**CYBER FORZA**

A reference document used to install pnp4nagios on Unbuntu

Revision: Draft specification 1.0

July 2017

**Cyber Forza Inc. Confidential and Proprietary**

Copyright © 2017, Cyber Forza Incorporation. All rights reserved.

This document contains information that is considered proprietary and the property of Cyber Forza Inc. The content of this document may not be used in any way, duplicated, or distributed by any means without the express written permission of Cyber Forza Inc

Unless otherwise agreed in writing, by Cyber Forza Inc., The Cyber Forza products are not designed or intended for any application in which the failure of the product could create a situation where personal injury or death may occur.

Cyber Forza Inc. may change specifications and product description at any time, without notice.

Contact Cyber Forza Inc. to obtain latest specification.

## **Contents**

1 [Revision History](#_Revision_History_1) 5

[2 Quick Start Guide](#_2_Quick_Start) 6

[2.1 System Specification](#_2.1_System_Specification) 6

[2.2 Support Information](#_2.2_Support_Information) 6

[2.3 Getting Started](#_2.3_Getting_Started) 6

[2.4 Prerequisite and compile process](#_2.4_Prerequisite_and) 7

[3 Modes to process the performance data](#_3__Modes) 10

[3.1 Configuration of pnp4nagios apache](#_3.1__) 12

[4 Nagios configuration to interact with pnp4nagios and allow visualization of the graphs](#_4_Nagios_configuration) 14

# Revision History

|  |  |  |
| --- | --- | --- |
| Date | Revision No. | Description |
| 12/07/17 | Draft Release 1.0 | Initial version |
|  |  |  |

# 2 Quick Start Guide

# 2.1 System Specification

*Table 1: System Configuration*

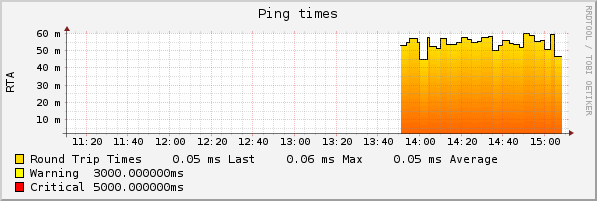
|  |
| --- |
| * Ubuntu 16.04.1 LTS (Linux Kernel 4.4) |
| * System RAM: 3911776 KB |
| * Hard Disk space: 21.1 GB |
| * OS type: 64-bit |
| * Memory: 3.7 GiB |
| * Processor: Intel Core 2 Duo CPU E8400 @ 3.00GHz x 2 |

# 2.2 Support Information

* Please contact the Cyber Forza Inc. team for support

# 2.3 Getting Started

PNP is an addon for the Nagios Network Monitoring System. pnp4nagios is a framework written in perl, PHP and C for automatically parsing performance data collected by Nagios plugins. The data is collected into RRD databases for display in the Nagios web interface. The graphs created by pnp4nagios are similar to other monitoring tools.

 Figure1 Example of graph created by pnp4nagios

# 

# 2.4 Prerequisite and compile process

We should have installed:

Nagios ( See the Guide to How To Install Nagios on Ubuntu )

The rrdtool (*rrdtool perl-rrdtool* and php\_gd) packages:

If are not installed, please execute:

**apt-get install** **perl** rrdtool librrds-perl php7-gd

We need to download the pnp4nagios.In this case we download the sources of the pnp4nagios-0.6.25 release

*wget* [*https://sourceforge.net/projects/pnp4nagios/files/PNP-0.6/pnp4nagios-*](https://sourceforge.net/projects/pnp4nagios/files/PNP-0.6/pnp4nagios-)*0.6.25.tar.gz/download*

*tar -zxvf pnp4nagios-0.6.25.tar.gz*

***NOTE\*\* If you install from the package cf.monitoring.utilities.1-0.deb then go to /usr/cf.monitoring and follow the instructions:***

*cd pnp4nagios-0.6.25*

*./configure -prefix=/usr/local/pnp4nagios -with-rrdtool=/usr/bin/rrdtool -with-nagios-user=nagios -with-nagios-group=nagios*

