA, Firewire oder USB)

Peripherie

- Ein- oder Ausgabegeräte (I/O)
  - Maus, Tastatur
  - Drucker, Scanner
  - Monitor
- zusätzlicher Speicher (externe Festplatten)
- meist über **U**niversal **S**erial **B**us (USB) angeschlossen

Bootloader

- befindet sich im **M**aster **B**oot **R**ecord (MBR) der Festplatte
- wird als erstes nach dem BIOS geladen
- lädt dann nächste Stufe des Bootloader von der Festplatte um weitere Funktionalitäten für den Bootvorgang bereitzustellen
- z.B. mit GRUB: Betriebssystemwahl möglich (verschiedene Kernel-Versionen, Windows)
- lädt nach Auswahl das eigentliche Betriebssystem
- bei neueren Rechnen kommt UEFI statt BIOS zum Einsatz, daher anderer Bootvorgang

Gerätedateien - `/dev/`

- Schnittstelle zur Hardware
- bekannte Beispiele
  - Festplatten - `/dev/sd*`, früher `/dev/hd*` für IDE-Festplatten
  - CD-ROM - `/dev/sr0`
  - Textkonsolen - `/dev/tty*`
- USB-Sticks werden wie SATA-Festplatten behandelt (`/dev/sd*`)

Kernel-Module

- Treiber für Geräte
- werden meistens von Linux-Entwicklern programmiert
- für fast jedes Gerät existieren Treiber
- werden in der Regel automatisch geladen

Hardware und Kernel-Module erkennen

- `lspci` und `lsusb` listet PCI- bzw. USB-Geräte auf
  - `-t` zeigt eine Baumstruktur an
  - `-v[vv]` gibt zusätzliche Informationen aus, funktioniert bei `lsusb` nicht mit `-t`
- `lsmod` zeigt geladene Kernel-Module an
- `modprobe` lädt neue Kernel-Moduke
- `modprobe -r` entfernt geladene Kernel-Module
- veraltet: `insmod` bzw. `rmmod`

Hardware und Kernel-Module erkennen

- `hwinfo` sucht nach sämtlicher angeschlossener Hardware
- Hardware-Typen können mittels Parameter angegeben werden:
  - `--cpu`: Prozessor(en)
  - `--disk`: Festplatten, USB-Sticks
  - `--gfxcard`: Grafikkarte
  - `--mem`: RAM
  - `--netcard`: Netzwerk-Adapter

Spickzettel

Alle Befehle

Befehl	Optionen
`lsusb`	`-t`, `-v`
`lspci`	`-v`
`lsmod`	
`modprobe`	`-r`
`hwinfo`	`--cpu`, `--disk`
	`--gfxcard`, `--mem`, `--netcard`

Kommerzialisierung des Softwaremarkts in den 80ern

- bis in die 70er/80er Jahre: Software wird (meist mit Quellcode) formlos verteilt
- Weitergeben und/oder Verändern von Software ist normal
- 80er Jahre: Firmen kommen auf die Idee, Software zu verkaufen
- Software wird nun ohne Quelltext ausgeliefert („Closed Source“)
- Weitergeben und Verändern nicht erlaubt

Nachteile von Closed-Source-Software

- Weiterbildung durch Studieren des Quelltextes nicht möglich
- Anpassen des Programms an eigene Bedürfnisse kaum möglich
- Verbesserungen können nur vom Hersteller vorgenommen werden

Gründung von GNU und der FSF

- vor allem Forscher stören sich an den Einschränkungen von Closed-Source-Software
- einer von ihnen ist Richard Stallman
- 1983: Gründung von GNU
- 1985: Gründung der Free Software Foundation (FSF)
- Ziele der FSF:
  - Unterstützung des GNU-Projekts
  - „Lobbyarbeit“ für freie Software

Definition freier Software nach FSF

Freie Software muss vier Freiheiten einräumen:

