Network UPS Tools (NUT)

Potřebné balíčky

- `nut` (debian) nebo `nut-server` a `nut-client` (ubuntu)
- `yum install nut nut-client` (centos) - *nutno zapnout repozitář EPEL*

Komponenty nutu

- **upsd** (nut-server.service)
  - komunikuje s jednotlivymy ups (drivery) a zprostredkuje komunikaci s nimi pro ostatní
  - v **upsd.conf** se nastavuje sitove spojeni a v **upsd.users** se nastavujou loginy pro pristup z ostatnich programu
  - v **nut.conf** se nastavuje jen ktere sluzby se maji spoustet, je mozne ze systemd to ignoruje
- **upsc, upscmd, upsrw**
  - klienti pro ovladani upsd z prikazoveho radku, umi v poradi jak jsou uvedeny: cist informace, spoustet prikazy a nastavovat parametry ups.
- **upsdrvctl** (nut-driver.service)
  - demony co se spousti po jednom procesu pro kazdou UPS, pro kazdy druh UPS je tenhle driver jinej. upsdrvctl je wrapper co spusti spravnou binarku pro kazdou UPS. upsd s nima pak pracuje.
  - seznam ups a driveru ktere maji pouzivat se nastavuje v **ups.conf**
- **upsmon** (nut-monitor.service)
  - pripojuje se k upsd, pravidelne zjistuje stav UPS a na zaklade toho rozesila warningy, vola skript **/etc/nut/notifycmd**, nebo shutdownuje server
  - seznam ups, ktery ma monitorovat a akci ktery ma provadet je nastavenej v **upsmon.conf**
- **upssched**
  - wrapper co lze pouzit na filtrovani eventu tak, ze se vola z upsmonu misto notifycmd, nicmene se v beznych usecasech nedoporucuje. konfiguruje se v **upssched.conf** a vetsina lidi ho pry vubec nepotrebuje.
- **nut-monitor** (neplest s upsmon, ktery pouziva systemd unitu nut-monitor.service!!!)
  - GUI aplikace s ikonkou na liste, ktera po pripojeni k upsd umoznuje pohodlny sledovani a spravu UPS
- **nut-cgi**
  - Zakladni webove rozhrani NUTu
  - Pripojuje se na upsd a UPS v nich zadane v **hosts.conf** (lze mit v jednom webu UPS z vice serveru)
  - Po zaheslovani webu a povoleni v **upsset.conf** lze i spoustet pres web akce a nastavovat UPS

Konfigurace

Základní konfigurace se děje pomocí souboru **/etc/nut/** (debian, ubuntu) nebo **/etc/ups/** (centos)

/etc/nut/nut.conf

```
#MODES: none/standalone/netserver/netclient
MODE=netserver
```

/etc/nut/ups.conf

```
[AEG]
        driver = blazer_ser
        port = /dev/ttyS0
        desc = "LocalUPS"
```

Pro usb:

/etc/nut/ups.conf

```
[AEG]
        driver = blazer_usb
        port = /dev/hidraw0
        desc = "LocalUPS"
```

- V souboru **ups.conf** definujeme jednotlivé UPS a jejich ovladače. Jaký ovladač je potřeba použít ke konkretní UPS se dozvíte v seznamu kompatibilního HW - http://www.networkupstools.org/stable-hcl.html
- Pokud používáme usb driver, je dobré před jakoukoliv další činností přehrát soubor **nut-usbups.rules** do */etc/udev/rules.d* . Tento soubor je vygenerován automaticky při instalaci balíčku a v různých systémech se může jmenovat trošku jinak:

Debian, Centos

```
ln -s /lib/udev/rules.d/62-nut-usbups.rules /etc/udev/rules.d/
```

Starej Debian, Ubuntu

```
ln -s /lib/udev/rules.d/52-nut-usbups.rules /etc/udev/rules.d/
```

Následně znovu načtěte pravidla udevu
`udevadm control --reload-rules`
`udevadm trigger`
Pokud to nepomůže, restartujte celý systém

Ještě ověříme práva a existenci slozek

```
mkdir /var/run/nut
chown root:nut /var/run/nut
chmod 770 /var/run/nut
```

Po restartu zkusíme, jestli máme nainstalován správný ovladač a jestli probíhá komunikace s UPS

korektně

```
upsdrvctl start
```

Pokud nám nenahlasí žádnou chybu, můžeme pokračovat dál:

/etc/nut/upsd.conf

```
#
==========================================================================
# LISTEN <address> [<port>]
LISTEN 127.0.0.1 3493
LISTEN 10.11.57.17 3493
# LISTEN ::1 3493
```

