

# GK641#

Christof Zlabinger 28.03.2023

## Table of Contents

1. [Installation von Webdiens](#)
2. [Dienste verwalten](#)
3. [Arbeitsspeicher und Systemressourcen](#)
4. [Festplattenplatz](#)
5. [Logdaten](#)
6. [Netzwerkverbindungen](#)
7. [Automatisierung und Task-Scheduler](#)
8. [Quellen](#)

## Installation von Webdienst

Updates Installieren:

```
~ : bash — Konsole
File Edit View Bookmarks Plugins Settings Help
New Tab Split View Copy Paste Find
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metadata [102 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [13,5 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [624 kB]
Get:17 http://archive.neon.kde.org/user jammy/main amd64 Packages [2.636 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [580 B]
Get:19 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [811 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metadata [41,5 kB]
Get:21 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [8.388 B]
Get:22 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [575 kB]
Get:23 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11 Metadata [265 kB]
Get:24 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [15,5 kB]
Get:25 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [9.696 B]
Get:26 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [640 kB]
Get:27 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 DEP-11 Metadata [940 B]
Get:28 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 DEP-11 Metadata [13,2 kB]
Get:29 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [11,3 kB]
Get:30 http://archive.neon.kde.org/user jammy/main amd64 DEP-11 Metadata [1.345 kB]
Get:31 http://archive.neon.kde.org/user jammy/main DEP-11 48x48 Icons [623 kB]
Get:32 http://archive.neon.kde.org/user jammy/main DEP-11 64x64 Icons [920 kB]
Get:33 http://archive.neon.kde.org/user jammy/main DEP-11 128x128 Icons [2.171 kB]
Fetched 14,4 MB in 6s (2.475 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
203 packages can be upgraded. Run 'apt list --upgradable' to see them.
W: file:/var/lib/preinstalled-pool/dists/jammy/Release.gpg: Key is stored in legacy trusted.gpg keyring (/etc/
apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
czlabinger@czlabinger-virtualbox:~$
```

Installation vn Apache2:

```

~ : bash — Konsole
File Edit View Bookmarks Plugins Settings Help
New Tab Split View Copy Paste Find
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metadata [102 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [13,5 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [624 kB]
Get:17 http://archive.neon.kde.org/user jammy/main amd64 Packages [2.636 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [580 B]
Get:19 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [811 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metadata [41,5 kB]
Get:21 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [8.388 B]
Get:22 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [575 kB]
Get:23 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11 Metadata [265 kB]
Get:24 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [15,5 kB]
Get:25 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [9.696 B]
Get:26 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [640 kB]
Get:27 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 DEP-11 Metadata [940 B]
Get:28 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 DEP-11 Metadata [13,2 kB]
Get:29 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [11,3 kB]
Get:30 http://archive.neon.kde.org/user jammy/main amd64 DEP-11 Metadata [1.345 kB]
Get:31 http://archive.neon.kde.org/user jammy/main DEP-11 48x48 Icons [623 kB]
Get:32 http://archive.neon.kde.org/user jammy/main DEP-11 64x64 Icons [920 kB]
Get:33 http://archive.neon.kde.org/user jammy/main DEP-11 128x128 Icons [2.171 kB]
Fetched 14,4 MB in 6s (2.475 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
203 packages can be upgraded. Run 'apt list --upgradable' to see them.
W: file:/var/lib/preinstalled-pool/dists/jammy/Release.gpg: Key is stored in legacy trusted.gpg keyring (/etc/
apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
czlabinger@czlabinger-virtualbox:~$

```

Welches Release ist in /etc/apt/sources.list eingetragen?

```

- : sudo nano — Konsole
File Edit View Bookmarks Plugins Settings Help
New Tab Split View Copy Paste Find
GNU nano 6.2 /etc/apt/sources.list
deb http://archive.ubuntu.com/ubuntu/ jammy main restricted universe multiverse
deb http://security.ubuntu.com/ubuntu/ jammy-security main restricted universe multiverse
deb http://archive.ubuntu.com/ubuntu/ jammy-updates main restricted universe multiverse
[ Read 3 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo

```

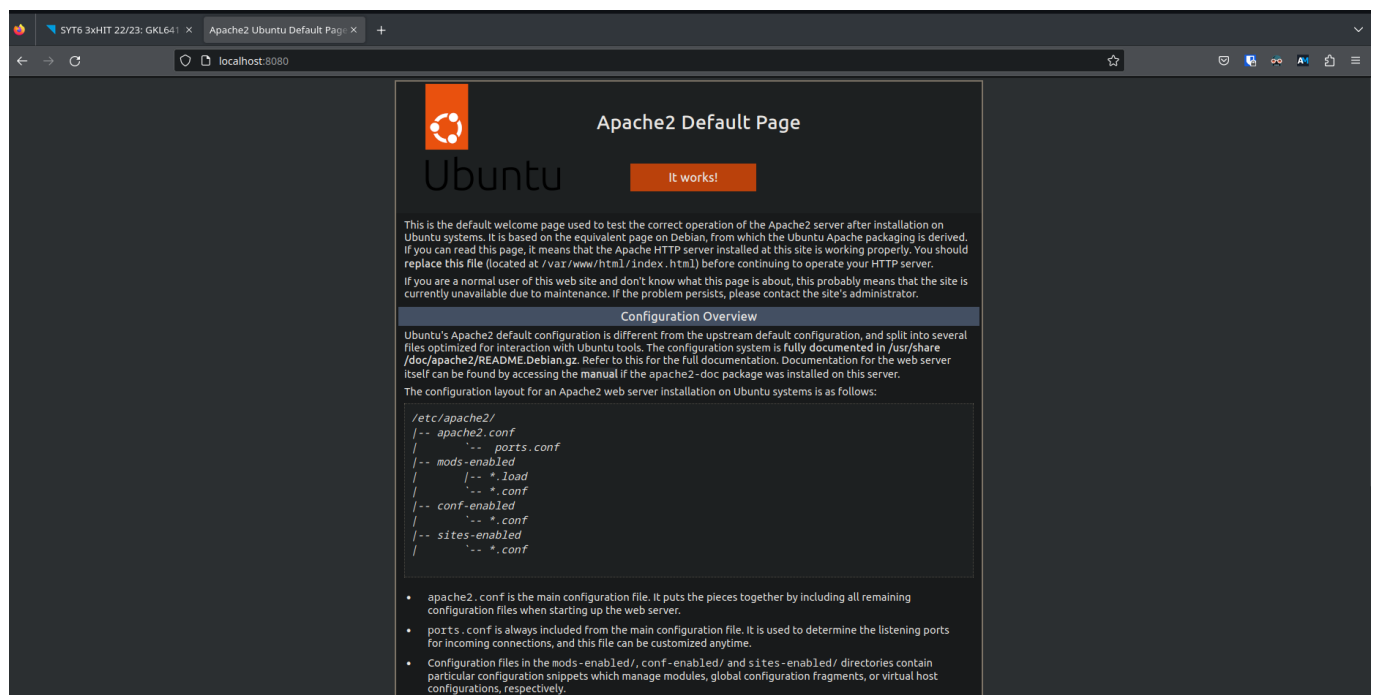
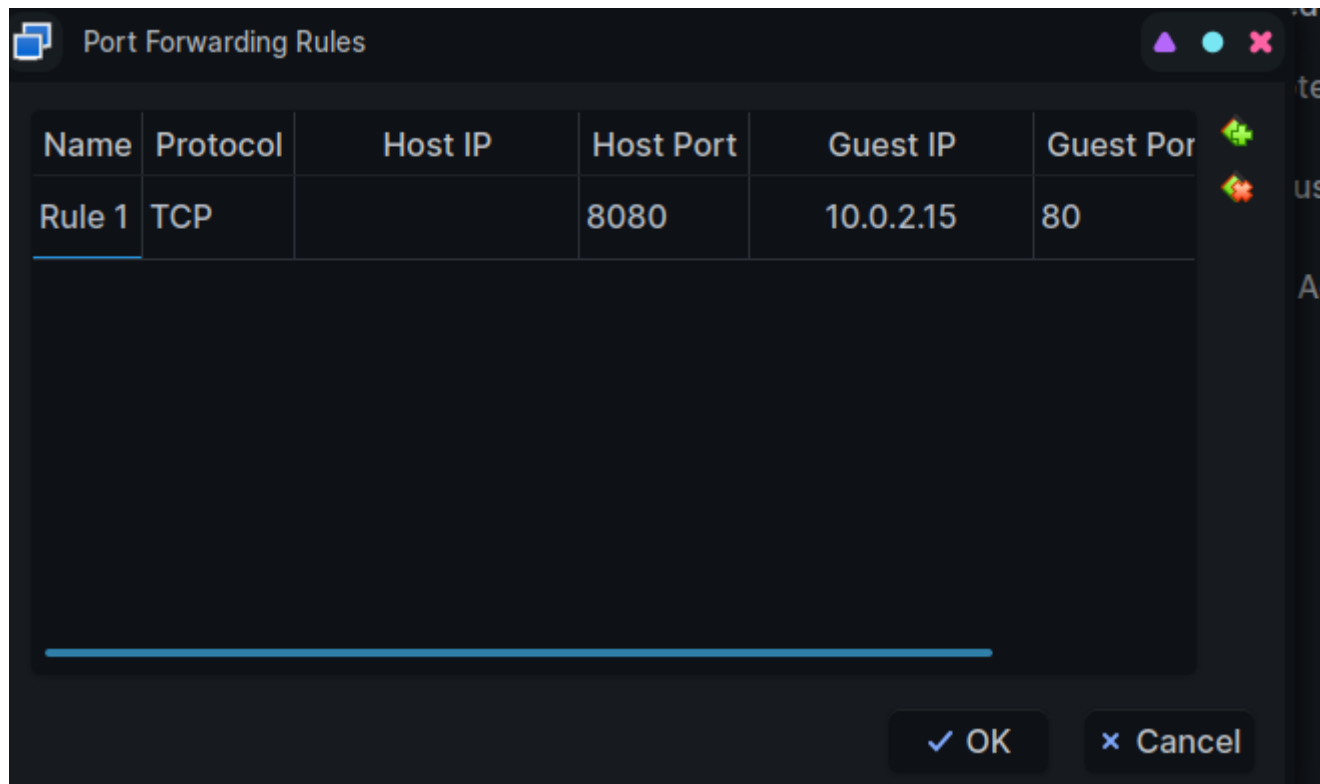
Wie lange wird dieses Release noch mit Updates unterstützt?

5 Jahre und 10 Mit subscription[1]

Was muss nach dem End-of-life-date (EOL) getan werden, um die installierte Distribution weiter verwenden zu können? Gibt es hierfür eine elegantere Methode?

