

# Node Exporter (Datensammler)

- [Node Exporter auf einem Debian Server installieren](#)
- [SNMP Exporter auf einem Debian-System installieren](#)

# Node Exporter auf einem Debian Server installieren

## Einleitung

In diesem Beitrag geht es darum, wie wir auf unserem *Debian Server* einen *Node Exporter* installieren können, um von diesem Server aus Metriken sammeln zu können. Diese können wir dann auswerten oder z.B. in *Grafana Dashboards* anzeigen lassen.

## Node Exporter installieren

Um den *Node Exporter* auf unserem System zu installieren, müssen wir eine laufende *Prometheus Instanz* laufen haben. Auf unserem zu überwachenden Server erstellen wir im ersten Schritt ein Verzeichnis, in dem sich die Dateien für unseren *Node Exporter* befinden.

```
mkdir /opt/node_exporter
```

Dann laden wir uns die aktuellste Version des *Node Exporters* für *Linux* herunter und entpacken diese.

```
cd /opt/node_exporter/  
wget https://github.com/prometheus/node_exporter/releases/download/v1.7.0/node_exporter-1.7.0.linux-amd64.tar.gz  
tar xvfz node_exporter-1.7.0.linux-amd64.tar.gz
```

**Linux Node Exporter:** [https://github.com/prometheus/node\\_exporter/releases/](https://github.com/prometheus/node_exporter/releases/)

Um die Übersichtlichkeit zu erhöhen, verschieben wir die Dateien eine Ebene höher und löschen den temporären Ordner und das heruntergeladene Archiv.

```
mv node_exporter-1.7.0.linux-amd64/* .  
rm -r node_exporter-1.7.0.linux-amd64  
rm node_exporter-1.7.0.linux-amd64.tar.gz
```

Jetzt müssen wir den *Node Exporter* als *Systemdienst* einrichten. Dazu erstellen wir eine Datei und fügen den nachstehenden Inhalt ein:

```
nano /etc/systemd/system/node_exporter.service
```

```
[Unit]
```

```
Description=Node Exporter
```

```
[Service]
```

```
User=root
```

```
Group=root
```

```
ExecStart=/opt/node_exporter/node_exporter
```

```
[Install]
```

```
WantedBy=multi-user.target
```

Im letzten Schritt starten wir den *Node Exporter*, damit dieser von unserem Prometheus Server abgerufen werden kann.

```
systemctl daemon-reload
```

```
systemctl start node_exporter
```

```
systemctl enable node_exporter
```

Jetzt passen wir noch die **prometheus.yml** an, und dann sollten die Metriken auf unserem *Prometheus Server* landen.

# SNMP Exporter auf einem Debian-System installieren

## Einleitung

In diesem kurzen Artikel geht es kurz darum, wie wir auf unserem Debian System den **Prometheus SNMP Exporter** einrichten können. Diesen benötigen wir, um z.B. **SNMP** fähige Netzwerkgeräte zu überwachen.

## Installation

Um jetzt unseren Server mit dem **Prometheus SNMP Exporter** auszustatten, müssen wir die folgenden Schritte befolgen. Im ersten Schritt müssen wir den SNMP Exporter herunterladen. Dies machen wir mit dem folgenden Befehl:

```
wget https://github.com/prometheus/snmp_exporter/releases/download/v0.26.0/snmp_exporter-0.26.0.linux-amd64.tar.gz
```

Im nächsten Schritt entpacken wir das Verzeichnis.

```
tar xzf snmp_exporter-0.26.0.linux-amd64.tar.gz
```

Wir können die Dateien dann in das Verzeichnis unserer Wahl kopieren. Ich lege ein Verzeichnis unter `/opt/snmp-exporter` an.

```
mkdir /opt/snmp-exporter  
mv ./snmp_exporter-0.19.0.linux-amd64/* /opt/snmp-exporter/
```

Jetzt legen wir einen **prometheus** Benutzer an, der den Dienst schlussendlich im Hintergrund ausführt.

```
sudo useradd --system prometheus
```

Jetzt legen wir eine Datei, an, die die Dienstinformationen enthält. Damit erstellen wir sozusagen einen eigenen Dienst, der im Hintergrund ausgeführt wird.

```
sudo nano /etc/systemd/system/snmp-exporter.service
```

Mit dem folgenden Inhalt:

```
[Unit]
Description=Prometheus SNMP Exporter Service
After=network.target

[Service]
Type=simple
User=prometheus
ExecStart=/opt/snmp-exporter/snmp_exporter --config.file="/opt/snmp-exporter/snmp.yml"

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

Jetzt starten wir den Dienst und dann können wir die Webseite des **SNMP-Exporters** erreichen.

```
systemctl daemon-reload
sudo service snmp-exporter start
sudo service snmp-exporter status
```