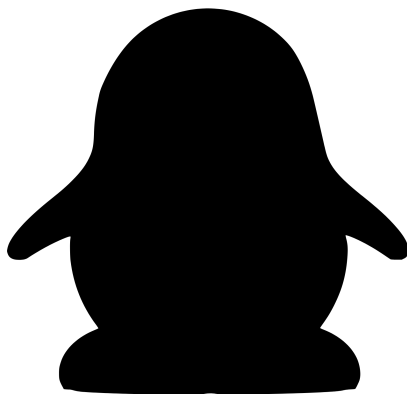




inovex

inovex classes

Linux



Session 2

Januar 2019

Agenda

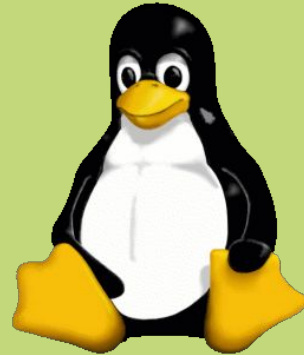
17:00 - Recap Session 1

17:10 - Bash Basics

17:40 - Benutzer & Benutzerverwaltung

18:10 - Paketverwaltung

Recap Session 1

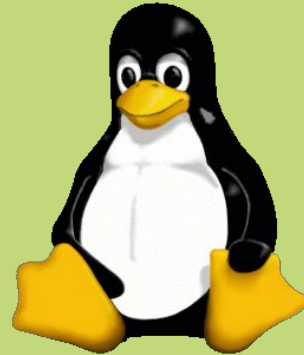


Recap Session 1

Aktualisierte Aufgabenstellung:

<https://github.com/inovex/linux-class>

Bash Basics



Bash Basics

Basics

Autovervollständigung: TAB-TAB

Abbrechen eines Befehls (eigentlich Kill des im Vordergrund laufenden Prozesses)

- STRG/CTRL + C

Schließen der Bash (äquivalent zu "exit")

- STRG/CTRL + D

Prozess im Vordergrund in den Hintergrund schieben

- STRG/CTRL + Z

Zurückholen eines Programms in den Vordergrund:

- fg <proc-name>

Bash Basics

Basics

Screen leeren:

- STRG/CTRL + L

Ans Zeilenende hüpfen:

- STRG/CTRL + E

An den Zeilenanfang springen:

- STRG/CTRL + A

Bash Basics

History

Einsehen aller vergangenen Befehle:

- history
- history | grep "ssh" (suchen)

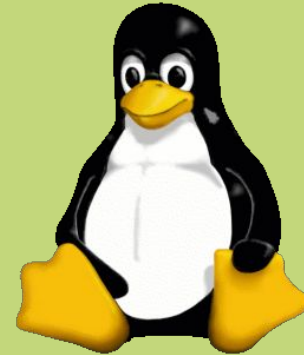
Navigieren durch abgesetzte Befehle:

- Pfeil hoch / runter

Reverse-Search:

- STRG/CTRL + R: und anfangen zu tippen
- STRG/CTRL + O: Befehl erneut absetzen
- STRG/CTRL + G (oder + C): zum abbrechen

Benutzerverwaltung und Dateirechte



Benutzerverwaltung und Dateirechte

Übersicht

- Multi-Tasking / Multi-User
- Grundlagen
- Wichtige Befehle
- Hinweise

Benutzerverwaltung und Dateirechte

Multi-Tasking / Multi-User

Multi-Tasking

- mehrere Prozesse (Tasks) gleichzeitig ausführen
- früher (80ths) nur ein (1) Prozess möglich (z.B. MS-DOS)

Multi-User

- mehrere Benutzer (User) arbeiten gleichzeitig auf dem selben System

Benutzerverwaltung und Dateirechte

Grundlagen

- generelle Aufteilung in “einfache” Benutzer (wenig Rechte, weniger Gefahr ein System “zu zerstören”) und Systemverwalter (aka “root”, alle Rechte)
- Benutzer (User) werden in Gruppen (Groups) zusammengefasst
- Systeminterne Rechtezuweisung an User-ID (UID) und Gruppen-ID (GID)
- Möglichkeit temporär auf Daten anderer User zuzugreifen (Effective UID / EUID, Effective GID / EGID)

