

NUT-Master für APC Smart-UPS SMC1500IC unter Ubuntu einrichten

Ziel

Die APC Smart-UPS SMC1500IC wird über ein USB-Datenkabel mit einem Ubuntu-Server verbunden.

Dieser Server übernimmt folgende Aufgaben:

- * Auslesen des USV-Status
- * Erkennen eines Stromausfalls
- * Kontrolliertes Herunterfahren des lokalen Servers
- * Bereitstellen der USV-Informationen für weitere Server im Netzwerk

Der direkt mit der USV verbundene Rechner wird in dieser Anleitung als **NUT-Master** bezeichnet.

<note important>

Das USB-Datenkabel der USV muss direkt mit dem Ubuntu-Server verbunden sein.

Der SmartConnect-Netzwerkanschluss ersetzt die USB-Verbindung für NUT nicht.

</note>

Voraussetzungen

- * APC Smart-UPS SMC1500IC
- * Ubuntu 22.04 oder Ubuntu 24.04
- * USB-Datenkabel zwischen USV und Server
- * Feste IP-Adresse für den NUT-Master
- * Root- oder sudo-Berechtigung

Beispiel-IP des NUT-Masters:

```
\\  
192.168.1.20\\
```

Diese Adresse muss durch die tatsächliche IP-Adresse des Servers ersetzt werden.

1. USB-Verbindung prüfen

Die USV mit dem USB-Datenkabel an den Ubuntu-Server anschließen.

Anschließend prüfen, ob die USV erkannt wird:

```
lsusb | grep -i -E 'APC|American Power Conversion'
```

Eine mögliche Ausgabe lautet:

```
Bus 001 Device 004: ID 051d:0003 American Power Conversion \UPS
```

Die Bus- und Gerätenummern können abweichen.

Die Kennung 051d steht für APC beziehungsweise American Power Conversion.

2. NUT installieren

Paketquellen aktualisieren:

```
sudo apt \update
```

NUT-Server und NUT-Client installieren:

```
sudo apt install nut-server nut-\client
```

Installierte Version anzeigen:

```
apt-cache policy nut-\server
```

3. Vorhandene Konfiguration sichern

Vor Änderungen sollte das komplette NUT-Konfigurationsverzeichnis gesichert werden:

```
sudo cp -a /etc/nut "/etc/nut.backup.$(date +%F-%H%M%S)"
```

Die Sicherung befindet sich anschließend beispielsweise unter:

```
/etc/nut.backup.2026-07-19-110000
```

4. NUT-Betriebsart einstellen

Die Datei /etc/nut/nut.conf öffnen:

```
sudo nano /etc/nut/nut.\conf
```

Folgende Einstellung eintragen:

```
MODE=\netserver
```

<note> Bei MODE=netserver werden der USV-Treiber, der NUT-Datenserver und der lokale Überwachungsdienst gestartet.

Zusätzlich können andere Server im Netzwerk auf die USV-Informationen zugreifen. </note>

Beim Eintrag dürfen um das Gleichheitszeichen keine Leerzeichen stehen.

Richtig:

```
MODE=\netserver
```

Falsch:

```
MODE = \netserver
```

5. APC-USV konfigurieren

Die Datei /etc/nut/ups.conf öffnen:

```
sudo nano /etc/nut/ups.conf
```

Folgenden Abschnitt eintragen:

```
[apc]
driver = usbhid-\ups
port = \auto
vendorid = \051d
desc = "APC Smart-UPS SMC1500IC"
```

Bedeutung der Einstellungen:

Einstellung	Bedeutung
[apc]	Interner Name der USV
driver	USB-HID-Treiber für kompatible APC-USV-Geräte
port = auto	Automatische Erkennung des USB-Anschlusses
vendorid = 051d	USB-Herstellererkennung von APC
desc	Frei wählbare Beschreibung

Wenn mehrere APC-USV-Geräte am selben Server angeschlossen sind, kann zusätzlich die Produkt-ID angegeben werden:

```
\\
productid = 0003\\
```

Eine vollständige Konfiguration kann dann so aussehen:

```
[apc]
driver = usbhid-\ups
port = \auto
vendorid = \051d
productid = \0003
```

```
desc = "APC Smart-UPS SMC1500IC"
```

6. IP-Adresse des Servers feststellen

Die vorhandenen Netzwerkadressen anzeigen:

```
ip -br \address
```

Beispiel:

```
enp3s0    UP    192.168.1.20/\24
docker0   UP    172.17.0.1/\16
```

Für NUT muss die normale LAN-Adresse verwendet werden:

```
192.168.1.\20
```

Nicht verwendet werden sollte eine interne Docker-Adresse wie:

```
172.17.0.\1
```

7. NUT-Netzwerkdienst konfigurieren

Die Datei /etc/nut/upsd.conf öffnen:

```
sudo nano /etc/nut/upsd.conf
```

Folgende Zeilen eintragen:

```
LISTEN 127.0.0.1 \3493
LISTEN 192.168.1.20 \3493
```

Dabei muss 192.168.1.20 durch die tatsächliche LAN-IP-Adresse des Servers ersetzt werden.

