

Installation of Nut v0.45.4-1 on Linux with the APC BackUPS Pro 280VA

(v1.0, 27Sept2002)

This How-To explains the steps involved in setting up a Dell or Penguin computer system with Red Hat Linux v7.3 for use with the APC BackUPS Pro 280VA uninterruptible power supply (UPS) and Network UPS Tools (NUT) software. The document is divided into 7 sections:

- 1. Requirements**
- 2. Hardware Configuration**
- 3. Software, Part 1** (installation and configuration)
- 4. OS Configuration**
- 5. Software, Part 2** (configuration)
- 6. Testing**
- 7. Troubleshooting**

1. Requirements:

Red Hat Linux v7.3 (RH7.1)
nut-0.45.4-1.i386.rpm
nut-client-0.45.4-1.i386.rpm
nut-cgi-0.45.4-1.i386.rpm
APC BackUPS Pro 280VA
APC UPS cable (PN: 940-0095B)

Note that the version of NUT listed here is has been tested and found to work with RedHat v7.3. It has not been tested on older systems so compatibility is uncertain.

New and/or beta versions of NUT are available for download at:

<http://www.exploits.org/nut>

2. Hardware Configuration

Make available one COM port and connect the UPS using the APC UPS cable with part number 940-0095B. The cable should have been supplied with the UPS unit. Otherwise request one from APC and they will send it out free of charge.

Turn the computer on and enter the bios settings display. Shortly after you power on the computer system you should see a brief message showing how to do this. Usually, depressing the F2 or Delete key during the boot process will get you there.

Once in, look for power settings and ensure that the following requirement is met:

state after AC power failure should be ON upon power restore

Save the change and continue booting.

3. Software Installation and Configuration, Part 1 (install)

Log in as root.

Obtain the following nut rpm files from the UNAVCO software download site:

1. nut-client-0.45.4-1.i386.rpm
2. nut-0.45.4-1.i386.rpm
3. nut-cgi-0.45.4-1.i386.rpm

Install the rpm files in the order listed above as follows:

```
[root@localhost]# rpm -Uvh <nut rpm>
```

The following files should have been installed:

/etc/ups/hosts.conf	(configuration files)
./multimon.conf	
./ups.conf	
./upsd.conf	
./upsd.users	
./upsmon.conf	
./upssched.conf	
./upsset.conf	
/etc/init.d/ups	(ups service)
/etc/sysconfig/ups	(additional configuration file)
/usr/sbin/upsd	(ups daemon)
./upsmon	(ups monitor)
./upssched	(event scheduler)
./upssched-cmd	(event scheduler tasks)
/usr/bin/apcsmart	(ups driver)

Note that there are other files that are not listed here that should have been installed. They are drivers for other UPSs and utilities for functions which will not be used. Some files listed above that are greyed out will not be used as well.

4. OS Configuration

Edit system file:

```
/etc/rc.d/init.d/halt
```

Just about at the end of file you'll find these lines:

```
if [ "$command" = halt ] ; then
if [ -r /etc/ups/upsmon.conf -a -f /etc/killpower -a -f /etc/sysconfig/ups ] ; then
. /etc/sysconfig/ups
[ "$SERVER" = "yes" -a "$MODEL" != "NONE" -a -n "$MODEL" -a -n "$DEVICE" ] && $MODEL -k $DEVICE
fi
fi
```

Change the line in boldface to:

```
[ "$SERVER" = "yes" -a "$MODEL" != "NONE" -a -n "$MODEL" -a -n "$DEVICE" ] && $MODEL -a $DEVICE -k
```

Note that the changes are highlighted in red.

Explanation:

The statements listed above translate to the following:

1. if a 'halt' command has been issued
2. check to see if the upsmon, killpower, and ups files exist
3. if they do exist, source the /etc/sysconfig/ups file (the variables that are assigned within that file are exported to the shell environment.
4. if the group of variables listed in the square brackets meet the conditions stated within then execute \$MODEL -a \$DEVICE -k

The last command tells the UPS to shut down. This is the last statement that is executed after the OS itself has successfully unmounted the filesystems.

\$MODEL is the upsdriver, *apcsmart*

\$DEVICE is the ups as described in /etc/ups.conf

5. Software Installation and Configuration, Part 2 (configuration)

A. Edit file:

```
/etc/rc.d/init.d/ups
```

Just short of half-way down you should see these lines:

```
echo -n $"Starting UPS monitor (master): "  
daemon /usr/sbin/upsmon  
echo  
  
else  
  
echo -n $"Starting UPS monitor (slave): "  
daemon /usr/sbin/upsmon  
echo
```

Add to *both* lines in boldface such that:

```
daemon /usr/sbin/upsmon -p
```

Note that the change is highlighted in red.