Benutzerverwaltung und Dateirechte

Grundlagen

Beispiel Passwort neu setzen als normaler User:

```
$ grep ubuntu /etc/passwd
```

```
ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash
```

```
$ ls -l /etc/shadow
```

```
-rw-r----- 1 root shadow 652 Jan 22 19:22 /etc/shadow
```

```
$ ls -l /usr/bin/passwd
```

```
-rwsr-xr-x 1 root root 54256 May 16 2017 /usr/bin/passwd
```

Benutzerverwaltung und Dateirechte

Wichtige Befehle

ls -l

- anzeigen der Dateirechte

```
$ ls -la /home/ubuntu/
```

```
drwxr-xr-x 2 ubuntu ubuntu 4096 Jan 28 13:36 .  
drwxr-xr-x 1 root root 4096 Jan 28 13:33 ..  
-rw-rw-r-- 1 ubuntu ubuntu 0 Jan 28 13:33 leere_datei.txt
```

Benutzerverwaltung und Dateirechte

Wichtige Befehle

stat

- anzeigen des Dateistatus

```
$ stat leere_datei.txt
```

```
File: leere_datei.txt
Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 801h/2049d Inode: 284794     Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1000/  ubuntu)   Gid: ( 1000/  ubuntu)
Access: 2019-01-29 06:59:19.045767587 +0000
Modify: 2019-01-29 06:59:19.045767587 +0000
Change: 2019-01-29 07:10:17.916757562 +0000
```

Benutzerverwaltung und Dateirechte

Wichtige Befehle

id

- Ermitteln der effektiven UIDs und GIDs

```
$ id  
uid=1000 (ubuntu) gid=1000 (ubuntu)  
groups=1000 (ubuntu),27 (sudo)
```


Benutzerverwaltung und Dateirechte

Wichtige Befehle

sudo

- temporär die effektive UID und GID wechseln

```
$ sudo id
```

```
uid=0 (root) gid=0 (root) groups=0 (root)
```

Benutzerverwaltung und Dateirechte

Wichtige Befehle

chmod

- ändern der Dateirechte

```
$ ls -l leere_datei.txt  
-rw-rw-r-- 1 ubuntu ubuntu 0 Jan 28 13:33 leere_datei.txt
```

```
$ chmod u=rw-,g=---,o=--- leere_datei.txt  
(oder: chmod 0600 leere_datei.txt )
```

```
$ ls -l leere_datei.txt  
-rw----- 1 ubuntu ubuntu 0 Jan 28 13:33 leere_datei.txt
```

Benutzerverwaltung und Dateirechte

Wichtige Befehle

chown

- ändern des Datei-Eigentümers oder -Gruppe

```
$ ls -l
```

```
-rw-rw---- 1 ubuntu sudo 0 Jan 28 13:33 leere_datei.txt
```

```
$ chown ubuntu:ubuntu leere_datei.txt
```

```
$ ls -l
```

```
-rw-rw---- 1 ubuntu ubuntu 0 Jan 28 13:33 leere_datei.txt
```

Benutzerverwaltung und Dateirechte

Wichtige Befehle

who | whoami

- aktuelle Benutzer im System anzeigen

```
$ who
```

```
jbrunk    console  Dec 18 14:34  
jbrunk    ttys000   Dec 18 14:37  
jbrunk    ttys001   Dec 18 14:37
```

```
$ whoami
```

```
jbrunk
```

Benutzerverwaltung und Dateirechte

Wichtige Befehle

useradd

- neuen Benutzer anlegen

usermod

- existierenden Benutzer ändern

groupadd

- neue Gruppe anlegen

groupmod

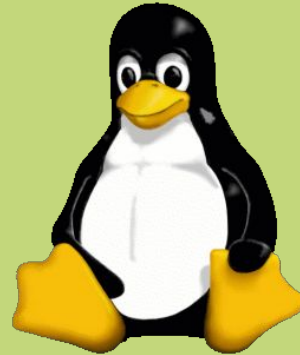
- existierende Gruppe ändern

Benutzerverwaltung und Dateirechte

Hinweise

- grundsätzlich nicht als **root** auf einem System arbeiten, nur temporär das **sudo** Kommando verwenden
- alle Dienste (zB. Webserver) mit möglichst **wenig** Rechten / Privilegien betreiben
- Benutzern nicht **zu viele** Rechte geben (nicht alle müssen **root** sein)

Paketverwaltung



Paketverwaltung

Wie wird unter Linux Software installiert?

