

Linuxwissen

immer genau das, was man fix mal braucht

- [Proxmox, Swap optimieren](#)
- [Proxmox, Mount durchreichen](#)
- [Raspi](#)
 - [Raspberry mit WLAN verbinden](#)
 - [Meldungen zur SD-Karte ausgeben](#)
 - [MagicMirror script](#)
- [CLI](#)
 - [User anlegen und zur sudogruppe hinzufügen](#)
 - [Zeitzone im Terminal ändern](#)
 - [Sprache im Terminal ändern](#)
 - [Debian Version ermitteln](#)
 - [fstab prüfen](#)
 - [Debian point-Release upgraden](#)
- [ioBroker](#)
 - [npm updaten](#)
 - [States via http rufen](#)
 - [Node.js updaten](#)
 - [States via http setzen](#)
- [MagicMirror config.js](#)
- [Paperless-ngx updaten](#)
- [Wiki updaten](#)
- [hstr - Befehlshistorytool installieren](#)
- [SSH Zugang & Key](#)
- [Senkrechter Strich](#)

- [Hostname ändern](#)
- [Wifi-Treiber nachinstallieren](#)
- [Einstellung aus fstab ohne Neustart neu laden](#)
- [unrar auf der CLI](#)
- [große Dateien finden](#)
- [PIP und Debian](#)
- [Nginx Proxy Pfad mitgeben](#)
- [NFS Mount](#)
- [PHP updaten](#)
- [Docker auf einem Raspberry Pi installieren](#)
- [Docker Logs Befehl](#)
- [CPU und Sensorwerte](#)
- [Wiki updaten_neu](#)
- [Portainer updaten](#)

Proxmox, Swap optimieren

```
sysctl vm.swappiness=0  
swapoff -a  
swapon -a
```

bzw um dies rückgängig zu machen:

```
sysctl vm.swappiness=1  
swapoff -a  
swapon -a
```

(kann direkt auf der Konsole, ohne reboot, durchgeführt werden, der swapoff dauert ein wenig)

<https://www.technik22.de/d/624-proxmox-swap-optimieren>

Proxmox, Mount durchreichen

```
pct set 308 -mp0 /mnt/hdd1T/,mp=/mnt/
```

Raspi

Raspi

Raspberry mit WLAN verbinden

<https://www.tecchannel.de/a/raspberry-pi-mit-wlan-ausstatten-so-geht-s,2076222,2>

Raspi

Meldungen zur SD-Karte ausgeben

```
dmesg | grep mmc
```

MagicMirror script

```
import requests
import os
import time

url = 'http://10.0.1.122:8087/getPlainValue/'
obj = 'javascript.0.Variablen.Magic_Mirror.motion'

url2 = 'http://10.0.1.122:8087/set/'
obj2 = 'javascript.0.Variablen.Magic_Mirror.reset'

resp = requests.get(url+obj)
#print (resp.text)
if (resp.text) == "true":
    os.system('vcgencmd display_power 1')
elif (resp.text) == 'false':
    os.system('vcgencmd display_power 0')

resp = requests.get(url+obj2)
#print (resp.text)
if (resp.text) == "true":
    #print("true")
    requests.get(url2+obj2+"?value=false")
    os.system('sudo service kiosk.sh restart')
elif (resp.text) == 'false':
    #print("false")
    None
time.sleep(10)

resp = requests.get(url+obj)
if (resp.text) == "true":
    os.system('vcgencmd display_power 1')
elif (resp.text) == 'false':
    os.system('vcgencmd display_power 0')
resp = requests.get(url+obj2)
#print (resp.text)
if (resp.text) == "true":
    #print("true")
    os.system('sudo service kiosk.sh restart')
```

```

elif (resp.text) == 'false':
    #print("false")
    None
time.sleep(10)

resp = requests.get(url+obj)
if (resp.text) == "true":
    os.system('vcgencmd display_power 1')
elif (resp.text) == 'false':
    os.system('vcgencmd display_power 0')
resp = requests.get(url+obj2)
#print (resp.text)
if (resp.text) == "true":
    #print("true")
    os.system('sudo service kiosk.sh restart')
elif (resp.text) == 'false':
    #print("false")
    None
time.sleep(10)

resp = requests.get(url+obj)
if (resp.text) == "true":
    os.system('vcgencmd display_power 1')
elif (resp.text) == 'false':
    os.system('vcgencmd display_power 0')
resp = requests.get(url+obj2)
#print (resp.text)
if (resp.text) == "true":
    #print("true")
    os.system('sudo service kiosk.sh restart')
elif (resp.text) == 'false':
    #print("false")

None
time.sleep(10)

resp = requests.get(url+obj)
if (resp.text) == "true":
    os.system('vcgencmd display_power 1')
elif (resp.text) == 'false':
    os.system('vcgencmd display_power 0')
resp = requests.get(url+obj2)
#print (resp.text)
if (resp.text) == "true":
    #print("true")
    os.system('sudo service kiosk.sh restart')