NUT verwendet standardmäßig den TCP-Port:

```
\3493
```

Die erste Zeile erlaubt lokale Zugriffe:

```
LISTEN 127.0.0.1 \3493
```

Die zweite Zeile erlaubt Zugriffe aus dem lokalen Netzwerk:

```
LISTEN 192.168.1.20 \3493
```

8. NUT-Benutzer und Passwörter anlegen

Für den lokalen Master und die Netzwerk-Clients sollten unterschiedliche Passwörter verwendet werden.

Zwei Passwörter erzeugen:

```
openssl rand -hex \24
```

Den Befehl zweimal ausführen und beide Passwörter sicher speichern.

Die Datei `/etc/nut/upsd.users` öffnen:

```
sudo nano /etc/nut/upsd.users
```

Ubuntu 22.04 beziehungsweise NUT 2.7

```
[monmaster]
password = \MASTER_PASSWORT_HIER_EINTRAGEN
upsmon master

[monclient]
password = \CLIENT_PASSWORT_HIER_EINTRAGEN
upsmon \slave
```

Ubuntu 24.04 beziehungsweise NUT 2.8

```
[monmaster]
password = \MASTER_PASSWORT_HIER_EINTRAGEN
upsmon primary

[monclient]
password = \CLIENT_PASSWORT_HIER_EINTRAGEN
upsmon \secondary
```

<note warning> Die Platzhalter müssen durch echte Passwörter ersetzt werden.

Die Passwörter sollten nicht in einer öffentlich zugänglichen Dokumentation gespeichert werden.
</note>

9. Lokalen NUT-Monitor konfigurieren

Die Datei `/etc/nut/upsmon.conf` öffnen:

```
sudo nano /etc/nut/upsmon.conf
```

Vorhandene aktive MONITOR-Zeilen sollten auskommentiert oder entfernt werden.

Ubuntu 22.04 beziehungsweise NUT 2.7

```
MONITOR apc@localhost 1 monmaster MASTER_PASSWORT_HIER_EINTRAGEN master
MINSUPPLIES 1
SHUTDOWNCMD "/sbin/shutdown -h +0"
POWERDOWNFLAG /etc/\killpower
```

Ubuntu 24.04 beziehungsweise NUT 2.8

```
MONITOR apc@localhost 1 monmaster MASTER_PASSWORT_HIER_EINTRAGEN primary
MINSUPPLIES 1
SHUTDOWNCMD "/sbin/shutdown -h +0"
POWERDOWNFLAG /etc/\killpower
```

Das Passwort muss mit dem Passwort des Benutzers monmaster in `/etc/nut/upsd.users` übereinstimmen.

Bedeutung der wichtigsten Einträge:

Einstellung	Bedeutung
apc@localhost	USV mit dem Namen apc auf dem lokalen Server
1	Power-Wert der USV
monmaster	Benutzer aus <code>upsd.users</code>
master beziehungsweise primary	Rolle des direkt angeschlossenen Servers
MINSUPPLIES 1	Mindestens eine Stromversorgung muss verfügbar sein
SHUTDOWNCMD	Befehl zum Herunterfahren des Servers
POWERDOWNFLAG	Kennzeichnung für das spätere Abschalten der USV

10. Dateirechte absichern

Die Konfigurationsdateien enthalten Passwörter und sollten nur für root und die Gruppe nut lesbar sein.

```
\\
sudo chown root:nut /etc/nut/upsmon.conf /etc/nut/upsd.users\\
```

```
sudo chmod 640 /etc/nut/upsmon.conf /etc/nut/upsd.\users
```

Dateirechte kontrollieren:

```
ls -l /etc/nut/upsmon.conf /etc/nut/upsd.\users
```

Eine mögliche Ausgabe lautet:

```
-rw-r----- 1 root nut /etc/nut/upsd.\users  
-rw-r----- 1 root nut /etc/nut/upsmon.\conf
```