- › **Kompilieren von Source**
 - › aufwändig, fehleranfällig
- › **Installation von Binärpaketen**
 - › z.B. unter debian-basierten Systemen mit “*dpkg*”
 - › keine Kontrolle über Abhängigkeiten
- › **Installation via Paketverwaltung**

Paketverwaltung

Eigenschaften

- › Laden von Paketen
- › Auflösung von Abhängigkeiten
- › Distributionsspezifische Paketierung
 - › Einheitliche Installations- und Konfigurationspfade
 - › Systemspezifische Anpassungen
- › Qualitätssicherung

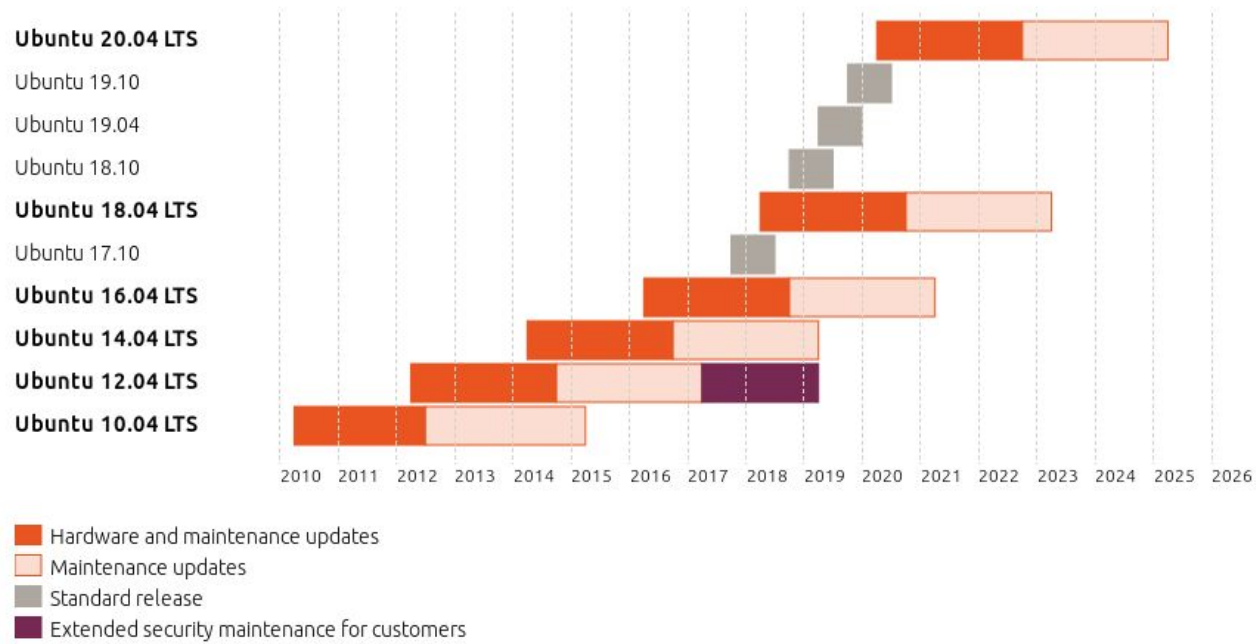
Paketverwaltung

Prominente Paketverwaltungen

Distribution	Manager	Format
Debian/Ubuntu	apt, aptitude	deb
RHEL/Centos/Fedora	yum, dnf	rpm
(open)SUSE	YaST, Zypper	rpm
Arch Linux/Manjaro	pacman	tar.xz

Paketverwaltung

Ubuntu Lebenszyklus



Paketverwaltung

Paketquellen

- › **Distributionsquellen:**
 - › mehr Sicherheits- als Versionsupdates
 - › ggf. Anpassungen der Pakete ans System
- › **Partnerquellen:**
 - › erweitern häufig Distributionsquellen um proprietäre Pakete
- › **Drittanbieter:**
 - › häufig Betreiber von Anwendungen oder Community
 - › Sicherheit, Stabilität und Aktualität variieren

Paketverwaltung

Wichtige Befehle beim Verwalten von Paketen

- › Aktualisierung der Paket Metadaten
 - › regelmäßig vor einer Installation durchführen
- › Installation von Paketen
 - › inklusive von Abhängigkeiten
- › Aktualisierung von Paketen
 - › manuell und automatisiert
- › Deinstallation von Paketen
 - › in der Regel bleibt Konfiguration bestehen

Paketverwaltung

Ziel

Installation und **Betrieb** des Webserver **NGINX** unter **Ubuntu**

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Paketverwaltung

Ubuntu: Paketverwaltung mit APT

“**Apt** is a **collection of tools** distributed in a package named *apt*.”