Nastavujeme, na kterých portech demon poslouchá. Pokud chceme monitorovat jen UPS, připojenou ke stejnému systému, tak nam staci nastavit jen localhost

/etc/nut/upsmon.conf

```
# Network UPS Tools: example upsmon configuration
#
# This file contains passwords, so keep it secure.

# ----------------------------------------------------------------------
-----
# RUN_AS_USER <userid>
#
# By default, upsmon splits into two processes.  One stays as root and
# waits to run the SHUTDOWNCMD.  The other one switches to another
userid
# and does everything else.
#
# The default nonprivileged user is set at compile-time with
#       'configure --with-user=...'.
#
# You can override it with '-u <user>' when starting upsmon, or just
# define it here for convenience.
#
# Note: if you plan to use the reload feature, this file (upsmon.conf)
# must be readable by this user!  Since it contains passwords, DO NOT
# make it world-readable.  Also, do not make it writable by the upsmon
# user, since it creates an opportunity for an attack by changing the
# SHUTDOWNCMD to something malicious.
#
# For best results, you should create a new normal user like "nutmon",
# and make it a member of a "nut" group or similar.  Then specify it
# here and grant read access to the upsmon.conf for that group.
#
# This user should not have write access to upsmon.conf.
```

```
#
# RUN_AS_USER nutmon

# ------------------------------------------------------------------------
-----
# MONITOR <system> <powervalue> <username> <password>
("master"|"slave")
#
# List systems you want to monitor.  Not all of these may supply power
# to the system running upsmon, but if you want to watch it, it has to
# be in this section.
#
# You must have at least one of these declared.
#
# <system> is a UPS identifier in the form
<upsname>@<hostname>[:<port>]
# like ups@localhost, su700@mybox, etc.
#
# Examples:
#
#  - "su700@mybox" means a UPS called "su700" on a system called
"mybox"
#
#  - "fenton@bigbox:5678" is a UPS called "fenton" on a system called
#    "bigbox" which runs upsd on port "5678".
#
# The UPS names like "su700" and "fenton" are set in your ups.conf
# in [brackets] which identify a section for a particular driver.
# If the ups.conf on host "doghouse" has a section called "snoopy", the
# identifier for it would be "snoopy@doghouse".
#
# <powervalue> is an integer - the number of power supplies that this
UPS
# feeds on this system.  Most computers only have one power supply, so
this
# is normally set to 1.  You need a pretty big or special box to have
any
# other value here.
#
# You can also set this to 0 for a system that doesn't supply any
power,
# but you still want to monitor.  Use this when you want to hear about
# changes for a given UPS without shutting down when it goes critical,
# unless <powervalue> is 0.
#
# <username> and <password> must match an entry in that system's
# upsd.users.  If your username is "monmaster" and your password is
# "blah", the upsd.users would look like this:
#
#       [monmaster]
#               password  = blah
```

```
#               upsmon master   (or slave)
#
# "master" means this system will shutdown last, allowing the slaves
# time to shutdown first.
#
# "slave" means this system shuts down immediately when power goes
critical.
#
# Examples:
#
# MONITOR myups@bigserver 1 monmaster blah master
# MONITOR su700@server.example.com 1 upsmon secretpass slave
# MONITOR myups@localhost 1 upsmon pass master  (or slave)
MONITOR AEG@localhost 0 master

# ---------------------------------------------------------------------
-----
# MINSUPPLIES <num>
#
# Give the number of power supplies that must be receiving power to
keep
# this system running.  Most systems have one power supply, so you
would
# put "1" in this field.
#
# Large/expensive server type systems usually have more, and can run
with
# a few missing.  The HP NetServer LH4 can run with 2 out of 4, for
example,
# so you'd set that to 2.  The idea is to keep the box running as long
# as possible, right?
#
# Obviously you have to put the redundant supplies on different UPS
circuits
# for this to make sense!  See big-servers.txt in the docs subdirectory
# for more information and ideas on how to use this feature.

MINSUPPLIES 0
# ---------------------------------------------------------------------
-----
# SHUTDOWNCMD "<command>"
#
# upsmon runs this command when the system needs to be brought down.
#
# This should work just about everywhere ... if it doesn't, well,
change it.

SHUTDOWNCMD "/sbin/shutdown -h +0"

# ---------------------------------------------------------------------
-----
```

```
# NOTIFYCMD <command>
#
# upsmon calls this to send messages when things happen
#
# This command is called with the full text of the message as one
argument.
# The environment string NOTIFYTYPE will contain the type string of
# whatever caused this event to happen.
#
# Note that this is only called for NOTIFY events that have EXEC set
with
# NOTIFYFLAG.  See NOTIFYFLAG below for more details.
#
# Making this some sort of shell script might not be a bad idea.  For
more
# information and ideas, see pager.txt in the docs directory.
#
# Example:
NOTIFYCMD /opt/sendnotify

# ----------------------------------------------------------------------
-----
# POLLFREQ <n>
#
# Polling frequency for normal activities, measured in seconds.
#
# Adjust this to keep upsmon from flooding your network, but don't make
# it too high or it may miss certain short-lived power events.

POLLFREQ 5

# ----------------------------------------------------------------------
-----
# POLLFREQALERT <n>
#
# Polling frequency in seconds while UPS on battery.
#
# You can make this number lower than POLLFREQ, which will make updates
# faster when any UPS is running on battery.  This is a good way to
tune
# network load if you have a lot of these things running.
#
# The default is 5 seconds for both this and POLLFREQ.

POLLFREQALERT 5
# ----------------------------------------------------------------------
-----
# HOSTSYNC - How long upsmon will wait before giving up on another
upsmon
#
# The master upsmon process uses this number when waiting for slaves to
```

