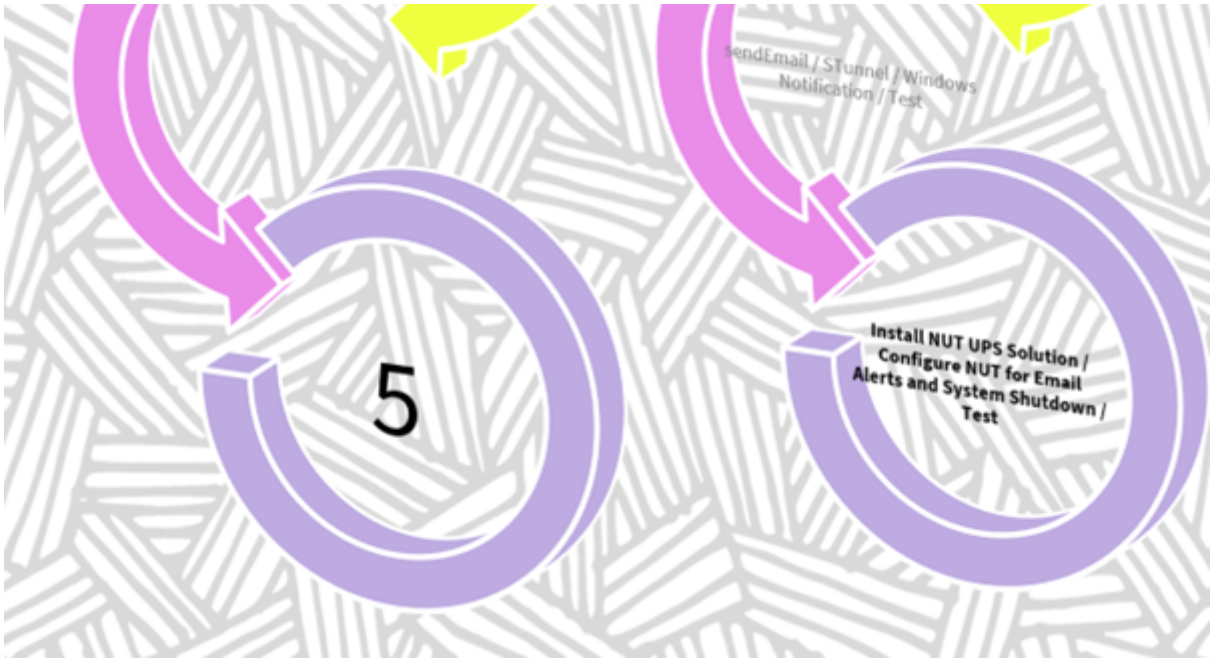
Proxmox-NUT Homelab HOWTO — Step 5 : Install NUT UPS Solution / Configure NUT for Email Alerts and System Shutdown / Test

chribonn.medium.com/proxmox-nut-homelab-howto-step-5-install-nut-ups-solution-configure-nut-for-email-alerts-and-be74838fdccb

Alan C. Bonnici January 16, 2025

Alan C. Bonnici



NUT will be installed on Debian (Proxmox) and will be linked to a single APC UPS. When there is a power failure, NUT will send an email. Its status will change from **Online** (**OL**) to **On Battery** (**OB**).

When either the remain battery percentage charge goes below a certain value (85% in this example) or the remaining run time goes below a certain amount (15 minutes in this example), the UPS status will change from **OB** to **Low Battery** (**LB**).

When the status becomes **LB**, NUT will send out an email and start shutting down Debian. As part of the Proxmox shutdown process it will shut down all guests running with it and then shut down itself. After that the OS shuts down.

If the LB condition does not trigger within 5 minutes the UPS will shut down Proxmox. This is discussed further below.

NUT manages other UPS status such as whether it is discharging and if the battery needs replacing. These setting will be factored into the configuration files but will not be discussed here.

After the **offdelay** number of seconds have passed, the UPS powers itself down.

