

Network UPS Tools (NUT) and NUT Monitor

Network UPS Tools (NUT) and NUT Monitor

For some strange reason, OpenIndiana doesn't ship with **Network UPS Tools (NUT)** as an available package. I found this blog from a person at Oracle. https://blogs.oracle.com/mikey/entry/network_ups_tools_on_solaris. I figured since he worked at Oracle, that he probably knew more about these things than I do, so I used this as a basis for how I installed NUT for the first time on an OpenIndiana machine.

I have made some corrections and adjustments to what is shown on the blog in order to make it work properly in OpenIndiana. I will make note of those changes below.

Also included in this page is how to get **NUT Monitor** to work in OpenIndiana, so you can see in real time what is happening with your UPS.

Installing and Configuring Network UPS Tools

1.) Download the latest stable version from <http://www.networkupstools.org/download.html>. For me, it was 2.6.5. If yours is different, you will need to adjust the paths below as necessary.

2.) Create the ups group and user NUT will use.

```
# groupadd -g 999 ups
# useradd -u 999 -g ups -d /var/ups -s /bin/false -m ups
# chown ups:ups /var/ups
# chmod 750 /var/ups
```

3.) Create the directory where you would like to install NUT. I made /usr/nut. It probably would have been better to put it in /usr/local/nut or /opt/nut (although that's where the Oracle person had the install go to), but this is where I put it.

Copy the downloaded file there. Verify the md5um (optional) and unpack it.

```
# mkdir /usr/nut
# cp nut-2.6.5.tar.gz /usr/nut/
# cd /usr/nut
# tar xzvpf nut-2.6.5.tar.gz
```

4.) In a terminal, "cd" into the directory you just unpacked. You should be in the directory that contains the configure file.

```
# cd /usr/nut/nut-2.6.5
```

5.) I had already downloaded gcc-3 for other things that I had compiled previously. I'm not sure the following command is necessary, but it certainly did not hurt. This makes sure the configure routine can find everything in the compiler.

```
# export PATH=/usr/gnu/bin/:$PATH
```

You can check it with this:

```
# echo $PATH
```

6.) Configure, compile and install with the following commands. Please note that the author of the blog had it install into /opt. This is all fine and well, but it left me with some stuff in /opt, and other important things that I still need in /usr. You might want to think about leaving the "--prefix=/opt/nut" out, or making it be a subdirectory of /usr/nut, or /usr/nut/nut-2.6.5.

```
# ./configure --prefix=/opt/nut --with-statepath=/var/ups --with-user=ups --with-group=ups
# make
# make install
```

7.) If all went well, you should now have an /opt/nut directory with, among other things, an etc directory with several sample files. The theory is that you copy those sample files to files with the same name, except without the .sample extension. Make changes to your "real" config files as necessary.

What I have found has worked best for me through the years with NUT is to leave everything commented out like they were in the sample files, and copy and uncomment what I want down at the bottom of each file. So this is what I did with everything except the upsmon.conf file, and the upssched.conf file, which already had only one line uncommented, and which I did not make any changes to.

The files below are for use with an APC SmartUPS, with the thought that I might want it to be a master controller. In initially writing this page, I noticed that I never created a nut.conf file, to set this as either a netserver or netclient mode. Like I said, I followed the expert's blog, and he didn't even mention this file. I would think that it shouldn't work at all without this, but it did for me. I have since edited this page, and included this file, after I set this up on a machine that would be acting as a netclient.

Also, system admin work is not my strong suit, so make adjustments to your files as necessary. You will at least need to change your IP Addresses, user names, passwords, etc. anyway.

[nut.conf](#)

[ups.conf](#)

[upspd.conf](#)

[upsd.users](#)

[upsmon.conf](#)

[upssched.conf](#)

8.) Now change permissions on the created config files so that only ups user is able to read them, as your password is stored there. If you don't do this, it won't work. I lost many hours trying to set NUT up the first time on Linux a few years ago because I neglected this one, very important step.

```
# cd /opt/nut/etc
# chgrp ups nut.conf ups.conf upsd.conf upsmon.conf upssched.conf
# chmod 640 nut.conf ups.conf upsd.conf upsmon.conf upssched.conf
```

9.) The Oracle expert made a small script to notify him via email whenever the UPS state changed. Since the server I installed this on is going to be placed in a remote location, I also wanted this functionality. So you may notice that in the upsmon.conf file, I had the NOTIFYFLAG lines set to SYSLOG+WALL+EXEC. Also, the NOTIFYCMD was set to his /opt/nut/bin/notify script.