11. NUT-Dienste starten

Ubuntu 22.04

Dienste aktivieren:

```
sudo systemctl enable nut-server nut-\monitor
```

NUT-Server neu starten:

```
sudo systemctl restart nut-\server
```

NUT-Monitor neu starten:

```
sudo systemctl restart nut-\monitor
```

Status anzeigen:

```
systemctl --no-pager --full status nut-server nut-\monitor
```

Ubuntu 24.04

Unter NUT 2.8 wird zusätzlich der Driver Enumerator verwendet.

```
sudo systemctl restart nut-driver-enumerator.\service
```

```
sudo systemctl restart nut-server.\service
```

```
sudo systemctl restart nut-monitor.\service
```

Status anzeigen:

```
systemctl --no-pager --full status nut-driver-enumerator.service nut-  
server.service nut-monitor.\service
```

12. USV-Status prüfen

Alle verfügbaren USV-Werte anzeigen:

```
upsc apc@\localhost
```

Einzelne Werte abfragen:

```
upsc apc@localhost ups.\model
```

```
upsc apc@localhost ups.\status
```

```
upsc apc@localhost battery.\charge
```

```
upsc apc@localhost battery.\runtime
```

```
upsc apc@localhost ups.\load
```

Bei normalem Netzbetrieb sollte der Status Folgendes enthalten:

```
\OL
```

Wichtige Statusmeldungen:

Status	Bedeutung
OL	Online: Netzversorgung vorhanden
OB	On Battery: USV läuft auf Batterie
LB	Low Battery: Batterie fast leer
RB	Replace Battery: Batterie sollte ersetzt werden
CHRG	Batterie wird geladen
DISCHRG	Batterie wird entladen
OVER	USV ist überlastet
OFF	USV-Ausgang ist ausgeschaltet

Mehrere Statuswerte können gleichzeitig angezeigt werden.

Beispiel:

```
\\  
OL CHRG\\
```

Das bedeutet:

* Netzversorgung \vorhanden * Batterie wird geladen

13. Netzwerkzugriff mit UFW erlauben

Wenn UFW aktiv ist, sollte der NUT-Port nur für das interne Servernetz freigegeben werden.

Beispiel für das Netzwerk 192.168.1.0/24:

```
sudo ufw allow from 192.168.1.0/24 to any port 3493 proto \tcp
```

Firewallstatus prüfen:

```
sudo ufw \status
```

Prüfen, ob NUT auf Port 3493 lauscht:

```
sudo ss -ltnp | grep \3493
```

Eine mögliche Ausgabe lautet:

```
LISTEN 0 16 127.0.0.1:\3493
LISTEN 0 16 192.168.1.20:\3493
```

14. Zugriff von einem anderen Server testen

Auf dem zweiten Ubuntu-Server den NUT-Client installieren:

```
sudo apt \update
```

```
sudo apt install nut-\client
```

Status des NUT-Masters abfragen:

```
upsc apc@192.168.1.\20
```

Wenn die Verbindung funktioniert, werden die USV-Werte angezeigt.

Falls folgende Meldung erscheint:

```
Error: Connection failure: Connection \refused
```

sollten folgende Punkte kontrolliert werden:

* Ist nut-server gestartet? * Ist die richtige IP in upsd.conf eingetragen? * Ist TCP-Port 3493 durch die Firewall freigegeben? * Ist die IP-Adresse vom Client erreichbar?

15. Vorsichtiger Stromausfalltest

Vor dem Test sollte die Batterie vollständig geladen sein.

Die wichtigsten Werte laufend anzeigen:

```
watch -n 2 'upsc apc@localhost ups.status; upsc apc@localhost battery.charge; upsc apc@localhost battery.runtime'
```

Anschließend nur den **Netzeingang der USV** kurz vom Stromnetz trennen.

Dabei sollten folgende Änderungen sichtbar sein:

Zustand	Erwarteter Status
Netzstrom vorhanden	OL
Netzstrom getrennt	OB
Netzstrom wiederhergestellt	OL

Die angeschlossenen Server müssen während des kurzen Tests weiterlaufen.

<note important>

Nicht die Ausgangsstecker der Server abziehen.

Nur den Netzanschluss der USV testweise vom Stromnetz trennen.

</note>

16. Gefährliche Testbefehle vermeiden

Der folgende Befehl darf nicht während des normalen Betriebs testweise ausgeführt werden:

```
\\  
upsdrvctl shutdown\\
```

Dieser Befehl kann den Ausgang der USV abschalten.

