

# USV verwalten mit acupsd

- Kurzakkte -

## Server:

<i>Installation</i>	
<ul style="list-style-type: none"> <li>• acupsd aus Paketgruppe: Dokumentation/Andere nach /sbin/apcupsd</li> <li>• Anschluss des seriellen Kabels zwischen USV und Server</li> <li>• alternativ apcupsd-x.x.x.tar.gz von der Webseite <a href="http://sourceforge.net/projects/apcupsd">http://sourceforge.net/projects/apcupsd</a>  <code>./configure --prefix=/usr/local/src/apcupsd --sbindir=/sbin \</code>  <code>--with-log-dir=/var/log/apcupsd --enable-pthreads --enable-net</code> </li> </ul>	
<i>persönliche Merkmale</i>	
<ul style="list-style-type: none"> <li>• Slave-Server werden am Port 3551 mit Statusinformationen der USV bedient</li> </ul>	
<i>Konfiguration</i>	
/etc/apcupsd/apcupsd.conf	<ul style="list-style-type: none"> <li>• Eintrag des Typs der UPS und der Datenleitung</li> <li>• Start des Network Information Servers NIS zur Bereitstellung von Statusinformationen der UPS für die Slave-Server  <code>NETSERVER on</code>  <code>NISPORT 3551</code> </li> </ul>
/etc/apcupsd/apccontrol	<ul style="list-style-type: none"> <li>• wenn acupsd ein Problem entdeckt, generiert er ein Event, das durch das Script apccontrol gehandelt wird  z. B.: Abziehen des seriellen Kabels generiert nach 6 sec einen wall-Broadcast, eine mail an root, informiert den systemlog und loggt ins eventlogfile  <code>/etc/apcupsd/apcupsd.events</code> </li> <li>• Achtung: Skript nicht anfassen!</li> </ul>
/etc/aliases	<ul style="list-style-type: none"> <li>• Umleitung der durch acupsd generierten Mail von daemon auf user daemon: user</li> <li>• je nach MTA muss daraus noch in eine Datei mit Hashwerten generiert werden (sendmail). Exim kommt damit auch so klar.</li> </ul>
<i>Start als Stand-Alone-Dienst</i>	
/etc/init.d/apcupsd rcapcupsd (S.u.S.E.)	<ul style="list-style-type: none"> <li>• start   stop   restart   reload   status - Skript startet/stoppt Dämonen acupsd und den NIS-Server manuell</li> </ul>
insserv apcupsd	<ul style="list-style-type: none"> <li>• automatischer Start des Samba-Servers (alternativ mit Runleveleditor im Yast)</li> </ul>
<i>Funktionsprüfung</i>	
/sbin/apcaccess status	<ul style="list-style-type: none"> <li>• Statusinformationen der USV auslesen</li> </ul>
/sbin/apcnisd	<ul style="list-style-type: none"> <li>• Network Information Server mit Satusinformationen über die UPS</li> <li>• Clients holen sich diese Informationen durch Polling des Serverports 3551</li> </ul>
/sbin/apctest	<ul style="list-style-type: none"> <li>• Testprogramm, kommuiziert mit UPS <ul style="list-style-type: none"> <li>Punkt 1: Output der Statusinfos im raw-Format</li> <li>Punkt 2: Kalibrierung der Batterie mit Entladung bis 30%, einmal jährlich empfohlen.</li> <li>Punkt 3: Abbruch einer laufenden Kalibrierung</li> <li>Punkt 4: Neustart des Monitorings der Kalibrierung</li> <li>Punkt 5: Programmierung des EEPROMs</li> <li>Punkt 6: direkte Kommunikation mit der UPS für raw-Kommandos (Vorsicht!)</li> <li>Punkt 7: beendet apctest</li> </ul> </li> </ul>
/sbin/apcupsd	<ul style="list-style-type: none"> <li>• Serverprogramm mit Optionen zur Beeinflussung der Eigenschaften der UPS</li> <li>• Hilfe mit <code>apcupsd --help</code></li> </ul>
ps afx	<ul style="list-style-type: none"> <li>• Check, ob der Daemon apcupsd läuft</li> </ul>
/var/log/messages	<ul style="list-style-type: none"> <li>• System-Logdatei</li> </ul>
/var/log/apcupsd.events	<ul style="list-style-type: none"> <li>• temporäre Logdatei</li> </ul>
Serielles Kabel abziehen	<ul style="list-style-type: none"> <li>• Warnmeldungen über wall und Mail</li> </ul>
Netzstecker ziehen an USV	<ul style="list-style-type: none"> <li>• Empfehlung: zum test TIMEOUT auf 60 setzen (Shutdown nach 60 sec)</li> </ul>
<i>Dokumentation</i>	
man: apcupsd (8)	