```

```
elif (resp.text) == 'false':
    #print("false")
    None
time.sleep(10)

resp = requests.get(url+obj)
if (resp.text) == "true":
    os.system('vcgencmd display_power 1')
elif (resp.text) == 'false':
    os.system('vcgencmd display_power 0')
resp = requests.get(url+obj2)
#print (resp.text)
if (resp.text) == "true":
    #print("true")
    os.system('sudo service kiosk.sh restart')
elif (resp.text) == 'false':
    #print("false")
    None
```

CLI

CLI

User anlegen und zur sudogruppe hinzufügen

```
adduser USERNAME
```

```
/usr/sbin/usermod -aG sudo hubobel
```

```
usermod -aG sudo USERNAME
```

CLI

Zeitzone im Terminal ändern

```
sudo dpkg-reconfigure tzdata
```

CLI

Sprache im Terminal ändern

```
sudo dpkg-reconfigure locales
```

CLI

Debian Version ermitteln

```
cat /etc/debian_version
```

CLI

fstab prüfen

die geänderte `/etc/fstab` neu einlesen:

```
systemctl daemon-reload
```

dann auf Fehler prüfen mittels:

```
findmnt --verify
```

CLI

Debian point-Release upgraden

```
apt update
```

```
apt full-upgrade
```

ioBroker

ioBroker

npm updaten

<https://213.136.68.177/topic/22867/how-to-node-js-für-iobroker-richtig-updaten/985?lang=de>

States via http rufen

```
import requests

url = 'http://10.0.1.122:8087/getPlainValue/javascript.0.Variablen.Benutzer'

resp = requests.get(url)
print(resp.text)
```

Node.js updaten

Vorgehensweise

1. Backup erstellen

Vorher immer sichern:

```
iob backup
```

Zusätzlich bei Proxmox oder VM:

- Snapshot erstellen
- oder komplettes Backup

Das ist alte Schule — aber genau damit rettet man sich im Fehlerfall den Abend.

2. System aktualisieren

```
sudo apt update && sudo apt upgrade -y
```

Danach prüfen:

```
node -v  
npm -v
```

3. Node.js 22 installieren

Der empfohlene Weg:

```
iob nodejs-update 22
```

Das Script:

- entfernt alte Node-Versionen sauber
- installiert passende npm-Version
- korrigiert Rechte
- prüft Abhängigkeiten

Genau dafür wurde es inzwischen optimiert.

4. Danach unbedingt prüfen

```
node -v  
npm -v  
iob diag
```

Typisch wäre aktuell etwa:

```
node v22.x  
npm 10.x oder 11.x
```

5. ioBroker neu starten

```
iob restart
```

oder komplett:

```
sudo reboot
```

Ein kompletter Neustart ist oft die sauberere Variante.

ioBroker

States via http setzen

http://10.0.1.122:8087/set/javascript.0.Variablen.Delta_TOPVerbraucher?value=0815