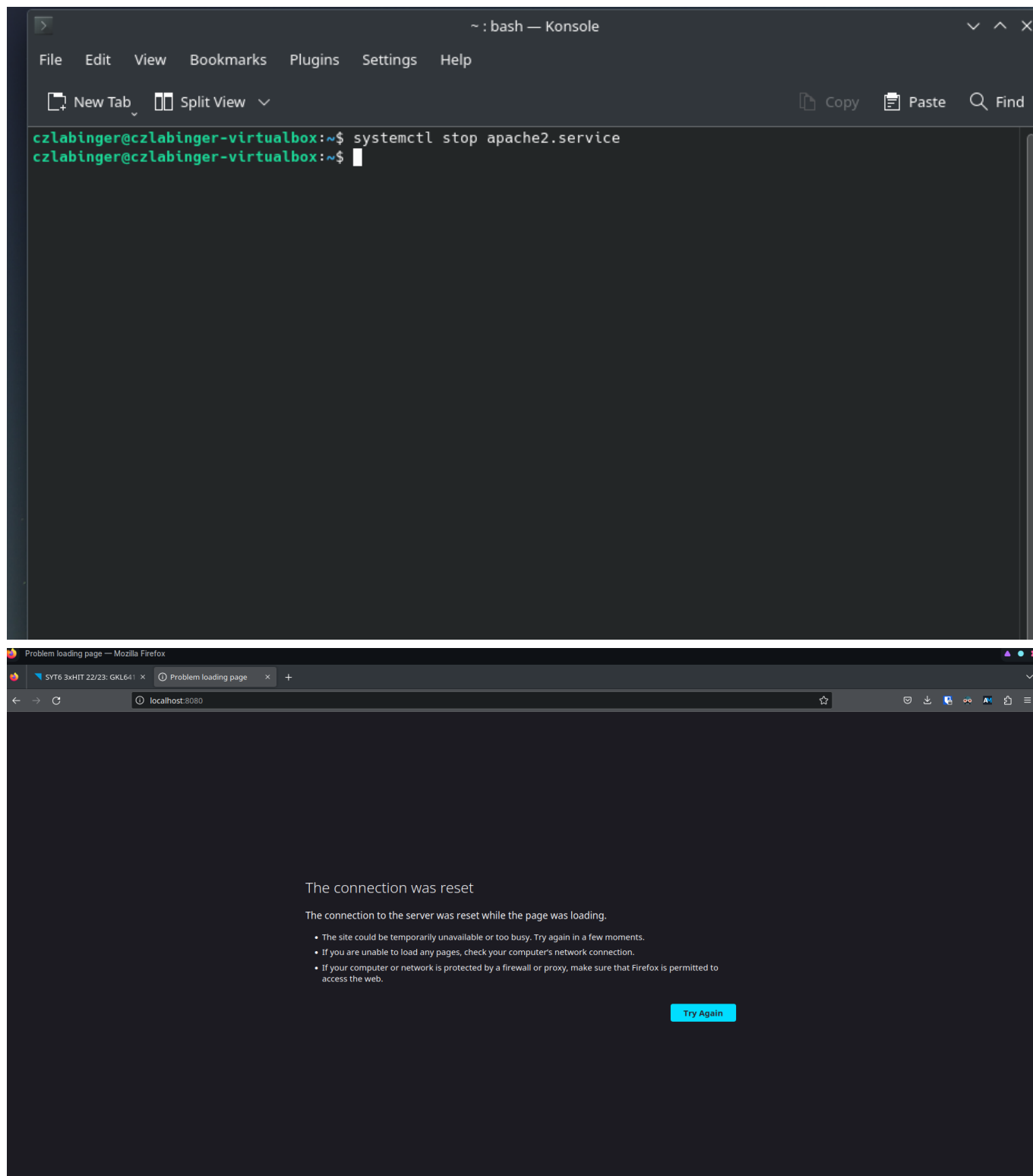
Link händisch austauschen, sudo do-release-upgrade oder neu installieren (Am saubersten)

Konfiguriere deine VM so, dass du von deinem Host mittels `http://localhost:8080` auf diesen Webserver zugreifen kannst



