

PNP4Nagios en Naemon (Thruk). Instalación.

Introducción

En artículos previos vimos como instalar PNP4Nagios con Nagios Core 4 en [CentOS / Redhat](#) y en [Debian 7](#). Con la reciente puesta en escena de Naemon (aún en fase beta) del que vimos su instalación en el artículo "[Naemon. Introducción e instalación](#)" y los [cambios](#) que parece va a suponer en la comunidad Nagios no podíamos quedarnos si probar esta herramienta clásica en Naemon. En principio no debería haber problema para que funcione también con este (como así es) ya que al menos de momento es prácticamente idéntico a Nagios Core 4.

Hay que tener en cuenta que al igual que con Nagios Core 4 no podremos usar el broker "npcdmod" así que NO podemos configurar PNP4Nagios en modo "Bulk Mode with npcmod". Como comentaba, la API "event broker" de Nagios Core 4 ha cambiado con respecto a la versión previa, con lo cual el broker npcmod no va a funcionar con este de momento (hablamos de versión pnp4nagios 0.6.21) hasta que no desarrollen un nuevo broker.

A su vez todos los paquetes de las distribuciones habituales están vinculados al paquete de Nagios Core ya que "necesitaban" lógicamente este instalado para tener sentido por lo que si instalamos PNP4Nagios mediante paquetes las dependencias instalarán también Nagios Core (a no ser que lo evitemos o que "desactivemos" Nagios posteriormente).

Instalación.

En nuestro caso partimos de una instalación base de [Naemon en CentOS](#) ya realizada. La instalación de PNP4Nagios la realizaremos compilando el software en lugar de mediante paquetes de la distribución, por las pegas ya mencionadas.

Instalamos primero algunos **prerequisitos**:

```
yum install rrdtool perl-Time-HiRes rrdtool-perl php-gd php
```

Instalaremos las herramientas de desarrollo solo si no las tenemos o nos falta alguna (al compilar pnp4nagios nos dé un error de compilador). Si no quieres instalar todo el grupo puedes ir probando para ver las que necesitas.

```
yum groupinstall "Development Tools"
```

Bajamos [pnp4Nagios](#), descomprimos, compilamos e instalamos. Le diremos que use el usuario y grupo naemon.

```
./configure --with-nagios-user=naemon --with-nagios-group=naemon  
make all
```

```
make fullinstall
```

Nos ofrece el siguiente resumen de configuración

```
*** Configuration summary for pnp4nagios-0.6.21 03-24-2013 ***
```

General Options:

```
-----
Nagios user/group:      naemon naemon
Install directory:     /usr/local/pnp4nagios
HTML Dir:              /usr/local/pnp4nagios/share
Config Dir:            /usr/local/pnp4nagios/etc
Location of rrdtool binary: /usr/bin/rrdtool Version 1.3.8
RRDs Perl Modules:     FOUND (Version 1.3008)
RRD Files stored in:   /usr/local/pnp4nagios/var/perfdata
process_perfdata.pl Logfile: /usr/local/pnp4nagios/var/perfdata.log
Perfdata files (NPCD) stored in: /usr/local/pnp4nagios/var/spool
```

Web Interface Options:

```
-----
HTML URL:              http://localhost/pnp4nagios
Apache Config File:    /etc/httpd/conf.d/pnp4nagios.conf
```

Configuramos el demonio npcd de pnp4nagios para que se inicie automáticamente y lo arrancamos:

```
chkconfig npcd on && service npcd start
```

Editamos el fichero `/etc/httpd/conf.d/pnp4nagios.conf` y apuntamos a nuestro fichero de passwords de Naemon (thruk)

```
# AuthUserFile /usr/local/nagios/etc/htpasswd.users
AuthUserFile /etc/naemon/htpasswd
```

Recargamos apache

```
service httpd reload
```

Si accedemos ahora al interface de pnp4nagios (thrukadmin / thrukadmin, por defecto):

```
http://nuestra_ip/pnp4nagios/
```

deberíamos ver todos los chequeos que realiza pnp4Nagios en verde. Procedemos como nos indica a renombrar el fichero `/usr/local/pnp4nagios/share/install.php` para que se dé por enterado de que está correctamente instalado.