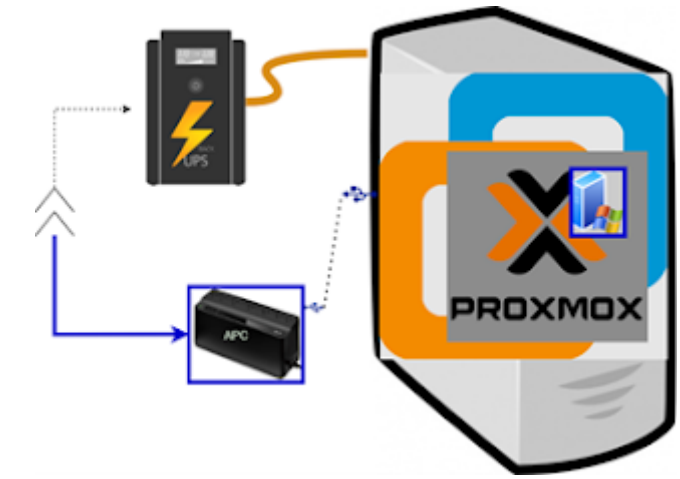
The VMWare Proxmox-NUT VM

Because Proxmox is installed as a guest VM in VMWare and because the session needed to be recorded the UPS setup had to be adjusted to cater for this situation.

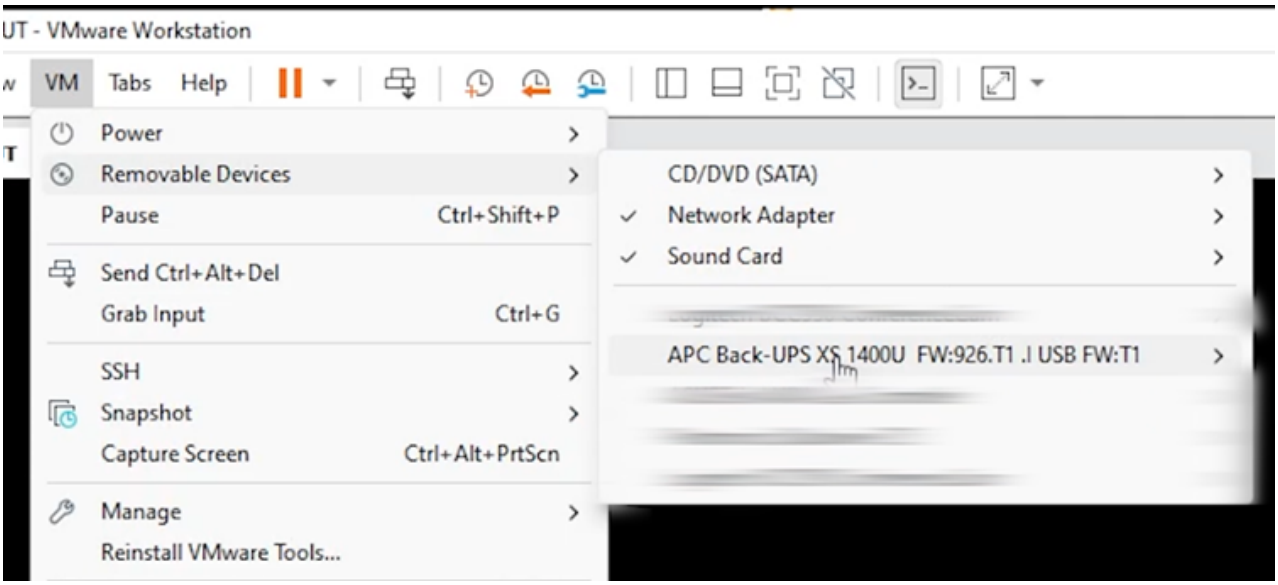
The setup is described in the diagram shown here.

A UPS was connected to the power input of a physical computer. A USB cable was not plugged in.

A second UPS (APC) was connected to the power source but was not plugged into anything. There was a USB cable between the APC and the physical computer.



When VMWare's Proxmox image was started, the APC was connected to the VM.