Well, as it turns out, when I went to test the software, when I pulled the plug on the UPS, the wall command didn't work. Theoretically, I would think that the wall command to use should be in one of NUT's config files somewhere. When a precursory look turned up otherwise, I modified the notify script to do a wall command that works on OpenIndiana.

Technically speaking, since I'm using the notify script to wall, I don't really need the WALL in the NOTIFYFLAG lines. I left it there for clarity's sake, however.

I also modified his mail command to show the subject line. With his command, it was giving me a blank subject, even though he had one specified.

You will need to put a proper email address in this, or comment out that line, as it suits you. If you decide to have it email you, I also recommend putting your hostname in the email address subject part so you know which server is emailing you.

This is the modified notify script:

[notify](#)

Make it executable with this:

```
# chmod 755 /opt/nut/bin/notify
```

Also, please note that in the upsmon.conf file, I also changed the SHUTDOWNCMD. What was in there would not have worked in OpenIndiana. I have drained my UPS battery down to the point to check to see if the UPS would indeed shut down, and have verified that the command I changed it to works.

10.) Now for the service method and manifest files. If you want to follow the Oracle expert's way of doing this, you will need to make a directory for the method file. He puts the method file in opt, and the manifest file in var. I don't know why some pieces of software put their svc files in /var/svc and some in /lib/svc, but now that I am looking at this more closely, I would tend to think it might be better to put them both in /lib/svc. But I followed his blog on this first computer, so that's what I'll do here. If you decide to put them in /lib/svc, you will need to edit the files appropriately.

```
# mkdir -p /opt/nut/lib/svc/method
```

Here is the method file.

[nut](#)

Be sure to make it executable. The expert forgot to mention this. This should save you a little troubleshooting.

```
chmod +x /opt/nut/lib/svc/method/nut
```

Note: This method file has been changed from earlier versions of this wiki page.

I've modified this a little from what was on the Oracle blog. The Oracle expert had the stop method returning 1. This caused the service to go into maintenance mode for me when I stopped or restarted the service. Sorry, but I didn't notice that right away. So I changed it to return 0, and it seems to work properly now. Also, when I compiled this on a machine that did not have a UPS attached to it, it did not even put upsdrvctl in the /opt/nut/bin directory, so I changed the method file to check to see if the upsdrvctl file even existed before trying to shut it down.

The /var/svc/manifest/network directory already exists, so all you need to do is copy the below file into there.

[nut.xml](#)

11.) Enable the service:

```
# svccfg import /var/svc/manifest/network/nut.xml
# svcadm enable svc:/application/nut:default
# svcs -xv nut
```

12.) You can test the configuration with this. Adjust the ups name and IP Address as necessary.

```
# /opt/nut/bin/upsc smartups@192.168.0.3
```

This should give you a snapshot of the state of your UPS.

A little additional info here. If you are going to use more than one computer on a UPS, and have master and slave configurations, I believe you have to connect to them via the IP Address, not the host name of the machine. I lost some time on that one once, too.

NUT Monitor

If you would like to be able to check the state of your UPS visually real-time, NUT Monitor is a nice little GUI that will allow you to do this. It comes shipped with NUT, so you already have it. You just need to do a few things to get it to work.

1.) They put the backend module that NUT Monitor uses in a different directory than the application. So, the easiest way to deal with this is to copy it into the same directory as the application. I used Gnome's Nautilus to do this, but this should work from the command line:

```
# cp /usr/nut/nut-2.6.5/scripts/python/module/PyNUT.py /usr/nut/nut-2.6.5/scripts/python/app/PyNUT.py
```

2.) Make it executable:

```
# chmod +x /usr/nut/nut-2.6.5/scripts/python/app/PyNUT.py
```

3.) I edited the nut-monitor.desktop file to point the application and the png file to the correct location. The modified file is below. Copy it to /usr/share/applications

```
nut-monitor.desktop
```

4.) Change the permissions to match the others in the directory

```
# chown root:bin /usr/share/applications/nut-monitor.desktop  
# chmod 100644 /usr/share/applications/nut-monitor.desktop
```

5.) To launch it, go into Gnome's "Applications->Internet->NUT Monitor"

Hopefully it will launch!

6.) Type in the IP Address (or possibly name) of the machine that is monitoring the UPS. Press the "Refresh" button. Hopefully you will see your UPS. Then press the "Connect" button to see its state.

I didn't have to use authentication on mine, and I'm not sure why, as I specified the user name and password in my files. If this is an issue for you, you may want to look into that further.

Disclaimer: I didn't take this detailed set of notes when I set this up the first time, and then used those notes to write this page. I've built another server since then, and found a few errors in the earlier version of this page. I've gone through this step by step to verify everything, and made changes to this page as necessary. Hopefully I got everything.

I hope it helps!