Volvemos a cargar la página y veremos que ahora nos da un error:

Please check the documentation for information about the following error. perfdata directory "/usr/local/pnp4nagios/var/perfdata/" is empty. Please check your Nagios config. Read FAQ online

De momento es normal, nos faltan configurar la integración con Naemon.

Configuración.

PNP4Nagios tiene varias formas de configurar la integración con Nagios. Usaremos el método bulk-mode. Si estás pensando en usar el más sencillo e igual de eficaz “Bulk Mode with npcdmod” olvídalos de momento, como comentaba en la introducción no va a funcionar.

En el directorio `/usr/local/pnp4nagios/etc` tenemos dos ficheros de ejemplo para hacer copy/paste.

Del fichero `nagios.cfg-sample` copiamos **la siguiente** configuración al fichero `/etc/nagios/naemon.cfg` (podemos copiarsela al final del fichero):

```
#
# Bulk / NPCD mode
#
process_performance_data=1
# *** the template definition differs from the one in the original nagios.cfg
#service_perfddata_file=/usr/local/pnp4nagios/var/service-perfddata
service_perfddata_file_template=DATATYPE::SERVICEPERFDATA\tTIMET::$TIMET$\tHOSTNAME::
$HOSTNAMES$\tSERVICEDESC::$SERVICEDESC$\tSERVICEPERFDATA::
$SERVICEPERFDATA$\tSERVICECHECKCOMMAND::$SERVICECHECKCOMMAND$\tHOSTSTATE::
$HOSTSTATES$\tHOSTSTATETYPE::$HOSTSTATETYPE$\tSERVICESTATE::
$SERVICESTATE$\tSERVICESTATETYPE::$SERVICESTATETYPE$
service_perfddata_file_mode=a
service_perfddata_file_processing_interval=15
service_perfddata_file_processing_command=process-service-perfddata-file

# *** the template definition differs from the one in the original nagios.cfg
#
host_perfddata_file=/usr/local/pnp4nagios/var/host-perfddata
host_perfddata_file_template=DATATYPE::HOSTPERFDATA\tTIMET::$TIMET$\tHOSTNAME::
$HOSTNAMES$\tHOSTPERFDATA::$HOSTPERFDATA$\tHOSTCHECKCOMMAND::
$HOSTCHECKCOMMAND$\tHOSTSTATE::$HOSTSTATES$\tHOSTSTATETYPE::$HOSTSTATETYPE$
host_perfddata_file_mode=a
host_perfddata_file_processing_interval=15
host_perfddata_file_processing_command=process-host-perfddata-file
```

Del fichero `misccommands.cfg-sample` copiamos **la siguiente** configuración a `/etc/naemon/conf.d/commands.cfg`:

```
#
# Bulk with NPCD mode
#
define command {
command_name process-service-perfddata-file
```

```

command_line /bin/mv /usr/local/pnp4nagios/var/service-perfdata /usr/local/pnp4nagios/var/spool/service-perfdata.
$TIMET$
}
define command {
command_name process-host-perfdata-file
command_line /bin/mv /usr/local/pnp4nagios/var/host-perfdata /usr/local/pnp4nagios/var/spool/host-perfdata.
$TIMET$
}

```

Reiniciamos el demonio npcd de pnp4nagios y el propio Naemon.

```
service npcd restart && service naemon restart
```

Y ya podemos conectarnos a la URL de PNP4Nagios. En cuanto empiece a crear ficheros de gráficas desaparecerá el error y veremos al rato que empieza a pintar nuestras gráficas.

Nos faltaría ahora una cosa importante. Poder acceder directamente desde el Host / Servicio en el interface de Naemon (Thruk) a las gráficas correspondientes de este sin tener que ir a el interface de PNP4 para localizarlo. Creamos unos templates para aplicarlos luego a los objetos.

En el fichero (p.e.) /etc/naemon/conf.d/templates/hosts.cfg (p.e.) añadimos:

```

define host {
name host-pnp
action_url /pnp4nagios/index.php/graph?host=$HOSTNAME&srv=_HOST_' class='tips'
rel='/pnp4nagios/index.php/popup?host=$HOSTNAME&srv=_HOST_'
register 0
}

```

En el fichero (p.e.) /etc/naemon/conf.d/templates/services.cfg (p.e.) añadimos:

```

define service {
name srv-pnp
action_url /pnp4nagios/index.php/graph?host=$HOSTNAME&srv=$SERVICEDESCS' class='tips'
rel='/pnp4nagios/index.php/popup?host=$HOSTNAME&srv=$SERVICEDESC$
register 0
}