0. das Programm für jeden Zweck verwenden
1. die Funktionsweise des Programms untersuchen (z. B. Quellcode lesen) und es anpassen
2. das Programm weitergeben
3. das Programm verbessern und die Verbesserungen veröffentlichen

Freie Software vs. Open Source

- „Freie Software“ ist nicht eindeutig
- „frei“ wird häufig mit „kostenlos“ gleichgesetzt (Beispiel: „Freeware“)
- Frei wie Freibier vs. Frei wie Freie Rede
- viele Leute bevorzugen daher den Begriff „Open Source“
- „Open Source“ bedeutet strenggenommen nur, dass der Quellcode offenliegt
- daher auch kein treffendes Schlagwort

Gründung der OSI

- 1998: Gründung der Open Source Initiative (OSI)
- Gründer: Bruce Perens und Eric Raymond
- gleiche Ziele wie die FSF, aber weniger radikal
- Open Source statt Freie Software als zentraler Begriff

FOSS und FLOSS

- sowohl „Freie Software“ als auch „Open Source“ haben als Schlagwörter ihre Nachteile
- Kompromiss: FOSS bzw. FLOSS
- FOSS: **F**ree and **O**pen **S**ource **S**oftware
- FLOSS: **F**ree, **L**ibre and **O**pen **S**ource **S**oftware

Wofür Lizenzen?

- Software ist grundsätzlich durch das Urheberrecht geschützt
- oder durch entsprechende Gesetze in anderen Ländern (Copyright, etc.)
- Verwendung der Software setzt Erlaubnis des Rechteinhabers voraus (Lizenz)

Unterschiede zwischen proprietären und freien Lizenzen

Proprietäre Lizenzen

- schränken die Rechte des Nutzers i. d. R. ein (kein Weiterverkauf, Nutzung nur für bestimmte Zwecke)
- Nutzer muss vorher zustimmen (Vertrag, EULA)

Freie Lizenzen

- gestehen dem Nutzer zusätzliche Rechte zu (Veränderung, Weitergabe)
- kein Nachteil für den Nutzer
- daher keine Zustimmung nötig

General Public License (GPL)

- am weitesten verbreitete FOSS-Lizenz
- bei der Weitergabe von GPL-Software müssen dem Empfänger auch die ihm nach der GPL zustehenden Rechte eingeräumt werden (Copyleft)
- dafür muss der Quellcode mit weitergegeben werden (oder im Nachhinein angefordert werden können)
- GPL macht keine Vorschriften über den Preis
- GPL-Software kann also verkauft werden, der Käufer darf sie aber beliebig oft weiterverkaufen oder -schenken

BSD-Lizenz

- ursprünglich für BSD-Unix verwendet
- erlaubt die Weitergabe solange der Lizenz-Text erhalten bleibt
- bei der Weitergabe muss der Quelltext nicht mit weitergegeben werden (kein Copyleft)
- die ursprüngliche Version forderte eine Anerkennung des Original-Autors in Werbetexten

Creative Commons

- viele Lizenzen (wie GPL und BSD) sind für Software ausgelegt
- für Texte, Bilder, Musik, etc. nicht geeignet
- Creative-Commons-Lizenzen können auch für solche Werke verwendet werden
- Autor kann auswählen, ob das Werk für kommerzielle Zwecke benutzt werden darf und ob es in veränderter oder unveränderter Form weitergegeben werden darf
- das Werk kann auch gemeinfrei (Public Domain) gemacht werden (Verzicht auf alle Urheberrechte, soweit möglich)

Geld verdienen mit FOSS

- Geld verdienen durch den Verkauf von Einzellizenzen funktioniert bei FOSS nicht
  - Kunden dürfen Software einfach weitergeben
  - Konkurrenten dürfen das Produkt kopieren und selbst verkaufen
- Open-Source-Geschäftsmodelle:
  - Support
  - Schulungen
  - individuelle Weiterentwicklungen
  - kostenpflichtige (eventuell unfreie) Zusatzfunktionen

FOSS-Anwendungen

Anwendungsgebiete

- Desktop
  - Büro
  - Multimedia
- Server
  - Web
  - Infrastruktur
- Programmierung/Entwicklung
  - Programmiersprachen
  - Entwicklungsumgebungen (IDE)
- Mobil
  - Apps

Büro-Anwendungen

Textverarbeitung, Tabellenkalkulation, etc.