This will generate the entire compilation process which will eventually yield something similar to the following:

*\*\*\* Configuration summary for pnp4nagios-0.6.25 03-01-2015 \*\*\**

*General Options:*

*------------------------- -------------------*

*Nagios user/group: nagios nagios*

*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 1.5.5*

*RRDs Perl Modules: FOUND (Version 1.5001)*

*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*

*Tuve que anadir la configuraion de pnp4nagios en el archivo probicilap de apache2*

Review the options above for accuracy. If they look okay,

type :

*make all && make install && make install-webconf && make install-config*

*&& make install-init*

We make sure that we do not have any type of error in the compilation process

*cd /usr/local/pnp4nagios/etc*

*cp misccommands.cfg-sample misccommands.cfg*

*cp nagios.cfg-sample nagios.cfg*

*cp rra.cfg-sample rra.cfg*

*cd /usr/local/pnp4nagios/etc/pages/*

*cp web\_traffic.cfg-sample web\_traffic.cfg*

*cd ../check\_commands*

*cp check\_all\_local\_disks.cfg-sample*

*check\_all\_local\_disks.cfg*

*cp check\_nrpe.cfg-sample check\_nrpe.cfg*

*cp check\_nwstat.cfg-sample check\_nwstat.cfg*

# 3 Modes to process the performance data

Before configuring pnp, we need to decide how we want Nagios to process the performance data. This largely depends on the number of monitored hosts and services of Nagios.

Default mode , where process\_perfdata.pl is executed after each host and service check, is acceptable for small installations.

Bulk mode, where performance information is appended to a temporary file and processed after a short interval, is fine for medium-sized installations.

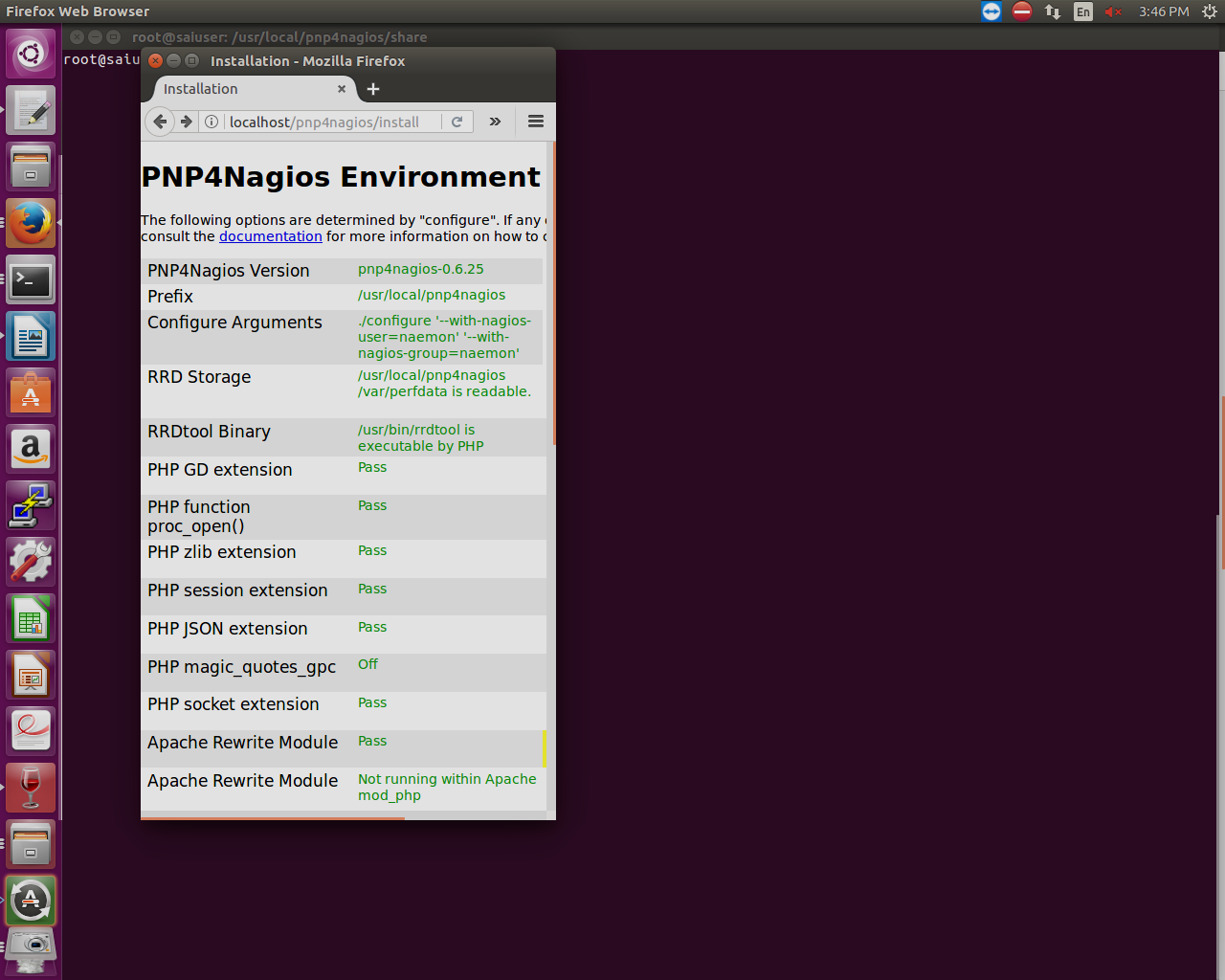
Setups with hundreds of hosts and services should use bulk mode with npcd , where a separate multi-threaded daemon handles the processing. We will use the Bulk Mode with npcd of pnp4nagios

