Newtest Smart Plug-in for Nagios

Installation Guide

Version 2.0 - October 2019

ip-label

90 boulevard National, 92250 La Garenne-Colombes | **www.ip-label.com**
SAS with capital of 549,728 € | RCS Nanterre B 327 139 309

Contents

1 PURPOSE

The purpose of this document is to explain how to integrate Newtest into Nagios monitoring. This integration relies on a Nagios plug-in which is provided to supervise Newtest probe and scenario objects. All modules interfacing Newtest with Nagios are Windows and Unix-compatible.

The advantage of this solution is to open a single flow between the Newtest Management Console and the Nagios server.

This document describes integration on a Linux platform. On a Windows platform the directory names for the Nagios files may be different.

2 REQUIREMENTS

2.1 NEWTEST REQUIREMENTS

The "CCNEP6: Newtest Operation Portal" option defined in the Newtest Management Console is required to enable the plug-in to retrieve the states of robots and scenarios.

There is no software to install on the Newtest Enterprise Portal.

2.2 NAGIOS REQUIREMENTS

- The Newtest Smart Plug-in for Nagios, compatible with Nagios V4.
- A Python code interpreter installed on the machine hosting the Nagios console.
- An account with "write" privileges to manage the Nagios configuration files and the five following components (provided by ip-label):
  - Newtestlib.py
  - Newtest_plugin_rbt.py
  - Newtest_plugin_rbt_sce.py
  - Newtest_pooler.py
  - xmltodict.py

3 CONCEPT

The main concept entails integrating Newtest probes and scenario monitoring by using two plug-ins and a pooler that collect data and status from these two types of Newtest object.

This integration solution is built on two modules:

- a pooler which uses web services requests to retrieve data pertaining to all robots and scenarios present on the Newtest Management Console
- a set of two Nagios plug-ins which transmit data to Nagios in a Nagios-compatible format.

Thus, the workflow of the solution includes the following steps:

- Pooler: the execution is scheduled on the server hosting the Nagios supervision console. It retrieves status requests for all robots and scenarios, as well as the values of the latest execution of all scenarios and robots present within the configuration of the NMC server.
- Call to the robot plug-in (one call per robot) via Nagios: analysis of the pooler's result file for the status of the target robot,
- Active check: call to the scenario plug-in (one call per scenario) via Nagios: analysis of the pooler's result file for the target scenario,

OR

- Passive check: triggers passive check of all scenarios on a robot when active check of the robot is enabled.