MagicMirror config.js

javascript.0.Variablen.Gewicht_CR

```
{ name: 'Gasverbrauch',
  deviceStates: [
    { id: 'javascript.0.Variablen.Tagesverbrauch_Gas', icon: 'fa-solid fa-fire-flame-simple', suffix: 'm³' },
    { id: 'javascript.0.Variablen.Tagesverbrauch_Gas_Vortag', icon: 'fa-regular fa-calendar-days', suffix: 'm³' }
  ]
},
```

```
{ name: 'Gewicht',
  deviceStates: [
    { id: 'javascript.0.Variablen.Gewicht_CR', icon: 'fa-solid fa-scale-unbalanced-flip', suffix: 'kg' },
    { id: 'javascript.0.Variablen.Gewicht_CR_delta', icon: 'fa-solid fa-plus-minus', suffix: 'kg' }
  ]
},
```

javascript.0.Variablen.Gewicht_CR_delta

<i class="fa-solid fa-plus-minus"></i>

Paperless-ngx updaten

IP: 10.0.1.136

User: hubobel

Login nur noch mit sshkey möglich

```
cd /home/hubobel/paperless/paperless-ng/
```

```
docker-compose down
```

```
docker-compose pull
```

```
docker-compose up -d
```

Wiki updaten

```
cd /var/www/bookstack/
```

```
sudo git pull origin release
```

```
sudo composer install --no-dev
```

```
sudo php artisan migrate
```

```
php artisan cache:clear
```

```
php artisan config:clear
```

```
php artisan view:clear
```

<https://www.bookstackapp.com/docs/admin/updates/>

[PHP aktualisieren](#)

hstr - Befehlshistorytool installieren

`sudo apt install hstr`

für root: `hstr --show-configuration>>/root/.bashrc`

für user: `hstr --show-configuration>>~/ .bashrc`

SSH Zugang & Key

```
sudo nano /etc/ssh/sshd_config
```

- PermitRootLogin no
- PubkeyAuthentication yes
- AuthorizedKeysFile
- PasswordAuthentication no
- UsePAM no

```
sudo systemctl restart sshd
```

Senkrechter Strich

Tastenkombination Alt Gr + <

Hostname ändern

First, run `sudo hostnamectl | grep hostname` and confirm that's the name of the PC you want to change.

Next, run `sudo hostnamectl set-hostname whatever` where *whatever* is the name you wish to use henceforth.

Wifi-Treiber nachinstallieren

```
sudo apt install bcmwl-kernel-source
```

Einstellung aus fstab ohne Neustart neu laden

Die einfachste Möglichkeit ist den **mount** Befehl wie folgt auszuführen.

```
$ sudo mount -a
```

Ein Nachteil davon ist jedoch, dass dabei bei den vorliegenden Systemen nur die noch nicht eingehängten Dateisysteme eingehängt wurden. Eine weitere und zuverlässigere Lösung stellt die Erweiterung der Parameter für den **mount** Befehl dar. Dies könnte dann wie folgt aussehen.

```
$ sudo mount -o remount /
```

In diesem Beispiel wird das Root-Dateisystem mit den in der fstab angegebenen Optionen neu eingehängt. Eine weitere Erleichterung brachte folgende Kombination von Befehlen. Dabei werden zunächst alle eingehängten Dateisysteme aufgelistet und anschließend neu eingehängt. Es bietet sich an zunächst den letzten Teil wegzulassen und sich die auszuführenden Befehle genauer anzuschauen.

```
$ LANG=en mount | awk '{print "mount -o remount \"$3\"'}' | sudo sh
```

unrar auf der CLI

```
unrar e tecmint.rar /home/tecmint/rarfiles
```

große Dateien finden

`sudo apt-get install ncdu`

<https://www.mawa-solutions.com/faq/artikel/wie-findet-man-unter-linux-gro%C3%9Fe-verzeichnisse>

PIP und Debian

Use Pip's `config` command to edit the above file:

```
python3 -m pip config set global.break-system-packages true
```

Nginx Proxy Pfad mitgeben

```
location = /{return 301 $scheme://$http_host/weather/;}
```

/weather/ =

Edit Proxy Host



[⚡ Details](#) [≡ Custom locations](#) [🛡 SSL](#) [⚙ Advanced](#)

These proxy details are available as nginx variables:

- `$server` Forward Hostname / IP
- `$port` Forward Port
- `$forward_scheme` Scheme

Custom Nginx Configuration

```
location = /{return 301 $scheme://$http_host/ubooquity/;}
```

⚠ Please note, that any `add_header` or `set_header` directives added here will not be used by nginx. You will have to add a custom location `'/'` and add the header in the custom config there.