Install and Configure NUT

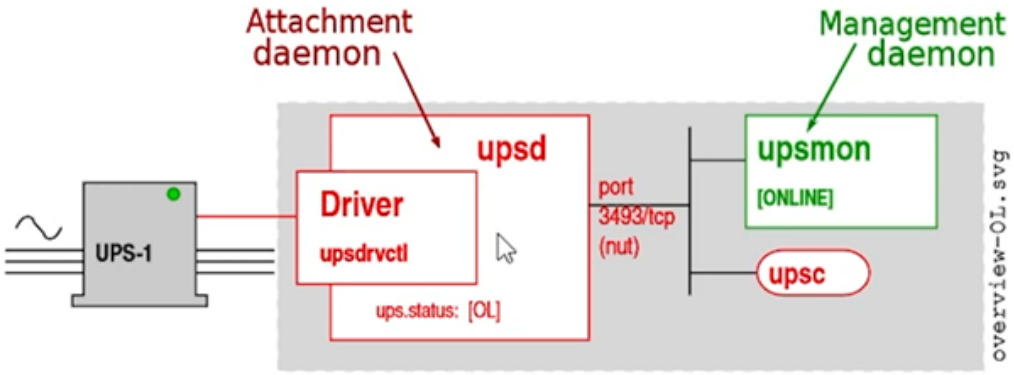


Figure 1: Overview of NUT.

The image above, from https://rogerprice.org/NUT/ConfigExamples.A5.pdf, consists of a number of daemons communicating with each other to manage UPS and the devices they are supporting.

NUT is not tied to any brand of UPS. The communication between the NUT and the UPS hardware is achieved through drivers (Driver in the image above). Through the driver the Attachment daemon exposes various ups parameters. The NUT solution will, based on the value of these parameters protect the equipment connected to the ups.

Many of the parameters are RO but some parameters may be RW. RW parameters can be altered by the NUT daemon or a provided NUT utility program.

Even though the above image shows one UPS, NUT is very flexible and can support setups with multiple UPSs working in sync. This topic outside the scope of the Proxmox-NUT topic.

Not all UPSs have the same capabilities or support identical functions. Further different UPS brands may trigger a state differently. For example, with certain UPS, Low Battery status indicates that the UPS is in a critical condition and will imminently power down. Another UPS brand would raise a Low Battery status earlier giving one time to shut down the supported equipment. The NUT website provides a database of supported UPS.

Installing NUT

1. In Proxmox open a terminal window and type the following:

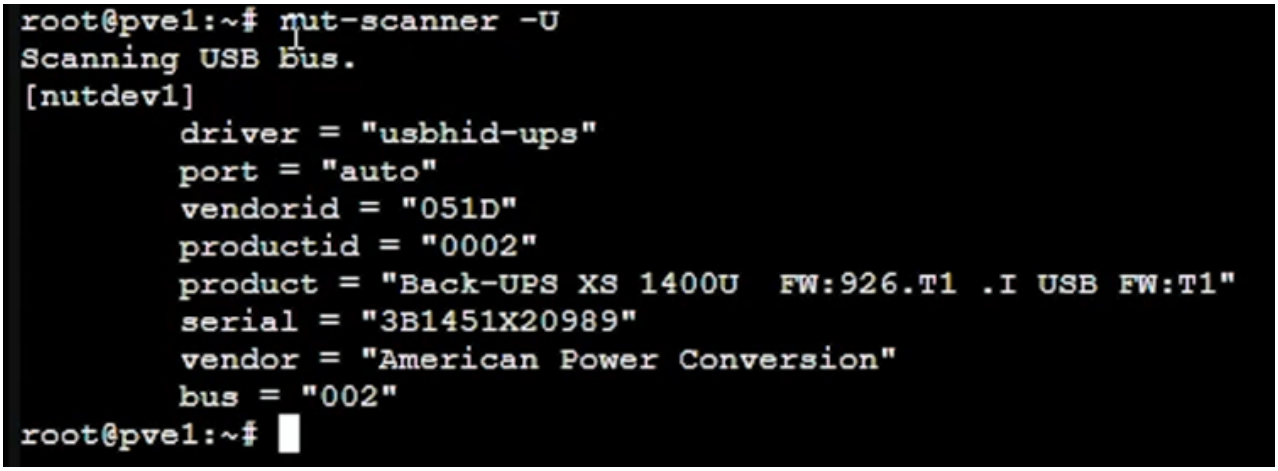
2. Install nut software.

3. Identify the Bus and Device of the UPS using the command: lsusb

```
root@pve1:~# lsusb
Bus 002 Device 005: ID 051d:0002 American Power Conversion Uninterruptible Power Supply
Bus 002 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 002 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
root@pve1:~#
```

4. (Optional) You can extract additional information from the device connected to the Bus / Device using the command lsusb -v -s <bus>:<device>.