- OpenOffice.org
  - wurde von Sun entwickelt
  - Sun wurde 2010 von Oracle aufgekauft
  - Übergabe an die Apache Software Foundation
  - wird weiterentwickelt, Zukunft derzeit jedoch unklar
- LibreOffice
  - entstand 2010 aus dem OpenOffice.org Projekt
  - „Nachfolger“ von OpenOffice.org
- Closed-Source Anwendungen
  - „Microsoft Office“
  - „Apple iWork“

Büro-Anwendungen

Browser

- Firefox
  - wird von der Mozilla Foundation entwickelt
  - beliebter Browser
- Chromium
  - Chromium verwendet Teile von „Google Chrome“
  - unter Linux-Distributionen wird meist Chromium verwendet
- Links, Lynx, w3m
  - text-basierte Browser für die Kommandozeile
- Closed-Source Anwendungen
  - „Microsoft Internet Explorer“
  - „Apple Safari“

Büro-Anwendungen

E-Mail Client

- Thunderbird
  - wird von der Mozilla Foundation entwickelt
  - beliebter E-Mail Client
  - zahlreiche Addons, z.B. Kalenderfunktion, LDAP-Adressbücher, etc
- Closed-Source Anwendungen
  - „Microsoft Outlook (Express)“
  - „Apple Mail“

Multimedia-Anwendungen

Audio

- Audacity
  - Audio-Schnittprogramm
- Amarok, Audacious, Banshee, Clementine, Rhythmbox, XMMS
  - Audio-Abspielprogramme

Video-Bearbeitung

- Blender
  - 3D-Render-Software
- Cinelerra
  - Video-Schnittprogramm

Multimedia-Anwendungen

Bild-Bearbeitung

- The GIMP
  - vielfältige Bild-Bearbeitungs-Software
  - Pendant zu „Adobe Photoshop“
- Inkscape
  - Bild-Bearbeitungs-Software zur Erstellung von Vektorgrafiken
  - Pendant zu „Adobe Illustrator“
- ImageMagick
  - konvertiert Bild-Dateien zwischen verschiedenen Formaten
  - skriptgesteuerte Bild-Manipulation möglich

Server-Dienste

HTTP-Server

- Apache
  - beliebter HTTP-Server
  - vielseitig erweiterbar durch zahlreiche Module
  - LAMP: **L**inux **A**pache **M**ySQL **P**HP (oder Perl/Python)

Mail-Server

- Postfix
  - beliebter und leistungsfähiger Mail-Server

Server-Dienste

Datenbank-Server

- MySQL
  - relationaler Datenbank-Server
  - für Websites gut geeignet und oft verwendet
- PostgreSQL
  - sehr leistungsfähiger Datenbank-Server
  - für verschiedene Anwendungen geeignet

Samba-Server

- Datei- und Druck-Server in gemischten Umgebungen, insbesondere Windows

Server-Dienste

NFS-Server

- Datei-Server für Unix-/Linux-Systeme
- Netzwerkdateisystem

OpenLDAP-Server

- Verzeichnisdienst
- Verwendung zur Authentifizierung, Adressbuch, etc.