We should execute the following commands to add npcd as service :

*update-rc.d npcd defaults*

*service npcd start*

We should see something similar to this:

Figure2: PNP4Nagios Environment detected

We can delete then the test page

*rm -rf /usr/local/pnp4nagios/share/install.php*

# 3.1 Configuration of pnp4nagios apache

As we noted previously in the compilation process:

*Apache Config File: /etc/httpd/conf.d/pnp4nagios.conf*

We will need modify this configuration file the configuration of pnp4nagios apache, where we have to instantiate our password file of nagios access , modifying the value:

Here a configuration file for reference:

/etc/httpd/conf.d/pnp4nagios.conf

*# SAMPLE CONFIG SNIPPETS FOR APACHE WEB SERVER  
  
Alias /pnp4nagios "/usr/local/pnp4nagios/share"  
  
<Directory "/usr/local/pnp4nagios/share">  
        AllowOverride None  
        Order allow,deny  
        Allow from all  
        #  
        # Use the same value as defined in nagios.conf  
        #  
        AuthName "Nagios Access"  
        AuthType Basic  
        AuthUserFile /usr/local/nagios/etc/htpasswd.users  
        Require valid-user  
        <IfModule mod\_rewrite.c>  
                # Turn on URL rewriting  
                RewriteEngine On  
                Options symLinksIfOwnerMatch  
                # Installation directory  
                RewriteBase /pnp4nagios/  
                # Protect application and system files from being viewed  
                RewriteRule "^(?:application|modules|system)/" - [F]  
                # Allow any files or directories that exist to be displayed directly  
                RewriteCond "%{REQUEST\_FILENAME}" !-f  
                RewriteCond "%{REQUEST\_FILENAME}" !-d  
                # Rewrite all other URLs to index.php/URL  
                RewriteRule "^.\*$" "index.php/$0" [PT]  
        </IfModule>  
</Directory>  
  
  
ScriptAlias /nagios/cgi-bin /usr/local/nagios/sbin  
<Directory "/usr/local/nagios/sbin">  
AllowOverride AuthConfig  
Options ExecCGI  
Order allow,deny  
Allow from all*