5. Use the utility to get information you need for your nut driver configuration is nut-scanner -U.



Configuring NUT

1. Create a backup of this file:

> cp /etc/nut/ups.conf /etc/nut/ups.conf.original

2. Insert the text below.

> ***[pve1]***
> ***driver = “usbhid-ups”***
> ***port = “auto”***
> ***vendorid = “051D”***
> ***productid = “0002”***
> ***product = “Back-UPS XS 1400U FW:926.T1 .I USB FW:T1”***
> ***serial = “3B1451X20989”***
> ***vendor = “American Power Conversion”***
> ***bus = “001”***
>
> ***desc = “Proxmox-NUT UPS” # NUT-upsuser: mailing list ‘Re: [Nut-upsuser] UPS not Shutting Down’ Charles Lepple # Some drivers reset battery.charge.low to the default value. #lowbatt addresses this # See chapter 2.7 in https://rogerprice.org/NUT/ConfigExamples.A5.pdf override.battery.charge.low = 85 # lowbatt = 85 # if the remaining charge is less than 800s # trigger a low battery status override.battery.runtime.low = 800 # Wait 2 minutes between the shutdown command and # the moment the UPS shuts itself down offdelay = 120 # Wait 5 minutes before attempting to restart the ups # load after a power cut ondelay = 300 # wait 5 minutes for ups to power off override.ups.delay.shutdown = 300 # When you specify this, the driver ignores a low # battery condition flag that is # reported by the UPS ; https://networkupstools.org/docs/man/ups.conf.html ignorelb***

3. Secure the configuration file by running:

Each registered device in NUT must have a unique name. In the script above line 04 and 33 define two devices: **pve1** and **heartbeat**.

For physical devices such as pve1, the lines 06–13 is copied from the output provided by nut-scanner (see point 5). If nut-scanner returns serial = 0, ignore the line.

Some UPSs take override.battery.charge.low = 85 (line 21) while for other brands the lowbatt = 85 (line 22) must be used. This is because some UPS brands do not support, with some brands ignoring this parameter. You will need to discover what your UPS takes. Try this setup first.

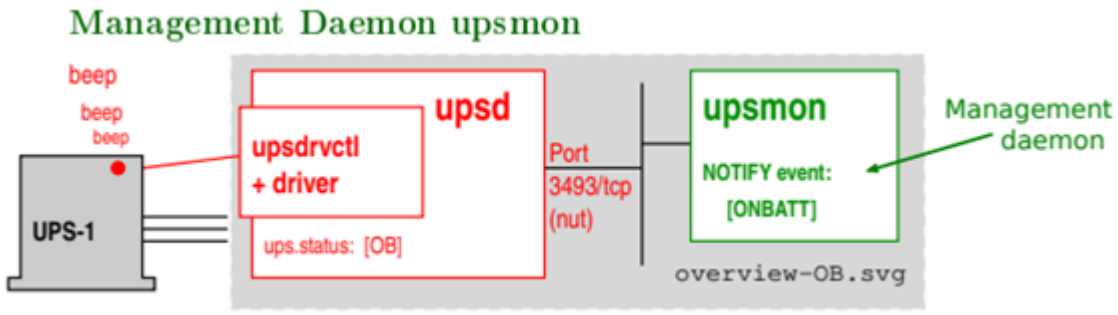
There is another UPS called **heartbeat**. This is a dummy ups that polls the communication between the computer and the UPS every 10 minutes. It is not evaluated in this HowTo.

4. Backup and replace the contents of the file */etc/nut/nut.conf* as shown below. In this HowTo, both **standalone** and **netserver** would have worked. standalone indicates that 1 UPS protecting the local system. netserver enables network connectivity and special care to protect from unauthorised access. If, for example, you add a NUT container on Proxmox to monitor the UPS (https://www.kreaweb.be/diy-home-server-2021-software-proxmox-ups/) this setting is necessary.

5. Secure access:

> chmod 600 /etc/nut/nut.conf

6. Backup and replace the contents of the */etc/nut/upsd.conf* as shown here.



7. Secure access:

> chmod 600 /etc/nut/upsd.conf

8. Backup and replace */etc/nut/upsd.users* as shown hereunder. This file defines the accounts that are authorised to read and adjust the UPS settings. In this example a single password was used everywhere. This should be avoided in production systems.