### Konfigurationsdatei /et/apcupsd/apcupsd.conf

```
UPSCABLE smart          # definiert den Kabeltyp
UPSTYPE smartups        # definiert den UPS-Typ
DEVICE /dev/ttyS0       # Geräteverbindung zur UPS
LOCKFILE /var/lock      # Pfad zur Sperrdatei der seriellen Schnittstelle
ONBATTERYDELAY 6        # Zeit in sec vom Entdecken eines Stromausfalls bis zum onbattery-Event
                        # Das erste erreichte Ereignis bewirkt das Herunterfahren:
BATTERYLEVEL 5         # Herunterfahren bei Unterschreiten der Restladung in %
MINUTES 3               # Herunterfahren bei Unterschreitung der Restlaufzeit der UPS in sec
TIMEOUT 0               # Timer zum Herunterfahren bei Stromausfall nach x sec (0 disabled)
ANNOY 300              # Aufforderung zum Ausloggen vor Shutdown alle 5 min
ANNOYDELAY 60          # erste Aufforderung zum Ausloggen 60 sec nach Stromausfall
NOLOGON disable        # Verhindert das Einloggen mit /etc/nologin nach bestimmter Zeit
KILLDELAY 0            #
NETSERVER on           # schaltet Network Information Server ein (Mastermodus)
NISIP 0.0.0.0          # NIS-Server lauscht auf alle eingehenden Verbindungen
NISPORT 3551           # Standardport des NIS-Servers (registriert bei IANA)
EVENTSFILE /var/log/apcupsd/apcupsd.events
EVENTSFILEMAX 10       # Datei überschreitet nie 10 Kbyte (älteres wird überschrieben)
UPSCCLASS standalone
UPSMODE disable
STATTIME 0             # Intervall zwischen dem Schreiben des Statusfiles
STATFILE /var/log/apcupsd/apcupsd.status # nur wenn STATTIME nicht 0
LOGSTATS off          # on erzeugt eine Menge Output
DATATIME 0            # Zeitintervall zum Schreiben der DATA-Records
```

### Standalone-Konfiguration

Für Standalone nur 3 Parameter wichtig:

Parameter / Kommunikation über:	serielle Verbindung	USB-Verbindung
UPSTYPE	smartups	usb
UPSCABLE	smart	usb
DEVICE	/dev/ttyS0	empfohlen: Auskommentieren dieser Direktive

### eine einfache Konfigurationsdatei für eine Smart-UPS

```
UPSCABLE smart
UPSTYPE smartups
DEVICE /dev/ttyS0
LOCKFILE /var/lock
UPSCCLASS standalone
UPSMODE disable
```

### Master-Konfiguration über NIS mit NetDriver:

Die UPS ist via Kabel mit dem Master-Server verbunden. Der Master kennt die Slaves nicht, macht die UPS-Statusinformationen aber über das Netzwerk verfügbar.

```
UPSCABLE smart          # Server direkt mit UPS verbunden
UPSTYPE smartups        # eine SmartUPS
DEVICE /dev/ttyS0       # Verbindung zur UPS via seriellem Kabel
LOCKFILE /var/lock
UPSCCLASS standalone   # netmaster nur für alten Master/Slave-Modus
UPSMODE disable        # abschalten, da für alten Master/Slave-Modus
NETSERVER on           # aktiviert den NIS-Server für Statusinformationen
NISPORT 3551           # Port mit verfügbarem UPS-Status
BATTERYLEVEL 10        # Shutdown bei Ladezustand unter 10%
```

### Slave-Konfiguration:

Die Slaves fragen den Master im Polling-Modus ab. Beim Status powerfail fährt der Slave geordnet herunter.

```
UPSCABLE ether          # Serververbindung via Netzwerkkarte
UPSTYPE net             # Netzwerkverbindung
DEVICE 192.168.1.1:3551 # Serveradresse und NIS-Port
LOCKFILE /var/lock
UPSCCLASS standalone   # netslave nur für alten Master/Slave-Modus
UPSMODE disable        # abschalten, da für alten Master/Slave-Modus
NETTIME 10             # Zeitintervall des Pollings zum Server in sec
BATTERYLEVEL 25        # Shutdown bei Ladezustand unter 25%
```