```
# disconnect once it has set the forced shutdown (FSD) flag.  If they
# don't disconnect after this many seconds, it goes on without them.
#
# Similarly, upsmon slave processes wait up to this interval for the
# master upsmon to set FSD when a UPS they are monitoring goes critical
-
# that is, on battery and low battery.  If the master doesn't do its
job,
# the slaves will shut down anyway to avoid damage to the file systems.
#
# This "wait for FSD" is done to avoid races where the status changes
# to critical and back between polls by the master.

HOSTSYNC 15

# ----------------------------------------------------------------------
-----
# DEADTIME - Interval to wait before declaring a stale ups "dead"
#
# upsmon requires a UPS to provide status information every few seconds
# (see POLLFREQ and POLLFREQALERT) to keep things updated.  If the
status
# fetch fails, the UPS is marked stale.  If it stays stale for more
than
# DEADTIME seconds, the UPS is marked dead.
#
# A dead UPS that was last known to be on battery is assumed to have
gone
# to a low battery condition.  This may force a shutdown if it is
providing
# a critical amount of power to your system.
#
# Note: DEADTIME should be a multiple of POLLFREQ and POLLFREQALERT.
# Otherwise you'll have "dead" UPSes simply because upsmon isn't
polling
# them quickly enough.  Rule of thumb: take the larger of the two
# POLLFREQ values, and multiply by 3.

DEADTIME 15

# ----------------------------------------------------------------------
-----
# POWERDOWNFLAG - Flag file for forcing UPS shutdown on the master
system
#
# upsmon will create a file with this name in master mode when it's
time
# to shut down the load.  You should check for this file's existence in
# your shutdown scripts and run 'upsdrvctl shutdown' if it exists.
#
# See the shutdown.txt file in the docs subdirectory for more
```

```
information.

POWERDOWNFLAG /etc/killpower
# ------------------------------------------------------------------------
-----
# NOTIFYMSG - change messages sent by upsmon when certain events occur
#
# You can change the default messages to something else if you like.
#
# NOTIFYMSG <notify type> "message"
#
NOTIFYMSG ONLINE        "UPS %s on line power"
NOTIFYMSG ONBATT        "UPS %s on battery"
# NOTIFYMSG LOWBATT     "UPS %s battery is low"
# NOTIFYMSG FSD         "UPS %s: forced shutdown in progress"
# NOTIFYMSG COMMOK      "Communications with UPS %s established"
NOTIFYMSG COMMBAD       "Communications with UPS %s lost"
NOTIFYMSG SHUTDOWN      "Auto logout and shutdown proceeding"
NOTIFYMSG REPLBATT      "UPS %s battery needs to be replaced"
# NOTIFYMSG NOCOMM      "UPS %s is unavailable"
# NOTIFYMSG NOPARENT    "upsmon parent process died - shutdown
impossible"
#
# Note that %s is replaced with the identifier of the UPS in question.
#
# Possible values for <notify type>:
#
# ONLINE   : UPS is back online
# ONBATT   : UPS is on battery
# LOWBATT  : UPS has a low battery (if also on battery, it's
"critical")
# FSD      : UPS is being shutdown by the master (FSD = "Forced
Shutdown")
# COMMOK   : Communications established with the UPS
# COMMBAD  : Communications lost to the UPS
# SHUTDOWN : The system is being shutdown
# REPLBATT : The UPS battery is bad and needs to be replaced
# NOCOMM   : A UPS is unavailable (can't be contacted for monitoring)
# NOPARENT : The process that shuts down the system has died (shutdown
impossible)

# ------------------------------------------------------------------------
-----
# NOTIFYFLAG - change behavior of upsmon when NOTIFY events occur
#
# By default, upsmon sends walls (global messages to all logged in
users)
# and writes to the syslog when things happen.  You can change this.
#
# NOTIFYFLAG <notify type> <flag>[+<flag>][+<flag>] ...
#
```

```
NOTIFYFLAG ONLINE       SYSLOG+WALL+EXEC
NOTIFYFLAG ONBATT       SYSLOG+WALL+EXEC
# NOTIFYFLAG LOWBATT    SYSLOG+WALL
# NOTIFYFLAG FSD        SYSLOG+WALL
# NOTIFYFLAG COMMOK     SYSLOG+WALL
NOTIFYFLAG COMMBAD      SYSLOG+WALL+EXEC
NOTIFYFLAG SHUTDOWN     SYSLOG+WALL+EXEC
NOTIFYFLAG REPLBATT     SYSLOG+WALL+EXEC
# NOTIFYFLAG NOCOMM     SYSLOG+WALL
# NOTIFYFLAG NOPARENT   SYSLOG+WALL
#
#
# Possible values for the flags:
#
# SYSLOG - Write the message in the syslog
# WALL   - Write the message to all users on the system
# EXEC   - Execute NOTIFYCMD (see above) with the message
# IGNORE - Don't do anything
#
# If you use IGNORE, don't use any other flags on the same line.

# ---------------------------------------------------------------------
-----
# RBWARNTIME - replace battery warning time in seconds
#
# upsmon will normally warn you about a battery that needs to be
replaced
# every 43200 seconds, which is 12 hours.  It does this by triggering a
# NOTIFY_REPLBATT which is then handled by the usual notify structure
# you've defined above.
#
# If this number is not to your liking, override it here.

RBWARNTIME 43200

# ---------------------------------------------------------------------
-----
# NOCOMMWARNTIME - no communications warning time in seconds
#
# upsmon will let you know through the usual notify system if it can't
# talk to any of the UPS entries that are defined in this file.  It
will
# trigger a NOTIFY_NOCOMM by default every 300 seconds unless you
# change the interval with this directive.

NOCOMMWARNTIME 300

# ---------------------------------------------------------------------
-----
# FINALDELAY - last sleep interval before shutting down the system
#
```

```
# On a master, upsmon will wait this long after sending the
NOTIFY_SHUTDOWN
# before executing your SHUTDOWNCMD.  If you need to do something in
between
# those events, increase this number.  Remember, at this point your UPS
is
# almost depleted, so don't make this too high.
#
# Alternatively, you can set this very low so you don't wait around
when
# it's time to shut down.  Some UPSes don't give much warning for low
# battery and will require a value of 0 here for a safe shutdown.
#
# Note: If FINALDELAY on the slave is greater than HOSTSYNC on the
master,
# the master will give up waiting for the slave to disconnect.

FINALDELAY 5
```