Dadurch verlieren alle an der USV angeschlossenen Geräte unmittelbar ihre Stromversorgung.

17. Fehlerdiagnose

Ubuntu-Version anzeigen:

```
lsb_release -\a
```

USB-Geräte anzeigen:

```
\lsusb
```

Status der Dienste anzeigen:

```
sudo systemctl status nut-server nut-monitor --no-pager
```

Systemprotokoll anzeigen:

```
sudo journalctl -u nut-server -u nut-monitor -n 100 --no-pager
```

Unter Ubuntu 24.04 zusätzlich:

```
sudo journalctl -u nut-driver-enumerator.service -u nut-server.service -u nut-monitor.service -n 150 --no-pager
```

Fehler: Driver failed to start

Treiber stoppen:

```
sudo upsdrvctl \stop
```

Treiber testweise im Debug-Modus starten:

```
sudo /lib/nut/usbhid-ups -a apc -\DDD
```

Den Debug-Modus mit Strg+C beenden.

Danach den normalen Dienst wieder starten:

```
sudo systemctl restart nut-server
```

Fehler: Can't claim USB device

Prüfen, ob ein anderer USV-Dienst auf das Gerät zugreift:

```
ps aux | grep -E 'apcupsd|usbhid-ups|upsd|upsmon'
```

Falls apcupsd installiert ist, kann es zu einem Konflikt kommen.

Status prüfen:

```
systemctl status \apcupsd
```

Falls NUT verwendet werden soll, apcupsd stoppen und deaktivieren:

```
sudo systemctl disable --now \apcupsd
```

Danach NUT neu starten:

```
sudo systemctl restart nut-server nut-\monitor
```

Fehler: Data stale

Treiberstatus prüfen:

```
sudo systemctl status nut-server --no-\pager
```

Protokoll prüfen:

```
sudo journalctl -u nut-server -n 100 --no-\pager
```

USB-Gerät prüfen:

```
lsusb | grep -i -E 'APC|American Power Conversion'
```

USB-Kabel und Batterieanschluss der USV kontrollieren.

18. Konfigurationsdateien im Überblick

Datei	Aufgabe
/etc/nut/nut.conf	Legt die NUT-Betriebsart fest
/etc/nut/ups.conf	Definiert die angeschlossene USV und den Treiber
/etc/nut/upsd.conf	Legt Netzwerkadressen und Port des NUT-Servers fest
/etc/nut/upsd.users	Definiert Benutzer, Passwörter und Berechtigungen
/etc/nut/upsmon.conf	Konfiguriert Überwachung und automatisches Herunterfahren

19. Beispiel einer vollständigen Konfiguration

/etc/nut/nut.conf

```
\\  
MODE=netserver\\
```

/etc/nut/ups.conf

```
\\  
[apc]\\  
driver = usbhid-ups\\  
port = auto\\  
vendorid = 051d\\  
desc = "APC Smart-UPS SMC1500IC"\\
```

/etc/nut/upsd.conf

```
\\  
LISTEN 127.0.0.1 3493\\  
LISTEN 192.168.1.20 3493\\
```

/etc/nut/upsd.users für Ubuntu 22.04

```
\\  
[monmaster]\\  
password = MASTER_PASSWORT_HIER_EINTRAGEN\\  
upsmon master  
  
[monclient]\\  
password = CLIENT_PASSWORT_HIER_EINTRAGEN\\  
upsmon slave\\
```

/etc/nut/upsmon.conf für Ubuntu 22.04

```
\\  
MONITOR apc@localhost 1 monmaster MASTER_PASSWORT_HIER_EINTRAGEN master  
  
MINSUPPLIES 1  
  
SHUTDOWNCMD "/sbin/shutdown -h +0"  
  
POWERDOWNFLAG /etc/killpower\\
```

Ergebnis

Nach erfolgreicher Einrichtung sollte folgender Befehl die USV-Daten anzeigen:

```
\\  
upsc apc@localhost\\
```

Bei normaler Netzversorgung sollte unter anderem folgender Status erscheinen:

```
\\  
ups.status: 0L\\
```

Der NUT-Master kann nun:

- * die APC Smart-UPS SMC1500IC überwachen
- * einen Stromausfall erkennen
- * den lokalen Ubuntu-Server herunterfahren
- * weitere NUT-Clients im Netzwerk versorgen

From:

<http://wiki.waldhofer.at/> - **Wiki von Franz**

Permanent link:

<http://wiki.waldhofer.at/doku.php?id=ubuntu:ups&rev=1784456301>

Last update: **2026/07/19 12:18**