Explanation:

The *-p* flag tells `upsmon` to run in privileged mode always. In other words, all child processes of `upsmon` will inherit root privileges from the `upsmon` parent process.

When `upsmon` (owned by *root*) is run it will spawn one child process which is owned by the user *nobody*.

B. Edit file: `/etc/sysconfig/ups`

Add these lines to the file (everything else may be deleted):

```
SERVER=yes  
MODEL=apcsmart  
DEVICE=myups  
OPTIONS="-a"
```

Explanation:

SERVER:	The UPS is connected to the same machine that NUT is controlling
MODEL:	The name of the device driver for the type of UPS being used
DEVICE:	The name assigned to the UPS. It may be arbitrary but it must be consistent across all configuration files where applicable.
OPTIONS:	Arguments passed to the device driver (MODEL). <i>-a</i> flag tells the driver to read the <code>/etc/ups/ups.conf</code> configuration file which contains information on where and how the UPS is connected.

C. Edit file `/etc/ups/ups.conf`

Add these lines to the file (everything else may be deleted):

```
[myups]
driver=apcsmart
port=/dev/ttyS1
cable=940-0095B
```

Explanation:

<code>[myups] :</code>	The name assigned to the UPS. It may be arbitrary but it must be consistent across all configuration files where applicable. Must be placed within <code>[]</code> .
<code>driver:</code>	The name of the device driver for the type of UPS being used
<code>port:</code>	The port to which the UPS is connected (e.g., <code>/dev/ttyS1</code>)
<code>cable:</code>	The APC cable with part number 940-0095B. This cable is required. Although others may work they have not been tested. The cable type must be specified since the default configuration assumes a different cable.

D. Edit file `/etc/ups/upsd.conf`

Add these lines to the file (everything else may be deleted):

```
ACL all 0.0.0.0/0
ACL localhost 127.0.0.1/32
ACCESS grant master localhost password
ACCESS deny all all
```

Explanation:

<code>ACL:</code>	Access Control List. Lists the hosts that may access the server.
<code>ACCESS:</code>	What level of access to grant, to whom, and what password to use.

E. Edit file `/etc/ups/upsmon.conf`

Add or modify existing lines within file to match the following:

```
MONITOR myups@localhost 1 password master
MINSUPPLIES 1
SHUTDOWNCMD "/sbin/shutdown -h +0"
NOTIFYCMD /usr/sbin/upssched
POLLFREQ 10
POLLFREQUALERT 5
HOSTSYNC 15
DEADTIME 15
POWERDOWNFLAG /etc/killpower
UPSNAME myups@locahhost
NOTIFYTYPE ONBATT
NOTIFYFLAG ONBATT SYSLOG+WALL+EXEC
NOTIFYFLAG ONLINE SYSLOG+WALL+EXEC
NOTIFYFLAG LOWBATT SYSLOG+WALL+EXEC
NOTIFYFLAG SHUTDOWN SYSLOG+WALL+EXEC
RBTWARNTIME 43200
NOCOMMWARNTIME 300
FINALDELAY 15
```

Explanation:

MONITOR:	Which UPS to monitor.
MINSUPPLIES:	How many UPSs that are being monitored. Cannot be less than 1.
SHUTDOWNCMD:	Which command to issue for shutdown.
NOTIFYCMD:	Which script to run when an event occurs (e.g., power failure).
POLLFREQ and POLLFREQUALERT:	Polling frequency.
HOSTSYNC:	Only used when NUT is serving more than one system.
DEADTIME:	Time to wait before warning about an unresponsive UPS.
POWERDOWNFLAG:	The file the ups monitoring software will look for when its time to shutdown. This signals the shutdown process.
UPSNAME:	Name of the ups in the form <code>[upsname]@localhost</code> . Required for <code>upssched</code> .
NOTIFYTYPE	Notify on type of event specified. Required for <code>upssched</code> .
NOTIFYFLAG	On event type (e.g., power failure, back line, etc.) send notification via syslog (<code>/var/log/messages</code>) or using the unix <code>wall</code> command or run an executable.
RBTWARNTIME:	Replace battery warning. Duration before warning in seconds.
NOCOMMWARNTIME:	Amount of time to wait before warning of no communication with UPS.
FINALDELAY:	Time to wait before final shutdown.

F. Edit file `/etc/ups/upssched.conf`

Add or modify existing lines in the file to match the following:

```
CMDSCRIPT /usr/sbin/upssched-cmd
PIPEFN /var/run/upssched.pipe
AT ONBATT * START-TIMER onbattwarn 120
AT ONLINE * CANCEL-TIMER onbattwarn
```

Explanation:

CMDSCRIPT: The script or executable that will perform the actions specified in this and any other related configuration file (e.g., `/usr/sbin/upssched-cmd`).