Tools: *apt-get*, *apt-cache*, *add-apt-repository*,...

Command: *apt* -> high level cli for these tools

Manuelles anpassen der Paketquellen:

Add sources in `/etc/apt/sources.list.d/*`:

`/etc/apt/sources.list.d/nginx.list`:

```
## Replace $release with your corresponding Ubuntu release.  
deb http://nginx.org/packages/ubuntu/ $release nginx  
deb-src http://nginx.org/packages/ubuntu/ $release nginx
```

Paketverwaltung

Ubuntu: Wichtige Befehle (1/3)

- › `$ apt-get update`
 - › Aktualisierung der Paket Metadaten
- › `$ apt-cache search <Suchbegriff>`
 - › Suche nach Paketen
- › `$ apt-cache show <Paketname>`
 - › Informationen über das Paket
- › `# apt-get upgrade`
 - › Aktualisierung aller installierten Pakete

Paketverwaltung

Ubuntu: Wichtige Befehle (2/3)

- › **# apt-get remove <Paketname(n)>**
 - › Deinstallation von Paketen
 - › Hinweis: Konfigurationen und ungenutzte Abhängigkeiten bleiben bestehen
 - › *--purge*: Entfernt auch die Konfigurationen
- › **# apt-get purge <Paketname(n)>**
 - › Macht das gleiche wie *apt-get remove --purge*
- › **# apt-get autoremove**
 - › Entfernt ungenutzte Abhängigkeiten
 - › *--purge*: Entfernt auch die Konfigurationen

Paketverwaltung

Ubuntu: Wichtige Befehle (3/3)

- › **# apt-add-apt-repository** <Paketquelle>
 - › Hinzufügen von weiteren Paketquellen
 - › *--remove*: Entfernen von Paketquellen
 - › auch manuell möglich durch Anpassung der Quellen unter */etc/apt/sources.list.d/*
- › **# apt-key add** <Keyfile>
 - › *sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys <keyId>*: Durchsucht den Ubuntu Key-Server nach öffentlichen Schlüssel und verifiziert ihn

Paketverwaltung

systemd (1/2)

“... provides a system and service manager that runs as PID 1 and starts the rest of the system”

Grundlegende Befehle:

- # **systemctl start** *<service>*: Startet den Dienst.
- # **systemctl stop** *<service>*: Beendet den Dienst.
- # **systemctl restart** *<service>*: Startet den Dienst neu.
- # **systemctl reload** *<service>*: Lädt die Konfigurationen des laufenden Dienstes neu.
- # **systemctl status** *<service>*: Fragt den Zustand des Dienstes ab.

Paketverwaltung

systemd (2/2)

Weitere Befehle:

*systemctl enable* <service>:

Fügt den Dienst zu der Liste der beim Hochfahren des Systems zu startenden Dienste hinzu.

*systemctl disable* <service>:

Entfernt den Dienst von der Liste der beim Hochfahren des Systems zu startenden Dienst.

Paketverwaltung

NGINX

Neben Apache-HTTP der am weitesten verbreitete Webserver unter Linux

Besonderheit bei der Konfiguration unter Ubuntu:

- › Webseitenkonfigurationen werden im Ordner */etc/nginx/sites-available* abgelegt.
- › Um sie zu aktivieren legt man eine symbolische Verknüpfung der jeweiligen Konfigurationen unter */etc/nginx/sites-enabled* ab.

Befehl:

```
# ln -s /etc/nginx/sites-available/<config> /etc/nginx/sites-enabled/<config>
```

Paketverwaltung

Übung

Aufgaben

- › Installiere den Webserver *NGINX*
- › Deaktiviere die Standard-Webseitenkonfiguration
- › Deinstalliere den Webserver und entferne sämtliche Konfigurationen

Bonus

- › Installiere die aktuellste stabile Version von nginx

Vielen Dank