*</Directory>*

Restart apache service

*service apache2 restart*

# 4 Nagios configuration to interact with pnp4nagios and allow visualization of the graphs

Now we have to configure nagios and that can interact and allow visualization of the graph. As we established before, for this we will use the Bulk Mode with npcd of pnp4nagios, for which we edit the configuration file nagios /etc/nagios/nagios.cfg, adding / modifying the following values

Edit the performance data section of */usr/local/nagios/etc/nagios.cfg*, making the changes noted in Bold:

# PROCESS PERFORMANCE DATA OPTION

# This determines whether or not Nagios will process performance

# data returned from service and host checks. If this option is

# enabled, host performance data will be processed using the

# host\_perfdata\_command (defined below) and service performance

# data will be processed using the service\_perfdata\_command (also

# defined below). Read the HTML docs for more information on

# performance data.

# Values: 1 = process performance data, 0 = do not process performance data

**process\_performance\_data=1**

# HOST AND SERVICE PERFORMANCE DATA PROCESSING COMMANDS

# These commands are run after every host and service check is

# performed. These commands are executed only if the

# enable\_performance\_data option (above) is set to 1. The command

# argument is the short name of a command definition that you

# define in your host configuration file. Read the HTML docs for

# more information on performance data.

#host\_perfdata\_command=process-host-perfdata

#service\_perfdata\_command=process-service-perfdata

# HOST AND SERVICE PERFORMANCE DATA FILES

# These files are used to store host and service performance data.

# Performance data is only written to these files if the

# enable\_performance\_data option (above) is set to 1.

**host\_perfdata\_file=/usr/local/pnp4nagios/var/host-perfdata**

**service\_perfdata\_file=/usr/local/pnp4nagios/var/service-perfdata**

# HOST AND SERVICE PERFORMANCE DATA FILE TEMPLATES

# These options determine what data is written (and how) to the

# performance data files. The templates may contain macros, special

# characters (\t for tab, \r for carriage return, \n for newline)

# and plain text. A newline is automatically added after each write

# to the performance data file. Some examples of what you can do are

# shown below.

**host\_perfdata\_file\_template=DATATYPE::HOSTPERFDATA\tTIMET::$TIMET$\tHOSTNAME::$HOSTNAME$\tHOSTPERFDATA::$HOSTPERFDATA$\tHOSTCHECKCOMMAND::$HOSTCHECKCOMMAND$\tHOSTSTATE::$HOSTSTATE$\tHOSTSTATETYPE::$HOSTSTATETYPE$\tHOSTOUTPUT::$HOSTOUTPUT$**

**service\_perfdata\_file\_template=DATATYPE::SERVICEPERFDATA\tTIMET::$TIMET$\tHOSTNAME::$HOSTNAME$\tSERVICEDESC::$SERVICEDESC$\tSERVICEPERFDATA::$SERVICEPERFDATA$\tSERVICECHECKCOMMAND::$SERVICECHECKCOMMAND$\tHOSTSTATE::$HOSTSTATE$\tHOSTSTATETYPE::$HOSTSTATETYPE$\tSERVICESTATE::$SERVICESTATE$\tSERVICESTATETYPE::$SERVICESTATETYPE$\tSERVICEOUTPUT::$SERVICEOUTPUT$**

# HOST AND SERVICE PERFORMANCE DATA FILE MODES

# This option determines whether or not the host and service

# performance data files are opened in write ("w") or append ("a")

# mode. If you want to use named pipes, you should use the special

# pipe ("p") mode which avoid blocking at startup, otherwise you will

# likely want the defult append ("a") mode.

**host\_perfdata\_file\_mode=a**

**service\_perfdata\_file\_mode=a**

# HOST AND SERVICE PERFORMANCE DATA FILE PROCESSING INTERVAL

# These options determine how often (in seconds) the host and service

# performance data files are processed using the commands defined

# below. A value of 0 indicates the files should not be periodically

# processed.

**host\_perfdata\_file\_processing\_interval=15**

**service\_perfdata\_file\_processing\_interval=15**

# HOST AND SERVICE PERFORMANCE DATA FILE PROCESSING COMMANDS

# These commands are used to periodically process the host and

# service performance data files. The interval at which the

# processing occurs is determined by the options above.

**host\_perfdata\_file\_processing\_command=process-host-perfdata-file**

**service\_perfdata\_file\_processing\_command=process-service-perfdata-file**

At the end of */usr/local/nagios/etc/objects/commands.cfg*, add the command definitions:

*define command {*

*command\_name process-service-perfdata-file*

*command\_line /usr/local/pnp4nagios/libexec/process\_perfdata.pl --bulk=/usr/local/pnp4nagios/var/service-perfdata*

*}*

*define command {*

*command\_name process-host-perfdata-file*

*command\_line /usr/local/pnp4nagios/libexec/process\_perfdata.pl --bulk=/usr/local/pnp4nagios/var/host-perfdata*

*}*

There is one other thing to check before running pnp4nagios for the first time. If you don’t want the default resolution for your performance data (1 minute intervals for 2 days up to 6 hour intervals for 4 years), you must create *a /usr/local/pnp4nagios/etc/rra.cfg* file with your preferences. You can use the snippet below as a template.

*# 2880 entries with 1 minute step = 48 hours*

*RRA:AVERAGE:0.5:1:2880*

*# 2880 entries with 5 minute step = 10 days*

*RRA:AVERAGE:0.5:5:2880*

*# 4320 entries with 30 minute step = 90 days*

*RRA:AVERAGE:0.5:30:4320*

*# 5840 entries with 360 minute step = 4 years*

*RRA:AVERAGE:0.5:360:5840*

*RRA:MAX:0.5:1:2880*

*RRA:MAX:0.5:5:2880*

*RRA:MAX:0.5:30:4320*

*RRA:MAX:0.5:360:5840*

*RRA:MIN:0.5:1:2880*

*RRA:MIN:0.5:5:2880*

*RRA:MIN:0.5:30:4320*

*RRA:MIN:0.5:360:5840*

Now restart Nagios. If you look in */usr/local/pnp4nagios/var/perfdata*, you should start to see rrd files created by pnp4nagios for all your monitored hosts and services.

Verify that you have all the components to use the web interface by checking *http://your-nagios-host/pnp4nagios/*

Correct any missing components, then delete the */usr/local/pnp4nagios/share/install.php* file to see your first graphs:

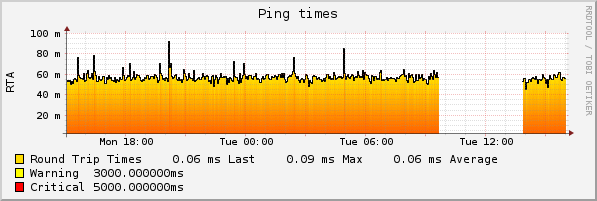


Figure3 : Graph generated for PNP4Nagios

There is one more step to complete the setup. We need to enable extended info in Nagios so that links to the graphs are created for each applicable host and service.

Append two entries to */usr/local/nagios/etc/objects/templates.cfg:*

*define host {*

*name host-pnp*

*action\_url /pnp4nagios/index.php/graph?host=$HOSTNAME$&srv=\_HOST\_*

*register 0*

*}*

*define service {*

*name srv-pnp*

*action\_url /pnp4nagios/index.php/graph?host=$HOSTNAME$&srv=$SERVICEDESC$*

*register 0*

*}*

These are templates that you add to each host and service definition with graphs:

*define host {*

*use linux-server,host-pnp*

*host\_name ubuntu*

*alias ubuntu*

*address 127.0.0.1*

*}*

*define service {*

*use local-service,srv-pnp*

*host\_name ubuntu*

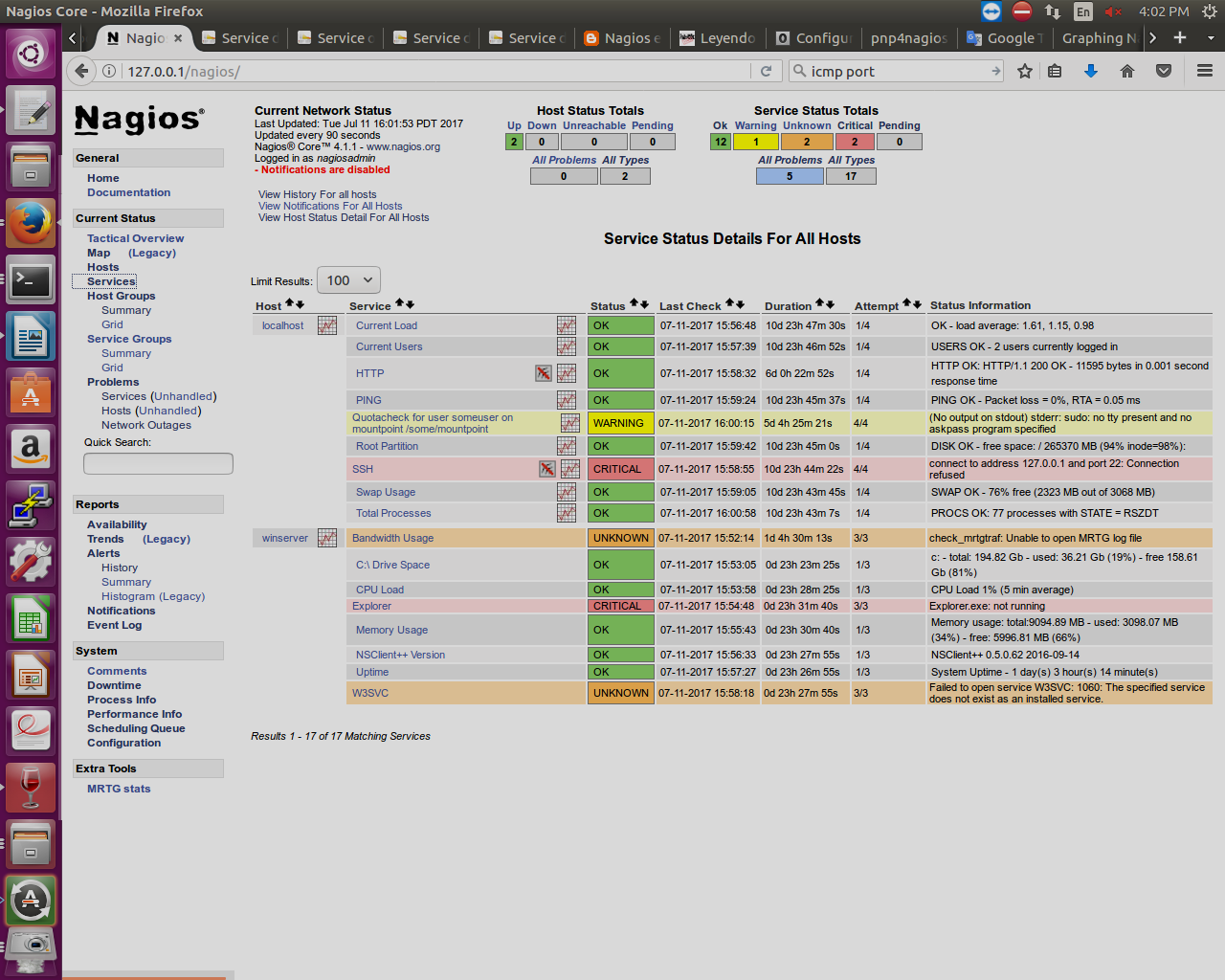
*service\_description PING*

*check\_command check\_ping!100.0,20%!500.0,60%*

*}*

After restarting Nagios again, the icons for the graphs should appear next to the hosts and services, as is showed in the following image. And by click in the graphs icon you gain access to the graphs itself.

***\*\*\* NOTE: Please review the file README from the package cf.monitoring.utilities.1-0.deb, to verify which files and/or configurations files should be replaced.***

Figure4 : Graphs Icons next to the Host and Services