9. Secure access:

> chmod 600 /etc/nut/upsd.users

10. Create the file */etc/nut/heartbeat.conf*. The ups.status on the dummy UPS is switched every 5 minutes (TIMER setting). In another file this will be checked for.

11. Secure access:

> chmod 600 /etc/nut/heartbeat.conf

12. Start the driver controller to verify that all settings are correct. With the Proxmox-NUT VM, the bus the APC is connected on can change. The configuration file /etc/nut/ups.conf has to be modified to reflect the bus if this happens.

The next group of files declare how NUT will handle the various events.

13. Backup and replace: */etc/nut/upsmon.conf* with:

14. Secure access:

> chmod 600 /etc/nut/upsmon.conf

15. Backup and replace: */etc/nut/upssched.conf*. "*The NOTIFY events detected by upsmon and tagged as EXEC in upsmon.conf become events for upssched when NOTIFYCMD points to upssched. The program upssched provides a richer set of actions than upsmon*" [https://rogerprice.org/NUT/ConfigExamples.A5.pdf].

In rows 18–21, when the UPS first detects that it is OB it sets a 5-minute timer after which it shuts down the UPS. If UPS status changes to OL, the timer is cancelled. This is used to shut down the UPS after a fixed period of time. A LB condition will still cause a shutdown if it happens before the 5 minutes have passed.

16. Secure access:

> chmod 600 /etc/nut/upssched.conf

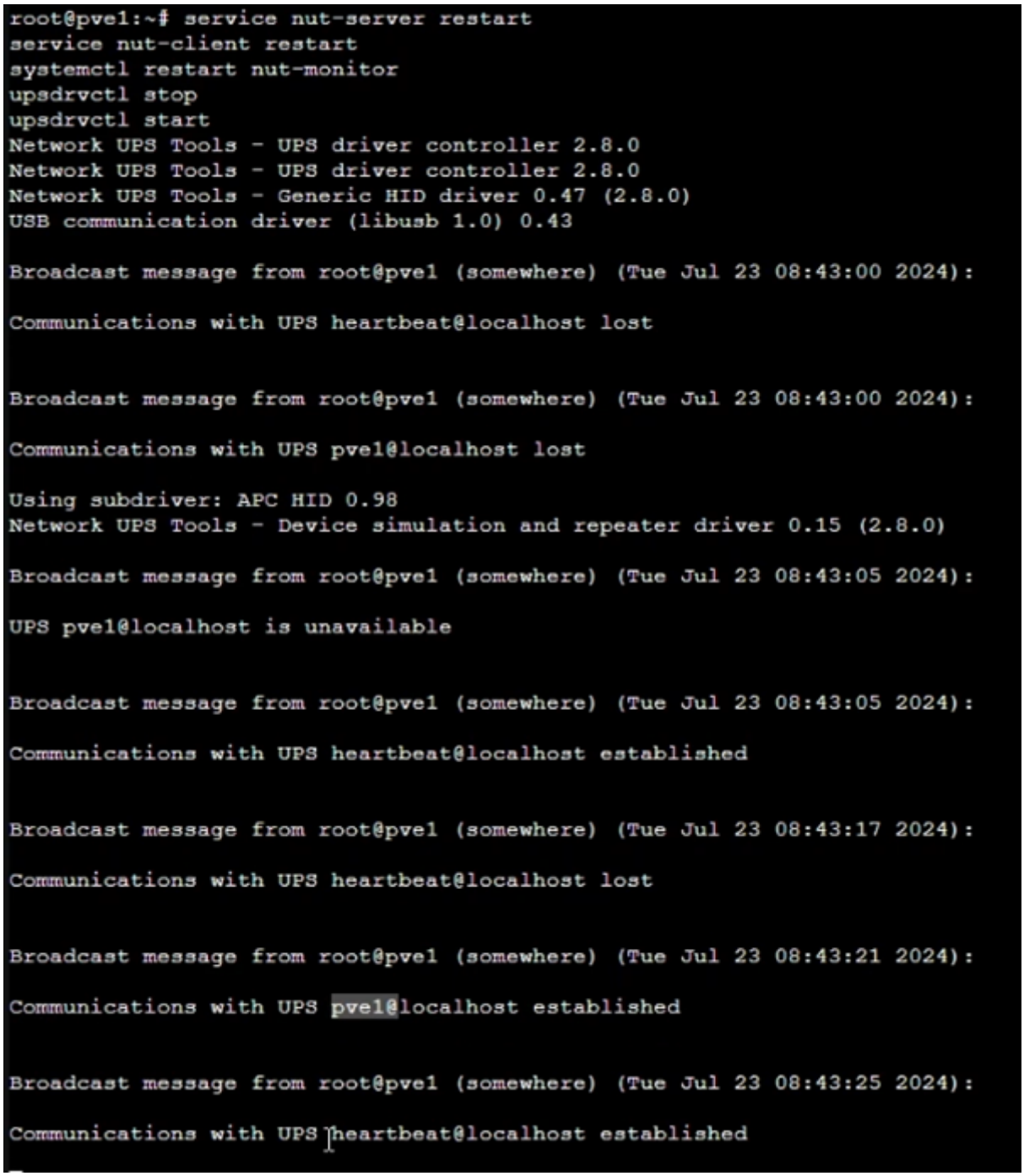
17. Backup and replace: */etc/nut/upssched-cmd*. This is a user-defined script defined in */etc/nut/upssched.conf* triggered by the **EXECUTE** You can change the name shown here (you need to change this file and the reference in upssched.conf) but, according to Price, "[while] *this is not the most elegant of names but if you use it, people in the NUT community will know immediately what you mean*".