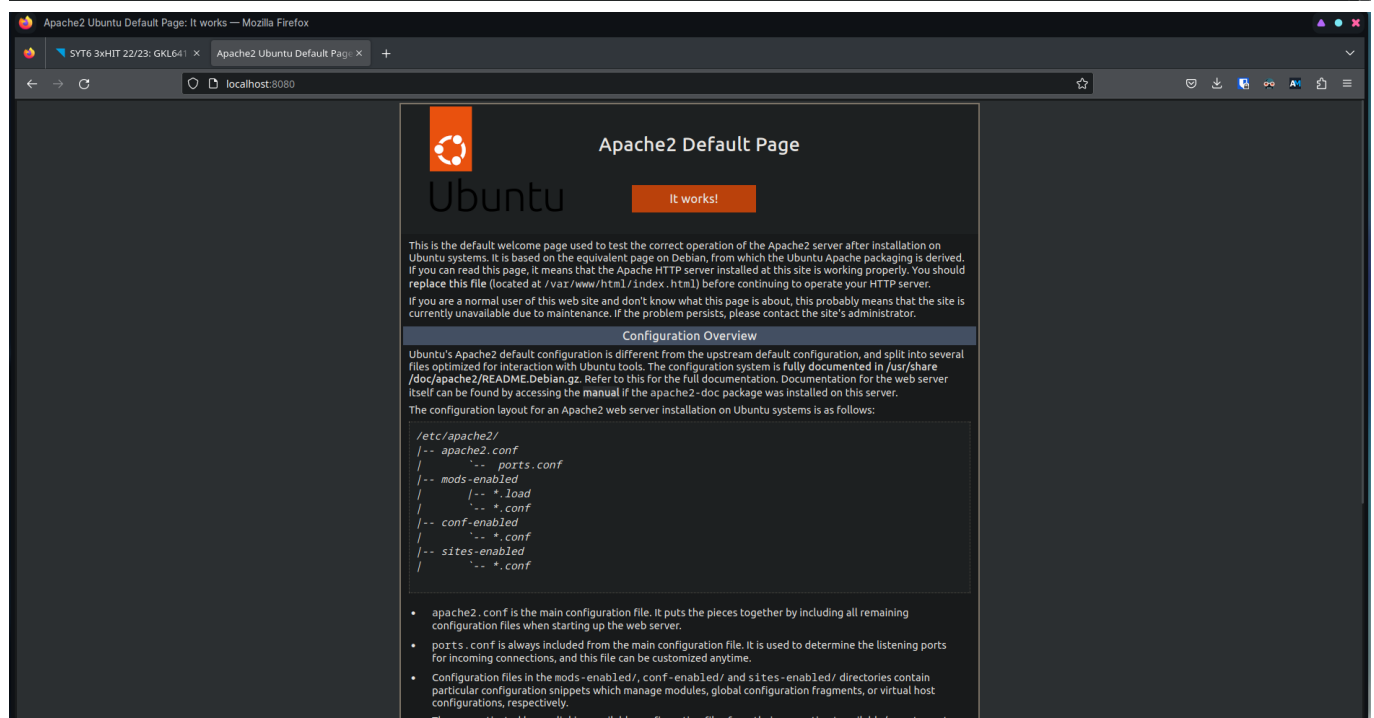
## Dienste verwalten

`systemctl stop apache2`



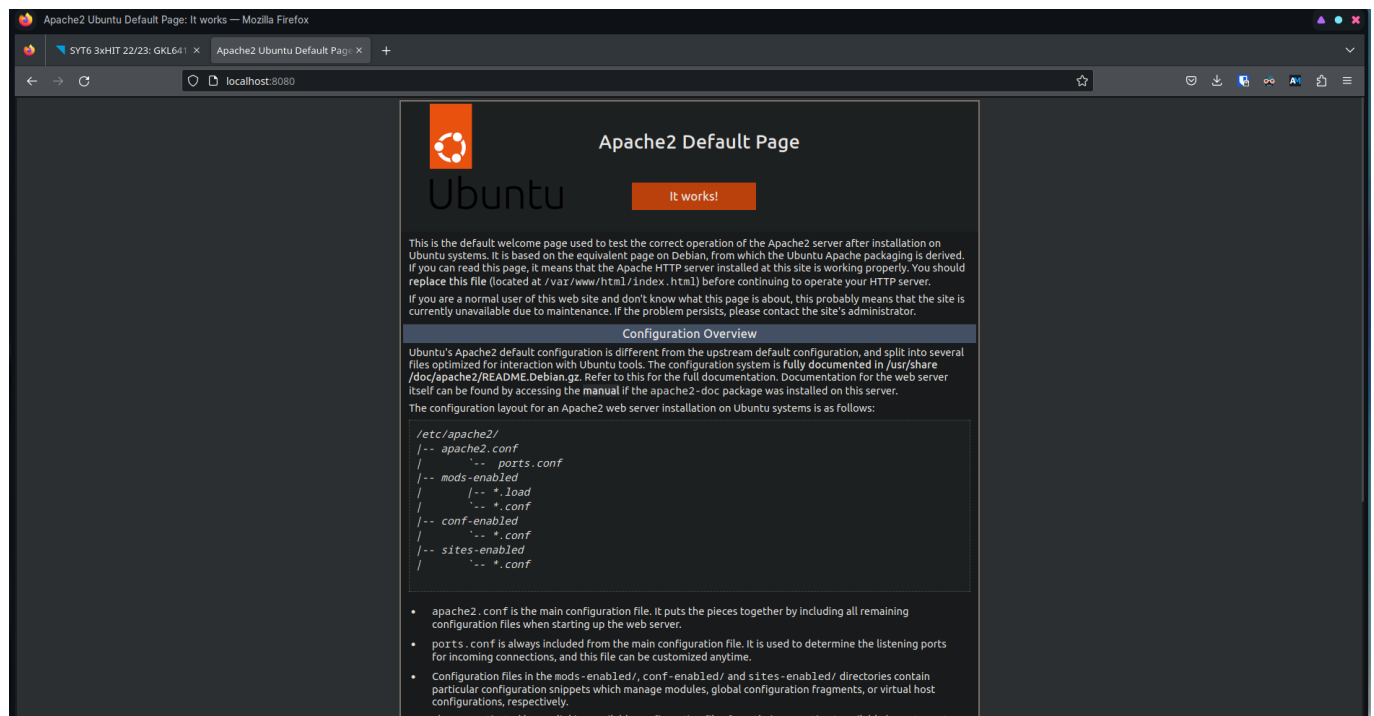
systemctl start apache2

```
czlabinger@czlabinger-virtualbox:~$ systemctl start apache2.service
czlabinger@czlabinger-virtualbox:~$
```



```
systemctl restart apache2
```

```
czlabinger@czlabinger-virtualbox:~$ systemctl restart apache2.service
czlabinger@czlabinger-virtualbox:~$
```



Wie kann der Status eines Dienstes mittels systemctl überprüft werden?

```

czlabinger@czlabinger-virtualbox:~$ systemctl status apache2.service
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-02-16 10:42:14 CET; 36s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 5270 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Main PID: 5274 (apache2)
    Tasks: 55 (limit: 11034)
   Memory: 5.2M
      CPU: 17ms
   CGroup: /system.slice/apache2.service
           └─5274 /usr/sbin/apache2 -k start
             └─5275 /usr/sbin/apache2 -k start
               └─5276 /usr/sbin/apache2 -k start

Feb 16 10:42:14 czlabinger-virtualbox systemd[1]: Starting The Apache HTTP Server...
Feb 16 10:42:14 czlabinger-virtualbox apachectl[5273]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, please add the appropriate entry to your host file.
Feb 16 10:42:14 czlabinger-virtualbox systemd[1]: Started The Apache HTTP Server.
lines 1-17/17 (END)

```

Wenn ein Dienst nicht automatisch gestartet werden soll, ist dieser disabled, welcher Befehl ist notwendig um die Verlinkung für systemd zu aktivieren?

enablen mittels `systemctl enable <Service>`

Was passiert dabei genau?