Nastavíme ruzné notifikace apod.

🔧 Fix Me!

Nyní nastartujeme démona

- Debian, Ubuntu - `/etc/init.d/nut start` nebo `service nut start`
- Centos - `/etc/init.d/ups start` nebo `service ups start`

Následně vyzkoušíme načíst z UPS nějaká data. Použijeme na to příkaz `upsc UPS@server` kde UPS je jméno záložního zdroje v hranatých závorkách, uvedený v souboru ups.conf a server je adresa serveru, kde poslouchá démon upsd

```
root@dionysus:~# upsc AEG@localhost
battery.charge: 100
battery.voltage: 54.70
battery.voltage.high: 52.00
battery.voltage.low: 41.60
battery.voltage.nominal: 48.0
device.mfr:
device.model: LIHVX1K4
device.type: ups
driver.name: blazer_ser
driver.parameter.pollinterval: 2
driver.parameter.port: /dev/ttyS0
driver.version: 2.6.5
driver.version.internal: 1.55
input.current.nominal: 6.0
input.frequency: 50.0
input.frequency.nominal: 50
input.voltage: 236.9
```

```
input.voltage.fault: 236.8
input.voltage.nominal: 220
output.voltage: 236.8
ups.beeper.status: enabled
ups.delay.shutdown: 30
ups.delay.start: 180
ups.firmware: 00140.01
ups.load: 59
ups.mfr:
ups.model: LIHVX1K4
ups.status: OL
ups.temperature: 30.0
ups.type: offline / line interactive
```

Pokud je všechno správně, uvidíme podobný výstup, jako je tady na předchozím výpisu.

