

# Einrichtung von Jupyter Notebooks mit Anaconda, CUDA und Fernzugriff

Diese Anleitung soll die Einrichtung eines Fernzugriffs auf Jupyter Notebooks, die auf einem PC mit Nvidia-Grafikkarte in einem Heimnetzwerk laufen, erläutern.

Als Betriebssystem wird hier [Rocky Linux 8](#) (binärkompatibel zu RHEL 8) genutzt. Mit leichten Anpassungen können auch andere Linux-Distributionen wie [OpenSUSE](#) oder [Debian](#) verwendet werden. Hierbei sollte auf die von der zu installierenden CUDA-Version unterstützten Distributions-Versionen geachtet werden.

Bei der Auswahl der zu installierenden CUDA- und Anaconda-Version sollte außerdem auf die Kompatibilität der mit Anaconda ausgelieferten PyTorch-Version mit der CUDA-Version geachtet werden. In dieser Anleitung werden Anaconda 2022.10 und CUDA 11.7 verwendet.

## Nutzer zur Wheel-Gruppe

```
usermod -aG wheel 'marc'
```

## CUDA

### Vorbereitung

```
dnf install -y epel-release gcc kernel-devel-$(uname -r) kernel-headers-$(uname -r)
```

Eine ausführliche Anleitung ist auf [Nvidias Dokumentations-Website](#) zu finden.

### Installation

Auf [Nvidias Developer-Website](#) kann man sich über einen Klick auf den Button rpm (local) eine Anleitung zum Herunterladen und Ausführen des Installationskriptes anzeigen lassen. Die Befehle müssen nun einzeln im Terminal ausgeführt werden und sollten ähnlich den folgenden aussehen.

```
wget https://developer.download.nvidia.com/compute/cuda/11.7.0/local_installers/cuda-repo-rhel8-11-7-local-11.7.0_515.43.04-1.x86_64.rpm
sudo rpm -i cuda-repo-rhel8-11-7-local-11.7.0_515.43.04-1.x86_64.rpm
sudo dnf clean all
sudo dnf -y module install nvidia-driver:latest-dkms
sudo dnf -y install cuda
```

# Anaconda

## Installation

Auf der [Anaconda-Website](#) zunächst über den Link „64-Bit (x86) Installer“ das Installationskript herunterladen.

Im Download-Verzeichnis anschließend folgende Befehle ausführen:

```
chmod +x Anaconda3-2022.10-Linux-x86_64.sh
./Anaconda3-2022.10-Linux-x86_64.sh
```

## PyTorch

Nach dem Ausführen des Anaconda-Installationskriptes muss noch PyTorch mit CUDA-Unterstützung installiert werden. Der passende Befehl kann auf der [PyTorch-Website](#) je nach Installationsart und CUDA-Version zusammengedrückt werden (in diesem Fall Stable / Linux / Conda / Python / CUDA 11.7).

```
conda install pytorch torchvision torchaudio pytorch-cuda=11.7 -c pytorch -c nvidia
conda install tensorboard torchvision
```

## Fernzugriff

### Installation benötigter Pakete

Zwischen den S

```
dnf install nginx certbot python3-certbot-nginx
systemctl start nginx
systemctl enable nginx
```

### DynDNS einrichten

### Systemd-Service für Notebooks anlegen

Um nicht nach jedem Start des Systems Jupyter Notebook händisch starten zu müssen, wird ein Systemd-Service eingerichtet.

```
/etc/systemd/system/jupyter-notebook.service
```

```
[Unit]
```

```
Description=JupyterNotebook
After=syslog.target
After=network.target
```

```
[Service]
Type=simple
User=[username]
Group=[groupname]
WorkingDirectory=[directory]
ExecStart=[executable]
Restart=always
```

```
[Install]
WantedBy=multi-user.target
```

```
systemctl start jupyter-notebook
systemctl enable jupyter-notebook
```

## Firewall

```
firewall-cmd --permanent --add-service=http
firewall-cmd --permanent --add-service=https
```

## Proxy-Konfiguration

/etc/nginx/conf.d/jupyter.conf

```
server {
    server_name [domain];

    listen 80;
    listen [::]:80;
    listen 443 ssl http2;
    listen [::]:443 ssl http2;

    if ($ssl_protocol = "") {
        return 301 https://$server_name$request_uri;
    }

    root /usr/share/nginx/html;

    #add_header Content-Security-Policy "default-src 'none'; img-src 'self'
data$
    add_header Strict-Transport-Security "max-age=31536000; preload;";
    add_header X-Content-Type-Options nosniff;
    add_header X-XSS-Protection "1; mode=block";
    add_header Referrer-Policy "no-referrer";
```

```
ssl_certificate      /etc/letsencrypt/live/[domain]/fullchain.pem;  
ssl_certificate_key  /etc/letsencrypt/live/[domain]/privkey.pem;  
  
#include /etc/nginx/tls-config;  
  
location / {  
    proxy_pass http://localhost:8888;  
    proxy_set_header Host $http_host;  
    proxy_set_header X-Real-IP $remote_addr;  
}  
  
access_log /var/log/nginx/access-jupyter.log;  
error_log /var/log/nginx/error-jupyter.log;  
}
```

**TODO:**

- DynDNS-Einrichtung
- tls-config ersetzen
- CSP
- Domains verallgemeinern

From:  
<https://wiki.schlagma.de/> - **schlagmaWiki**

Permanent link:  
<https://wiki.schlagma.de/oeffentlich/dl4cv-einrichtung>

Last update: **05.05.2023 12:08**