Erstellt einen Symlink vom Service in einen Folder indem alle services die automatisch gestartet werden sollen hingelinkt werden.

## Arbeitsspeicher und Systemressourcen

Wie viele Prozesse sind das bei dir?

2 (Bash und ps)

Wie viele Prozesse verwendet der Apache Webserver?

```

czlabinger@czlabinger-virtualbox:~$ ps ax | grep apache
2967 ?        Ss      0:00 /usr/bin/htcacheclean -d 120 -p /var/cache/apache2/mod_cache_disk -l 300M -n
5274 ?        Ss      0:00 /usr/sbin/apache2 -k start
5275 ?        Sl      0:00 /usr/sbin/apache2 -k start
5276 ?        Sl      0:00 /usr/sbin/apache2 -k start
5561 pts/1    S+      0:00 grep --color=auto apache

```

Wie kannst du dir zu den Prozessen den Benutzer anzeigen lassen, unter dem sie laufen?

```

czlabinger@czlabinger-virtualbox:~$ ps axu | grep apache
www-data  2967  0.0  0.0  3736  156 ?        Ss   10:29   0:00 /usr/bin/htcacheclean -d 120 -p /var/cache/
apache2/mod_cache_disk -l 300M -n
root      5274  0.0  0.0   6768  4624 ?        Ss   10:42   0:00 /usr/sbin/apache2 -k start
www-data  5275  0.0  0.0 1998248 5444 ?        Sl   10:42   0:00 /usr/sbin/apache2 -k start
www-data  5276  0.0  0.0 1998176 4540 ?        Sl   10:42   0:00 /usr/sbin/apache2 -k start
czlabin+ 5656  0.0  0.0   6956  2548 pts/1    S+   10:57   0:00 grep --color=auto apache

```

Installiere es auf deinem System und rufe htop auf.



```

~: htop — Konsole
File Edit View Bookmarks Plugins Settings Help
New Tab Split View Copy Paste Find

0[ 0.0%] 3[ 0.0%]
1[ 0.0%] 4[ 0.7%]
2[ 0.0%] 5[ 0.0%]
Mem[|||||] 976M/9.09G Tasks: 78, 366 thr; 1 running
Swp[ 0K/10.0G] Load average: 0.29 0.10 0.10
Uptime: 00:33:26

PID USER PRI NI VIRT RES SHR S CPU%MEM% TIME+ Command
840 root 20 0 336M 116M 83336 S 0.7 1.2 0:14.16 /usr/lib/xorg/Xorg -nolisten tcp -auth /var/r
1132 czlabinge 20 0 1661M 173M 135M S 0.7 1.9 0:20.92 /usr/bin/kwin_x11 --replace
1210 czlabinge 20 0 3806M 431M 191M S 0.7 4.6 0:34.12 /usr/bin/plasmashell --no-respawn
5749 czlabinge 20 0 8916 4852 3688 R 0.7 0.1 0:00.03 htop
1 root 20 0 163M 12744 8056 S 0.0 0.1 0:01.85 /sbin/init splash
350 root 19 -1 48092 17312 15956 S 0.0 0.2 0:00.27 /lib/systemd/systemd-journald
390 root 20 0 26452 6928 4720 S 0.0 0.1 0:00.21 /lib/systemd/systemd-udev
404 systemd-r 20 0 25392 13540 9472 S 0.0 0.1 0:00.11 /lib/systemd/systemd-resolved
405 systemd-t 20 0 89380 6628 5768 S 0.0 0.1 0:00.08 /lib/systemd/systemd-timesyncd
455 systemd-t 20 0 89380 6628 5768 S 0.0 0.1 0:00.00 /lib/systemd/systemd-timesyncd
703 root 20 0 8580 4932 1760 S 0.0 0.1 0:00.26 /usr/sbin/haveged --Foreground --verbose=1
714 root 20 0 232M 7728 6900 S 0.0 0.1 0:00.08 /usr/libexec/accounts-daemon
715 root 20 0 2812 1116 1032 S 0.0 0.0 0:00.08 /usr/sbin/acpid
717 avahi 20 0 7628 3600 3264 S 0.0 0.0 0:00.02 avahi-daemon: running [czlabinger-virtualbox.
718 root 20 0 7240 3140 2880 S 0.0 0.0 0:00.00 /usr/sbin/cron -f -P
719 messagebu 20 0 10084 6116 4036 S 0.0 0.1 0:01.50 @dbus-daemon --system --address=systemd: --no
720 root 20 0 252M 18648 15856 S 0.0 0.2 0:00.16 /usr/sbin/NetworkManager --no-daemon
727 root 20 0 82696 3988 3636 S 0.0 0.0 0:00.04 /usr/sbin/irqbalance --foreground
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice + F9Kill F10Quit

```

Welche sind das?

htop zeigt cpu, mem, tasks, threads, uptime, average load, swap

Warum zeigt htop z.B. für den Apache-Webserver mehr Prozesse an als ps?