Přes systemd mi nestartovaly drivery po rebootu, tak jsem musel přidat pomlčku před cestu na tomhle řádku:

/lib/systemd/system/nut-driver.service

```
...
ExecStart=-/sbin/upsdrvctl start
...
```

Užitečné odkazy

- http://www.networkupstools.org/
- http://tedfelix.com/software/nut-network-ups-tools.html
- http://blog.shadypixel.com/monitoring-a-ups-with-nut-on-debian-or-ubuntu-linux/
- http://www.blasterspike.it/2011/03/28/how-to-install-nut-on-ubuntu-10-10-maverick-meerkat/

Monitoring

Nagios

Chceme-li UPS dohledovat pomocí nagiosu, použijeme na to tento plugin: check_nut.gz
Do nrpe.conf následně přidáme tento řádek
`command[check_nut]=/usr/lib/nagios/plugins/check_nut -d AEG -w 99 -c 50 -t 35 -T 40 -l 65 -L 80`

```
gandalf@dionysus:~$ /usr/lib/nagios/plugins/check_nut --help
check_nut v$Revision: 1.0 $ (nagios-plugins 1.4.16)
The nagios plugins come with ABSOLUTELY NO WARRANTY. You may redistribute
copies of the plugins under the terms of the GNU General Public License.
For more information about these matters, see the file named COPYING.
```

```
Copyright (c) 2007 Luca Bertoncello <lucabert@lucabert.de>

This plugin reports the status of the UPS using NUT

Usage:
check_nut -d <device> -w <battery charge warning> -c <battery charge
critical> -t <temperature warning> -T <temperature critical> -l <ups load
warning> -L <ups load critical>

Send email to nagios-users@lists.sourceforge.net if you have questions
regarding use of this software. To submit patches or suggest improvements,
send email to nagiosplug-devel@lists.sourceforge.net.
Please include version information with all correspondence (when possible,
use output from the --version option of the plugin itself).
```

```
gandalf@dionysus:~$ /usr/lib/nagios/plugins/check_nut -d AEG -w 99 -c 50 -t
35 -T 40 -l 65 -L 80
UPS AEG (LIHVX1K4) OK: battery charge: 100, temperature: 29.6, load: 59
```

Munin

Debiani balicek munin-plugins-extra

```
UPS=AEG
ln -s /usr/share/munin/plugins/nut_misc /etc/munin/plugins/
ln -s /usr/share/munin/plugins/nut_volts /etc/munin/plugins/
ln -s /usr/share/munin/plugins/nutups_
/etc/munin/plugins/nutups_"$UPS"_voltages
ln -s /usr/share/munin/plugins/nutups_ /etc/munin/plugins/nutups_"$UPS"_freq
ln -s /usr/share/munin/plugins/nutups_
/etc/munin/plugins/nutups_"$UPS"_charge
ln -s /usr/share/munin/plugins/nutups_
/etc/munin/plugins/nutups_"$UPS"_current
#zatim neni v debianu
#ln -s /usr/share/munin/plugins/nutups2_
/etc/munin/plugins/nutups2_"$UPS"_charge
#ln -s /usr/share/munin/plugins/nutups2_
/etc/munin/plugins/nutups2_"$UPS"_current
#ln -s /usr/share/munin/plugins/nutups2_
/etc/munin/plugins/nutups2_"$UPS"_frequency
#ln -s /usr/share/munin/plugins/nutups2_
/etc/munin/plugins/nutups2_"$UPS"_voltage
```

/etc/munin/plugin-conf.d/nut

```
[nut*]
env.upsname AEG@127.0.0.1
```

Nedistribucni (asi zbytecny)

https://github.com/munin-monitoring/contrib/tree/master/plugins/ups

```
curl 'https://raw.githubusercontent.com/munin-monitoring/contrib/master/plugins/ups/nut' > /usr/share/munin/plugins/nut_
chmod +x /usr/share/munin/plugins/nut_
ln -s /usr/share/munin/plugins/nut_ nut_AEG_AT_localhost
```

From:
https://wiki.spoje.net/ - **SPOJE.NET**

Permanent link:
https://wiki.spoje.net/doku.php/howto/network/nut

Last update: **2017/09/08 23:50**