Cancel

Save

NFS Mount

```
mount -t nfs -o soft 10.0.1.253:/volume3/Downloadstation /mnt/nas/Downloadstation/
```

um das mounten beim Systemstart/Boot automatisiert durchzuführen:

```
nano /etc/fstab
```

```
10.0.1.253:/volume3/Downloadstation /mnt/nas/Downloadstation nfs soft,rw,noexec,nolock,_netdev  
0 0
```

prüfen mittels:

```
mount -a
```

```
mount | grep Downloadstation
```

sollte ergeben:

```
10.0.1.253:/volume3/Downloadstation on /mnt/nas/Downloadstation type nfs  
(rw,soft,nolock,_netdev)
```

und im Pfad /mnt/nas/Downloadstation sollt verfügbar sein.

PHP updaten

Für Linux (Ubuntu/Debian) mit Apache

1. Alte PHP-Version anzeigen

```
php -v
```

2. Repositories hinzufügen (z. B. Ondřej Surý PPA)

```
sudo apt update  
sudo apt install software-properties-common  
sudo add-apt-repository ppa:ondrej/php  
sudo apt update
```

3. Neue PHP-Version installieren (z. B. PHP 8.3)

```
sudo apt install php8.3
```

Optional auch die benötigten Module:

```
sudo apt install php8.3-cli php8.3-common php8.3-mbstring php8.3-xml php8.3-curl php8.3-mysql
```

4. Apache oder Nginx umstellen

```
sudo a2dismod php7.4 # alte Version deaktivieren  
sudo a2enmod php8.3 # neue aktivieren  
sudo systemctl restart apache2
```

Docker auf einem Raspberry Pi installieren

per SSH Verbindung aufnehmen (Putty)

Docker installieren in Shell

```
curl -fsSL https://get.docker.com -o get-Docker.sh
```

```
sudo sh get-Docker.sh
```

```
sudo usermod -aG docker $USER
```

```
newgrp docker
```

```
docker run hello-world
```

Quelle: <https://www.heise.de/news/Wie-man-Docker-auf-dem-Raspberry-Pi-in-15-Minuten-einrichtet-7524692.html>

Docker Logs Befehl

Alle laufenden Container anzeigen, um die Container-ID zu erhalten

```
$ docker container ls
```

Logs eines Containers anzeigen

```
$ docker logs <container_id>
```

Logs in Echtzeit streamen

```
$ docker logs -f <container_id>
```

Die letzten 100 Zeilen der Logs anzeigen

```
$ docker logs --tail 100 <container_id>
```

Logs ab einem bestimmten Zeitpunkt anzeigen

```
$ docker logs --since="2025-05-20T10:00:00" <container_id>
```

Logs bis zu einem bestimmten Zeitpunkt anzeigen

```
$ docker logs --until="2025-05-20T12:00:00" <container_id>
```

Logs mit Zeitstempeln anzeigen

```
$ docker logs --timestamps <container_id>
```

CPU und Sensorwerte

Temperatur und Frequenz zusammen

```
watch -n2 '  
echo "=== CPU ==="  
grep "cpu MHz" /proc/cpuinfo | head  
echo  
sensors  
'
```

Aktuelle Frequenz aller Kerne live anzeigen

```
watch -n1 "grep 'cpu MHz' /proc/cpuinfo"
```

Aktualisiert jede Sekunde.

Kompakter Überblick

```
watch -n1 "grep 'cpu MHz' /proc/cpuinfo | awk '{print \$4}'"
```

Zeigt nur die Frequenzen an.

Min/Max und aktuelle Policy

```
cpupower frequency-info
```

und danach einige Stunden beobachten:

watch -n5 sensors

Oft sinken die Lastspitzen dann von 70–80 °C auf 50–60 °C, wodurch der Lüfter deutlich seltener hochdreht.