Da htop auch threads anzeigt

Mittels htop lassen sich auch Prozesse beenden. Starte in einem neuen Reiter die Ausgabe von Zufallsbytes mit dem Befehl `cat /dev/urandom`. Finde diesen neuen, laufenden Prozess in htop und beende diesen mittels den möglichen Funktionen im Prozessbetrachter.

```

0[|||||] 100.0% 3[|||||] 1.3%
1[|||||] 4[|||||]
2[|||||] 20.8% 5[|||||]
Mem[|||||] 2.15G/9.09G Tasks: 79, 374 thr; 6 running
Swp[ 0K/10.0G] Load average: 1.83 0.50 0.26
Uptime: 00:41:20

PID USER PRI NI VIRT RES SHR S CPU%MEM% TIME+ Command
1237 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.60 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1243 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1244 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.06 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1254 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1255 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1256 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1257 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1258 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1259 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1260 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1261 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1262 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1263 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1264 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1265 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1266 czlabinge 39 19 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1285 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1
1287 czlabinge 20 0 1540M 114M 96844 S 0.0 1.2 0:00.00 /usr/lib/x86_64-linux-gnu/libexec/polkit-kde-authentication-agent-1

```



## Festplattenplatz

Recherchiere in der Manpage zu du, was diese beiden Optionen bewirken

-c gib total aus -s gibt total für jedes argument aus

Was sagt dir die Ausgabe?

Wie groß die einzelnen files sind und was ihre totalgröße ist.

```
czlabinger@czlabinger-virtualbox:~$ sudo du -sch /var/log/*
[sudo] password for czlabinger:
28K    /var/log/alternatives.log
12K    /var/log/apache2
180K   /var/log/apt
16K    /var/log/auth.log
8,0K   /var/log/boot.log
120K   /var/log/bootstrap.log
0      /var/log/btmp
8,0K   /var/log/cups
4,0K   /var/log/dist-upgrade
52K    /var/log/dmesg
1,1M   /var/log/dpkg.log
8,0K   /var/log/faillog
4,0K   /var/log/fontconfig.log
4,0K   /var/log/gpu-manager.log
8,0K   /var/log/hp
660K   /var/log/installer
17M    /var/log/journal
68K    /var/log/kern.log
40K    /var/log/lastlog
4,0K   /var/log/private
4,0K   /var/log/samba
0      /var/log/sddm.log
276K   /var/log/syslog
4,0K   /var/log/wtmp
28K    /var/log/Xorg.0.log
19M    total
```

Was bewirkt ein Aufruf von df -h ohne Angabe eines Verzeichnisses?

Es wird als verzeichniss die ranze festplatte verwendet und alle partitions angegeben.

```
czlabinger@czlabinger-virtualbox:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           932M  1,2M  930M   1% /run
/dev/sda1       15G   5,6G   8,4G  41% /
tmpfs           4,6G   0     4,6G   0% /dev/shm
tmpfs           5,0M   4,0K   5,0M   1% /run/lock
tmpfs           932M   60K   932M   1% /run/user/1000
```

## Logdaten

Wie lassen sich nun alle Logdaten zu einem Service mittels systemd anzeigen?

mittels systemctl -l status <Service>

Wie unterscheiden sich diese Informationen zu denen im Verzeichnis /var/log?

in /var/log/ gibt es mehrere logs welche auch ältere ereignisse beinhalten können und systemctl zeigt nur seit dem letzten start

Finde den Filter, der nur Logeinträge für das heutige Datum anzeigt!

```
journalctl --since "2023-02-28 00:00:00" --until "2023-02-28 23:59:59"
```

## Netzwerkverbindungen

Führe dieses Kommando in deiner VM aus, nachdem du die Webseite aus dem ersten Schritt oben vom Host aus aufgerufen hast, und erkläre die Ausgabe.

Es werden alle offenen Sockets welche das TCP protokoll verwenden angezeigt.

```
czlabinger@czlabinger-virtualbox:~$ netstat -talp
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN      -
tcp        0      0 localhost:ipp            0.0.0.0:*               LISTEN      -
tcp        0      0 czlabinger-virtua:51018 kazooie.canonical.:http TIME_WAIT   -
tcp        0      0 czlabinger-virtua:33170 banjo.canonical.co:http TIME_WAIT   -
tcp6       0      0 [::]:1716               [::]:*                  LISTEN      1206/kdeconnectd
tcp6       0      0 localhost:ipp            [::]:*                  LISTEN      -
tcp6       0      0 [::]:http                [::]:*                  LISTEN      -
```

Wie funktioniert dies für das Protokoll UDP?

```
czlabinger@czlabinger-virtualbox:~$ netstat -ualp
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
udp        0      0 0.0.0.0:36476           0.0.0.0:*               -           -
udp        0      0 0.0.0.0:45032           0.0.0.0:*               -           -
udp        0      0 0.0.0.0:mdns             0.0.0.0:*               -           -
udp        0      0 localhost:domain         0.0.0.0:*               -           -
udp        0      0 czlabinger-virtu:bootpc _gateway:bootps         ESTABLISHED -
udp        0      0 0.0.0.0:631              0.0.0.0:*               -           -
udp6       0      0 [::]:mdns                [::]:*                  -           -
udp6       0      0 [::]:55980               [::]:*                  -           -
udp6       0      0 [::]:1716                [::]:*                  -           1206/kdeconnectd
```