AT ONBATT/: Do action specified when certain condition exists. E.g., when on battery, start the shutdown timer. When back online, cancel the shutdown timer.

G. Edit file `/usr/sbin/upssched-cmd`

Add or modify existing lines in the file to match the following:

```
#!/bin/sh
case $1 in
    onbattwarn)
        echo "Utility power failure...shutting down..."; /usr/sbin/upsmon -c fsd
        ;;
    *)
        logger -t upssched-cmd "Unrecognized command: $1"
        ;;
esac
```

Explanation:

This script is executed during the final stage of system shutdown when the read-only filesystems are remounted. A command is issued to the UPS instructing it to turn off. The computer system and UPS will remain on for about twenty seconds after this command is sent.

H. Remaining Items...

Be sure that ownership/group and permissions for each file is set correctly. They should be as follows:

-rw-----	1	root	root	747	Feb	26	2002	hosts.conf
-rw-----	1	root	root	2174	Feb	26	2002	multimon.conf
-rw-----	1	root	root	441	Sep	27	15:02	ups.conf
-rw-----	1	root	root	465	Sep	27	15:02	upsd.conf
-rw-----	1	root	root	2179	Feb	26	2002	upsd.users
-rw-----	1	root	root	791	Sep	27	15:02	upsmon.conf
-rw-----	1	root	root	535	Sep	27	15:02	upssched.conf
-rw-----	1	root	root	1420	Feb	26	2002	upsset.conf
-rwxr-xr-x	1	root	root	1656	Sep	27	15:02	/etc/init.d/ups
-rwxr-xr-x	1	root	root	19331	Feb	26	2002	/usr/sbin/upssched
-rwxr-xr-x	1	root	root	602	Sep	27	15:02	/usr/sbin/upssched-cmd
-rw-r--r--	1	root	root	412	Sep	27	15:02	/etc/sysconfig/ups
-rwxr-xr-x	1	root	root	5098	Sep	27	15:02	/etc/init.d/halt

Add the new UPS service and activate it in runlevels 3, 4, and 5 by issuing the following commands:

```
[root@localhost]# /sbin/chkconfig --add ups
[root@localhost]# /sbin/chkconfig --levels 345 ups on
```

Check to see that it has been turned on:

```
[root@localhost]# /sbin/chkconfig --list | grep ups
ups                0:off   1:off   2:off   3:on    4:on    5:on    6:off
```

You are done! Reboot the system and begin testing.

6. Testing

Simulate a power failure to test the machine. Perform the following three tests to confirm a properly working system:

- disconnect power, wait for full shutdown (inverter kill), restore power
- disconnect power *during bootup*, wait for full shutdown, restore power
- disconnect power, wait for shutdown process to begin, then *restore power during shutdown* process

The computer should recover from all these circumstances. Keep in mind the time delays you have set. Impatience may bring about the appearance of a system not working properly. Just wait and see what happens keeping in mind the time delay you set. For the purpose of expediency you may want to temporarily set the shutdown timer to 30 seconds (or less), for example. Just remember to change them back after confirming positive test results. If the system does not appear to be functioning as you would expect, refer to the troubleshooting section of this document.

7. Troubleshooting

Problem	Remark
1. NUT does not appear to install properly or won't install.	<i>a.</i> Check that the version of rpm file was created for NUT matches rpm installed on the OS.
2. driver does not appear to be communicating with the UPS	<i>a.</i> Make sure you're using the correct UPS serial cable (940-0095B). <i>b.</i> Did you choose the right UPS driver? <i>c.</i> Is there a port conflict? Make sure the port is available for use. <i>d.</i> Is the UPS plugged into the port?
3. Computer won't shut down after disconnecting utility power.	<i>a.</i> Did you reboot machine after installing the software? <i>b.</i> Refer to Remarks for Problem #2
4. Computer powers down completely after disconnecting utility power. Does not reboot automatically after restoring utility power. Must manually turn it back on.	Remove the <i>-p</i> option from the <code>HALTARGS="HALTARGS -p"</code> line in <code>/etc/rc.d/init.d/halt</code> file.
5. Computer remains on after cutting utility power to the UPS and the UPS inverter shuts down.	Did you plug the computer into the UPS in the first place???
6. The OS is halted (proper shutdown) and the inverter shuts off after utility power disconnect but after restoring the utility power the UPS turns back on but the computer doesn't.	Make sure the bios power settings are set so that the computer is on after a power failure recovery.
7. If all else fails...	Contact Jay Sklar via sklarj@ucar.edu