Lines 18–23 is the block of code that handles a shutdown after 5 minutes not factoring a LB condition.

18. Grant root use read-write-execute, and group root read access.

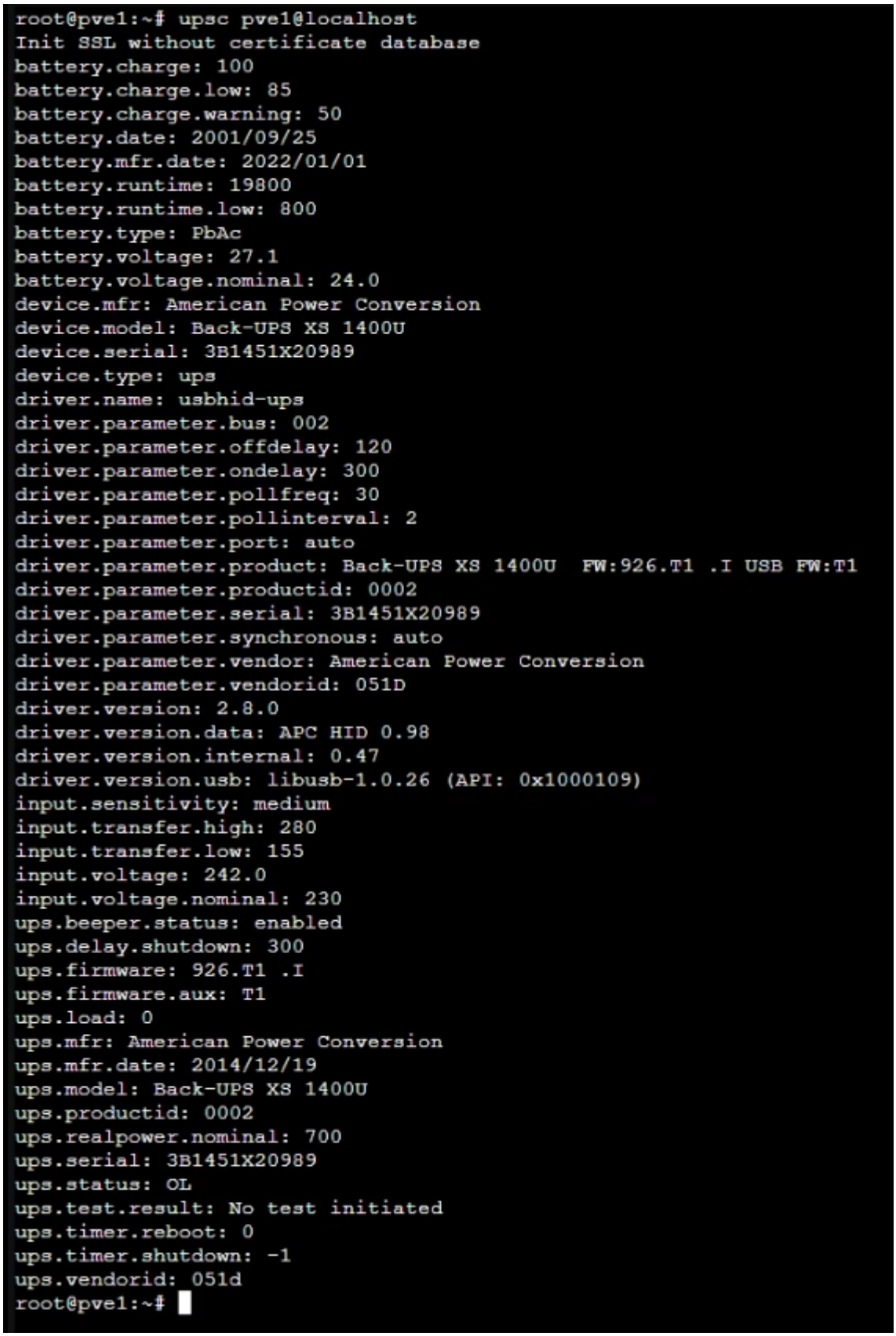
> chmod 740 /etc/nut/upssched-cmd

19. Start NUT solution. You should receive confirmation that both heartbeat and pve1 are connected.



Other NUT UPS commands

The following commands allow you to query your UPS and make adjustments. Not all UPSs consider adjusted parameters!!



```
root@pve1:~# upsc pve1@localhost
Init SSL without certificate database
battery.charge: 100
battery.charge.low: 85
battery.charge.warning: 50
battery.date: 2001/09/25
battery.mfr.date: 2022/01/01
battery.runtime: 19800
battery.runtime.low: 800
battery.type: PbAc
battery.voltage: 27.1
battery.voltage.nominal: 24.0
device.mfr: American Power Conversion
device.model: Back-UPS XS 1400U
device.serial: 3B1451X20989
device.type: ups
driver.name: usbhid-ups
driver.parameter.bus: 002
driver.parameter.offdelay: 120
driver.parameter.ondelay: 300
driver.parameter.pollfreq: 30
driver.parameter.pollinterval: 2
driver.parameter.port: auto
driver.parameter.product: Back-UPS XS 1400U  FW:926.T1 .I USB FW:T1
driver.parameter.productid: 0002
driver.parameter.serial: 3B1451X20989
driver.parameter.synchronous: auto
driver.parameter.vendor: American Power Conversion
driver.parameter.vendorid: 051D
driver.version: 2.8.0
driver.version.data: APC HID 0.98
driver.version.internal: 0.47
driver.version.usb: libusb-1.0.26 (API: 0x1000109)
input.sensitivity: medium