## Automatisierung und Task-Scheduler

Konfiguriere das automatische Upgrade von Security-Updates in deiner VM

```

czlabinger@czlabinger-virtualbox:~$ sudo apt install unattended-upgrades apt-listchanges
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Starting pkgProblemResolver with broken count: 0
Starting 2 pkgProblemResolver with broken count: 0
Done
The following packages were automatically installed and are no longer required:
  liblockfile-bin liblockfile1
Use 'sudo apt autoremove' to remove them.
Suggested packages:
  bsd-mailx needrestart
The following NEW packages will be installed:
  apt-listchanges unattended-upgrades
0 upgraded, 2 newly installed, 0 to remove and 337 not upgraded.
Need to get 135 kB of archives.
After this operation, 872 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 apt-listchanges all 3.24 [85,3 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy/main amd64 unattended-upgrades all 2.8ubuntu1 [49,4 kB]
Fetched 135 kB in 1s (116 kB/s)
Preconfiguring packages ...
Selecting previously unselected package apt-listchanges.
(Reading database ... 158421 files and directories currently installed.)
Preparing to unpack .../apt-listchanges_3.24_all.deb ...
Unpacking apt-listchanges (3.24) ...
Selecting previously unselected package unattended-upgrades.
Preparing to unpack .../unattended-upgrades_2.8ubuntu1_all.deb ...
Unpacking unattended-upgrades (2.8ubuntu1) ...
Setting up unattended-upgrades (2.8ubuntu1) ...
Setting up apt-listchanges (3.24) ...
Processing triggers for man-db (2.10.2-1) ...
Not building database; man-db/auto-update is not 'true'.

```

```

czlabinger@czlabinger-virtualbox:~$ sudo dpkg-reconfigure -pnow unattended-upgrades

```

```

czlabinger@czlabinger-virtualbox:~$ sudo nano /etc/apt/apt.conf.d/50unattended-upgrades

```

Wiederkehrende Aufgaben (Tasks) können mit dem Dienst cron oder auch dem systemd.timer durchgeführt werden. Definiere einen Prozess, der dir die Information über deine Festplatten-Auslastung jeden Tag um 06:00 in der Früh in eine Datei speichert, die in das Verzeichnis /var/log/harddisk/ als 20230212-0600.log (nach dem aktuellen Datum) gespeichert wird. Kleiner Tipp, du kannst ein Bash-Script schreiben, welches von den oben genannten Diensten aufgerufen wird und teste dein Script bevor du die Timer einsetzt.

```

GNU nano 6.2 ./auslast
#!/bin/bash

df -h > /home/czlabinger/$(date "+%Y%m%d-%H%M").log

```

```

GNU nano 6.2
[Unit]
Description=Gibt jeden Tag um 06:00 die Festplattenauslastung in einem File aus
[Service]

Type=simple
ExecStart=/home/czlabinger/auslastung.sh
Type=oneshot

```

**[Unit]**

Description=Schreibt jeden Tag um 06:00 die Festplattenauslastung in ein File  
 RefuseManualStart=no  
 RefuseManualStop=no