Server-Dienste

DNS-Server

- wichtiger Dienst in jedem Netzwerk
- führt Namensauflösung durch

DHCP-Server

- wichtiger Dienst in Netzwerken
- Vergabe von Netzwerk-Adressen

Programmierung

Programmiersprachen

- C, C++
- Java

Skriptsprachen

- Perl
- PHP
- Python
- Shell

Entwicklungsumgebungen

Editoren

- vi, vim (Konsolen-basiert)
- gedit, kate (grafisch)

IDEs

- Eclipse
- Netbeans

Virtualisierung

- Virtualisierungslösungen erlauben es, mehrere virtuelle Maschinen (VMs) auf einem physikalischen Rechner laufen zu lassen
- Einsatzzwecke:
  - Verwendung mehrerer Betriebssysteme parallel
  - Betrieb von Serverdiensten in verschiedenen virtuellen Maschinen: erhöhte Sicherheit durch Trennung der Dienste, trotzdem weniger Hardwareaufwand, da mehrere VMs auf einem physikalischen System laufen
  - Ausfallsicherheit: VMs können sehr schnell auf ein anderes physikalisches System umgezogen werden
- moderne Prozessoren unterstützen hardwarebasierte Virtualisierung, dadurch nur minimale Performanzverluste gegenüber echter Hardware

Virtualisierungslösungen (Auswahl)

- KVM (frei)
- Xen (frei)
- OpenVZ (frei)
- Microsoft Hyper-V (unfrei)
- VMware Workstation (unfrei)
- VMware ESXi/vSphere (unfrei)
- Oracle VM VirtualBox (freie und unfreie Version erhältlich)
- diverse Verwaltungslösungen: libvirt, Ganeti, Proxmox, ...

Benutzung von Webbrowsern und Privatsphäre

- Webbrowser speichern standardmäßig alle aufgerufenen Seiten im Verlauf (History)
- Verlauf kann möglicherweise von anderen Benutzern des Rechners betrachtet werden
- Webseiten können Informationen in Form von Cookies dauerhaft auf dem Rechner speichern und diese bei späteren Besuchen wieder auslesen, z. B. um festzustellen, ob sich ein Benutzer angemeldet hat
- Cookies werden häufig von Werbediensten verwendet um Profile über den Benutzer anzulegen (Tracking)
- Werbedienste verwenden auch andere Informationen (Browserversion, IP-Adresse, ...) zum Tracking
- die meisten Browser bieten einen privaten Modus (auch Inkognito-Modus genannt): Verlauf und Cookies werden nur gespeichert bis der Browser geschlossen wird

Cloud-Computing I

- Cloud-Computing stellt Dienste (Daten, Anwendungen, Rechenleistung, ...) über das Netzwerk bereit, sodass von überall mit verschiedenen Endgeräten darauf zugegriffen werden kann
- die dafür nötige Infrastruktur ist für den Benutzer nicht sichtbar, sie wird nur als „Wolke“ wahrgenommen
- Daten können auf beliebigen Servern an beliebigen Orten gespeichert sein
- finanzielle Vorteile durch einfacherere IT-Infrastruktur auf Nutzerseite

Cloud-Computing II

- bei kostenlosen Cloud-Diensten finanziert der Anbieter sich üblicherweise durch Auswertung der Nutzerdaten und -aktivitäten um diese für angepasste Werbung zu verwenden oder zu verkaufen
- auch bei kostenpflichtigen Angeboten kann der Zugriff auf Nutzerdaten durch Dritte nicht ausgeschlossen werden (z. B. aufgrund von Sicherheitslücken, finanzieller Interessen des Anbieters oder rechtlicher Vorgaben [vgl. PRISM])

Passwörter

- Passwörter dienen zur Legitimation des Benutzer gegenüber einem Dienst
- werden dazu auf Seite des Dienstes in einer Datenbank gespeichert
- gespeicherte Passwörter werden i. d. R. gehasht gespeichert (aus dem Hash lässt sich nicht das ursprüngliche Passwort berechnen)
- Angriffe erfolgen üblicherweise mittels Bruteforce (sehr viele mögliche Passwörter werden ausprobiert)
- kurze und einfache Passwörter können mittels Bruteforce sehr schnell ermittelt werden
- sinnvoll: mindestens acht zufällige Zeichen, Kleinbuchstaben, Großbuchstaben, Zahlen, Sonderzeichen
- für verschiedene Dienste verschiedene Passwörter verwenden