```

Y modificamos nuestros objetos Host / Servicios para que hereden dicha plantilla. P.e.

```

define host{
use linux-server,host-pnp
host_name localhost
alias localhost
address 127.0.0.1
}
define service{
use local-service,srv-pnp

```

```
host_name localhost
service_description PING
check_command check_ping!100.0,20%!500.0,60%
}
```

Si no controlas bien el tema de los ficheros de configuración de Nagios es que no leíste el artículo de [“Ficheros de configuración en Nagios”](#) . Si no te aclaras con los objetos tienes también un [estupendo artículo](#) al respecto.

Podemos añadirle la plantilla a todos aquellos host / servicios que queramos tengan gráfica. Las estrategias pueden ser diferentes; uno a uno, definición de servicios asignadas a grupos de host en lugar de a hosts (lo normal), asignar la plantilla a la plantillas a otra plantilla (local-service p.e.),

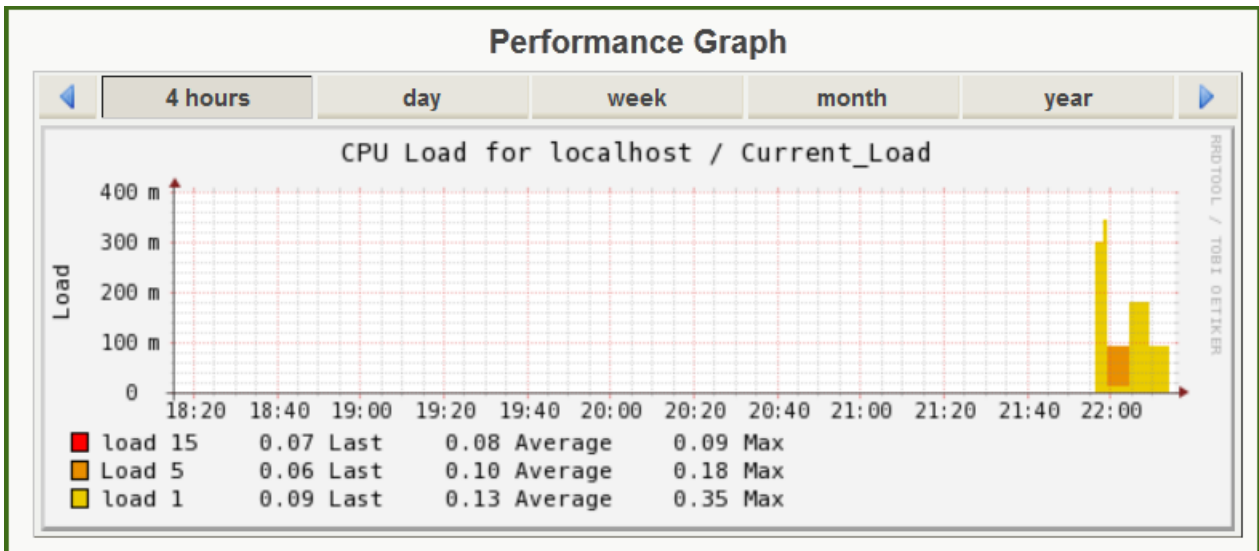
Reiniciamos naemon y ncmd (módulo de pnp):

```
service npcd restart && service naemon restart
```

Veremos que nos han aparecido los iconos para acceder pinchando directamente a las gráficas del Host / Servicio directamente desde este.

localhost	Current Load		OK
	Current Users		OK
	HTTP		WARNING
	PING		OK
	Root Partition		OK
	SSH		OK
	Swap Usage		OK
	Total Processes		OK

Además Thruk integra el gráfico de pnp4nagios en la pantalla de detalles del servicio:



Si quieres saber más de PNP4Nagios échale un ojo al artículo de [Características de PNP4Nagios](#).

Otra aplicación imprescindible para nuestro entorno de Monitorización funcionando para Naemon. Habrá que esperar probablemente bastante tiempo para que se vaya normalizando el tema de los paquetes en las distribuciones. ¿Tendremos en un futuro un paquete PNP4Nagios para Nagios Core y otro para Naemon? ¿Cambiará PNP4Nagios su nombre a PNP4Monitoring :-D?