**[Timer]**

Persistent=true  
 OnCalendar=\*-\*-\* 06:00:00  
 Persistent=true

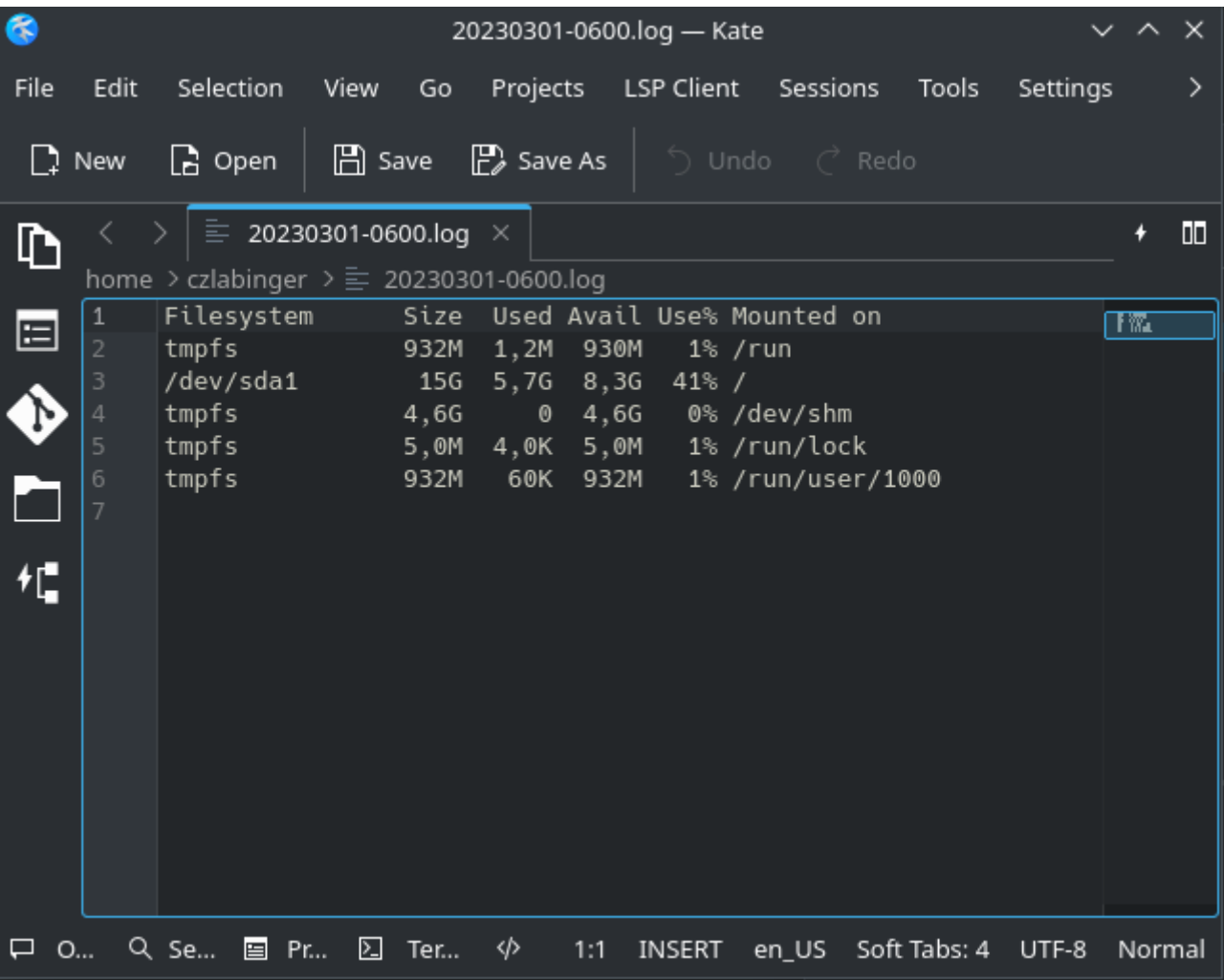
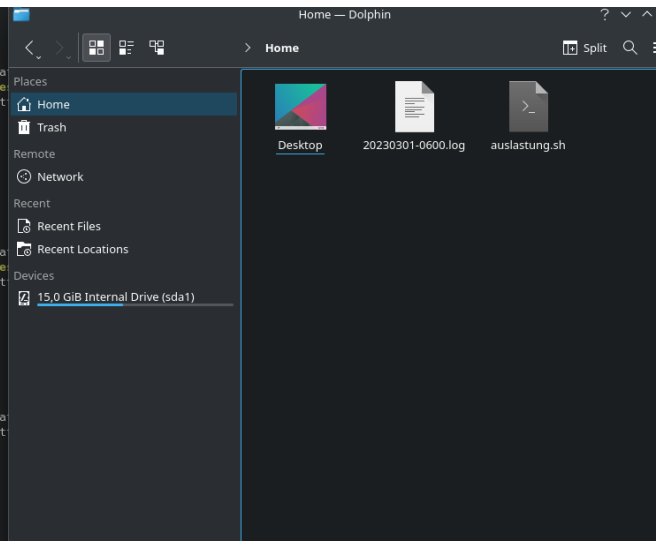
```
czlabinger@czlabinger-virtualbox:~$ systemctl --user start auslastung.timer
```

```
Active: active (waiting) since Tue 2023-02-28 06:02:19 CET; 2min 23s left
Trigger: Wed 2023-03-01 06:00:00 CET; 24h left
Triggers: ● auslastung.service

Feb 28 06:02:19 czlabinger-virtualbox systemd[853]: Stopping Schreibt jeden Tag um 06:00 die Festplatte
Feb 28 06:02:19 czlabinger-virtualbox systemd[853]: auslastung.timer: Not using persistent file time
Feb 28 06:02:19 czlabinger-virtualbox systemd[853]: Started Schreibt jeden Tag um 06:00 die Festplatte
czlabinger@czlabinger-virtualbox:~$ date
Di 28 Feb 2023 06:00:00 CET
czlabinger@czlabinger-virtualbox:~$ systemctl --user status auslastung.timer
● auslastung.timer - Schreibt jeden Tag um 06:00 die Festplattenauslastung in ein File
   Loaded: loaded (/etc/xdg/systemd/user/auslastung.timer; static)
   Active: active (waiting) since Tue 2023-02-28 06:02:19 CET; 2min 11s left
   Trigger: Wed 2023-03-01 06:00:00 CET; 23h left
   Triggers: ● auslastung.service

Feb 28 06:02:19 czlabinger-virtualbox systemd[853]: Stopping Schreibt jeden Tag um 06:00 die Festplatte
Feb 28 06:02:19 czlabinger-virtualbox systemd[853]: auslastung.timer: Not using persistent file time
Feb 28 06:02:19 czlabinger-virtualbox systemd[853]: Started Schreibt jeden Tag um 06:00 die Festplatte
czlabinger@czlabinger-virtualbox:~$ systemctl --user restart auslastung.timer
czlabinger@czlabinger-virtualbox:~$ systemctl --user status auslastung.timer
● auslastung.timer - Schreibt jeden Tag um 06:00 die Festplattenauslastung in ein File
   Loaded: loaded (/etc/xdg/systemd/user/auslastung.timer; static)
   Active: active (waiting) since Wed 2023-03-01 05:59:50 CET; 1s ago
   Trigger: Wed 2023-03-01 06:00:00 CET; 7s left
   Triggers: ● auslastung.service

Mär 01 05:59:50 czlabinger-virtualbox systemd[853]: Stopping Schreibt jeden Tag um 06:00 die Festplatte
Mär 01 05:59:50 czlabinger-virtualbox systemd[853]: Started Schreibt jeden Tag um 06:00 die Festplatte
czlabinger@czlabinger-virtualbox:~$ date
Mi 01 Mär 2023 05:59:50 CET
czlabinger@czlabinger-virtualbox:~$ date
Mi 01 Mär 2023 05:59:50 CET
czlabinger@czlabinger-virtualbox:~$ date
Mi 01 Mär 2023 06:00:00 CET
czlabinger@czlabinger-virtualbox:~$ systemctl --user status auslastung.timer
```



## Quellen

[1] <https://ubuntu.com/about/release-cycle>