input.transfer.high: 280
input.transfer.low: 155
input.voltage: 242.0
input.voltage.nominal: 230
ups.beeper.status: enabled
ups.delay.shutdown: 300
ups.firmware: 926.T1 .I
ups.firmware.aux: T1
ups.load: 0
ups.mfr: American Power Conversion
ups.mfr.date: 2014/12/19
ups.model: Back-UPS XS 1400U
ups.productid: 0002
ups.realpower.nominal: 700
ups.serial: 3B1451X20989
ups.status: OL
ups.test.result: No test initiated
ups.timer.reboot: 0
ups.timer.shutdown: -1
ups.vendorid: 051d
root@pve1:~#
```

The query a single parameter, specify its name upsc pve1@localhost ups.status.

List the parameters that can be adjusted on this UPS:

Modify a parameter:

Testing the solution

Unplug the UPS (APC in this case) that is connected to the VMWare workstation. You can use **upsc** to check the variables.

NUT's Versatility and Flexibility

Besides this article and video series, if you want to configure NUT refer to some of the links shared below. Getting a basic understanding of how NUT works and how all the pieces interact with each other before you start slapping code into your solution will pay out almost immediately.

Video

Links

An Introduction to Network UPS Tools (Editor: Roger Price)
https://rogerprice.org/NUT/ConfigExamples.A5.pdf

Network UPS Tools
https://networkupstools.org/

NUT Users mailing list
http://lists.alioth.debian.org/mailman/listinfo/nut-upsuser

Network UPS Tools (NUT) Ultimate Guide
https://technotim.live/posts/NUT-server-guide/

DIY Home Server 2021 Software Proxmox UPS
https://www.kreaweb.be/diy-home-server-2021-software-proxmox-ups/