Wiki updaten_neu

```
cd /var/www/bookstack
php artisan down
git pull --ff-only origin release
composer install --no-dev
php artisan migrate --force
sudo -u www-data php artisan cache:clear
php artisan up
sudo systemctl restart apache2
```

Portainer updaten

Portainer CE auf Docker aktualisieren

Diese Anleitung aktualisiert eine Portainer-CE-Installation von **2.19.4** auf **2.39.2 LTS**.

Aktuelle Konfiguration

- Containername: `portainer`
- Image bisher: `portainer/portainer-ce:latest`
- Ziel-Image: `portainer/portainer-ce:lts`
- Portainer HTTPS: `9443`
- Edge-Agent-Tunnel: `8000`
- Datenvolume: `portainer_data`
- Docker-Socket: `/var/run/docker.sock`

Die vorhandenen Einstellungen, Benutzer, Stacks und Environments befinden sich im Volume `portainer_data`.

1. Aktuelle Version prüfen

```
docker exec portainer /portainer --version
```

2. Neues LTS-Image herunterladen

Portainer läuft während des Downloads weiter:

```
docker pull portainer/portainer-ce:lts
```

3. Portainer stoppen

```
docker stop portainer
```

Andere Container und Anwendungen bleiben davon unberührt.

4. Portainer-Daten sichern

```
sudo tar \
  -C /var/lib/docker/volumes/portainer_data/_data \
  -czf /home/hubobel/portainer-data-pre-2.39.2.tar.gz \
  .
```

Backup kontrollieren

```
sudo ls -lh /home/hubobel/portainer-data-pre-2.39.2.tar.gz
```

Inhalt stichprobenartig anzeigen:

```
sudo tar -tzf /home/hubobel/portainer-data-pre-2.39.2.tar.gz | head
```

Die Ausgabe sollte unter anderem `portainer.db` enthalten.

Backup gegen unberechtigten Zugriff schützen:

```
sudo chmod 600 /home/hubobel/portainer-data-pre-2.39.2.tar.gz
```

Der alte Container sollte erst entfernt werden, wenn das Backup vorhanden und größer als `0` Byte ist.

5. Alten Portainer-Container entfernen

```
docker rm portainer
```

Dabei wird nur der Container entfernt. Das Volume `portainer_data` bleibt erhalten.

6. Neue LTS-Version starten

```
docker run -d \  
  --name portainer \  
  --restart=always \  
  -p 8000:8000 \  
  -p 9443:9443 \  
  -v /var/run/docker.sock:/var/run/docker.sock \  
  -v portainer_data:/data \  
  portainer/portainer-ce:lts
```

Beim ersten Start aktualisiert Portainer seine interne Datenbank. Der Container darf währenddessen nicht neu gestartet oder gestoppt werden.

7. Startprotokoll kontrollieren

```
docker logs portainer --tail 100
```

Bei Bedarf kann das Protokoll fortlaufend beobachtet werden:

```
docker logs -f portainer
```

Die Anzeige wird mit `Strg+C` beendet, ohne den Container zu stoppen.

8. Neue Version prüfen

```
docker exec portainer /portainer --version
```

Erwartete Ausgabe:

```
2.39.2
```

9. Funktion prüfen

Anschließend in der Portainer-Weboberfläche kontrollieren:

- Anmeldung funktioniert
- Docker-Environment ist erreichbar
- vorhandene Stacks werden angezeigt
- Container und Volumes sind vorhanden
- Container-Konsole und Protokollanzeige funktionieren

Backup aufbewahren

Das Backup befindet sich unter:

```
/home/hubobel/portainer-data-pre-2.39.2.tar.gz
```

Es sollte mindestens so lange aufbewahrt werden, bis die aktualisierte Portainer-Installation vollständig geprüft wurde.

Wichtige Hinweise

- Das Volume `portainer_data` nicht löschen.
- Kein neues Volume mit einem anderen Namen verwenden.
- Nicht von `portainer/portainer-ce` zu `portainer/portainer-ee` wechseln.
- Ein Downgrade nach erfolgter Datenbankmigration sollte nicht mit dem bestehenden Volume versucht werden.
- Für eine Wiederherstellung muss das vor dem Update erzeugte Backup verwendet werden.