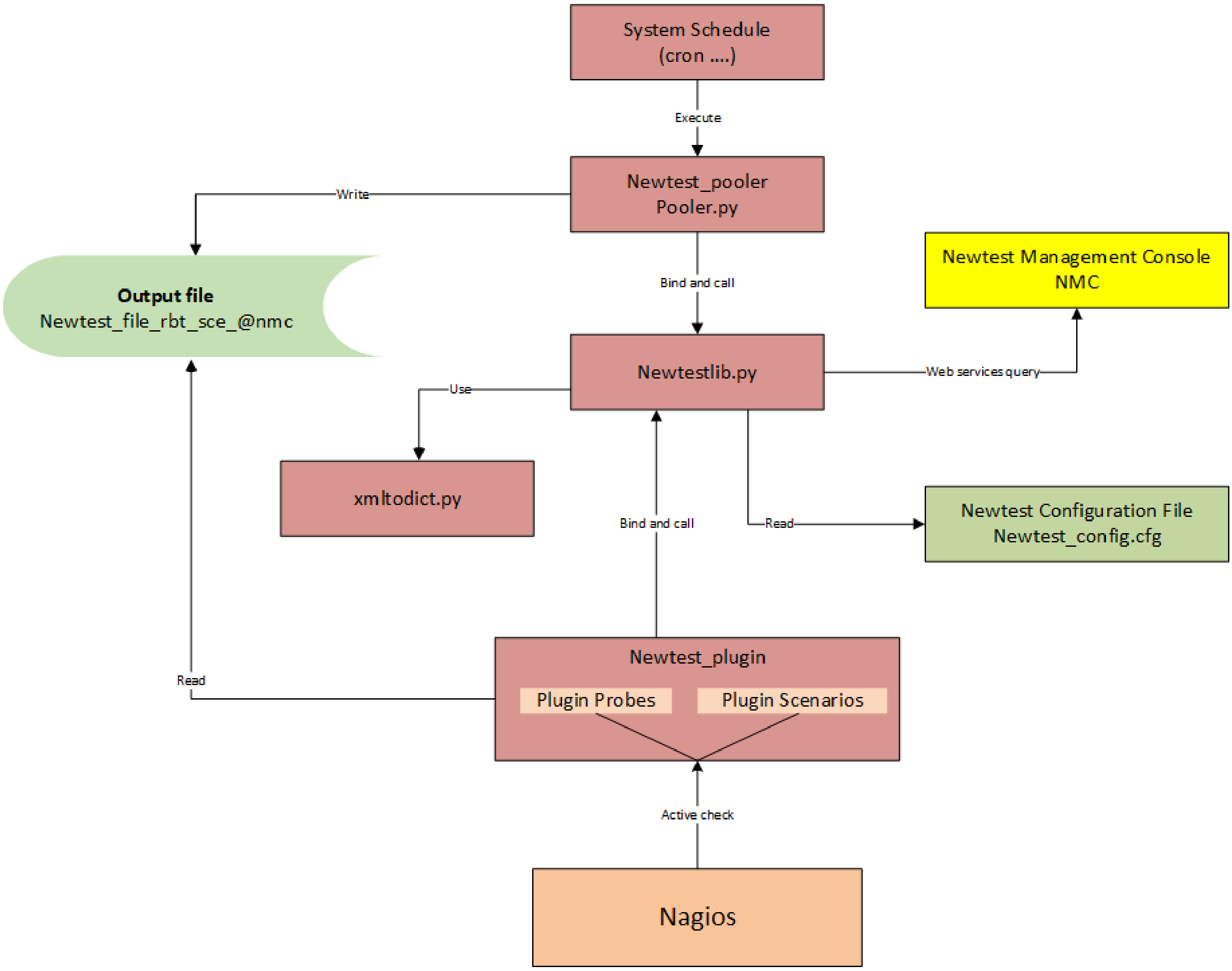
3.1 SOFTWARE COMPONENTS OF THE SOLUTION

The Newtest Smart Plug-in for Nagios comprises seven files:

- Newtest_pooler.py: module which collects statuses and measurements from scenarios and robots via web services requests to NMC
- Newtestlib.py: Newtest library used by the Newtest_pooler.py script and the Newtest_plugin_rbt.py script in addition to Newtest_plugin_rbt_sce.py
- Newtest_plugin_rbt.py: Nagios plug-in for robot statuses
- Newtest_plugin_rbt_sce.py: Nagios plug-in for scenario statuses
- xmltodict.py: library that makes working with XML feel like you are working with JSON
- newtest_config.cfg: configuration file for the plug-ins and pooler
- Newtest_file_rbt_sce_@NMC: file resulting from pooler execution, containing all data on statuses collected by web services requests, for robot and scenario objects

Note: "@NMC" represents the name or IP address of the Newtest server where the dot character is replaced by the underscore character.

3.1 ARCHITECTURE OF NEWTEST INTEGRATION INTO NAGIOS



General outlline of operation of the plugin

3.2 DESCRIPTION OF THE COMPONENTS

3.2.1 NEWTEST_POOLER.PY SCRIPT

Newtest_pooler.py configuration_file debug_mode	
configuration_file	configuration filename including the directory
debug_mode	1: debug mode is enabled 0: debug mode is disabled (any value different from 1) The progress of the pooler is displayed in the standard output

Example:

```
python3 Newtest_pooler.py /usr/local/nagios/etc/newtest_config.cfg 1
```

3.2.2 NEWTEST_PLUGIN_RBT.PY SCRIPT

Newtest_plugin_rbt.py	configuration_file *mode* NMC_server *robot_name* Nagios_robot_host
configuration_file	configuration filename including the directory
mode	activation / deactivation of passive pooling of scenario statuses p: passive mode a: active mode (any value different from p) Defines whether the scenarios are also retrieved or not
NMC_server	IP address or hostname of the NMC server
robot_name	name of the Newtest robot on which the search will be run
Nagios_robot_host	the Nagios name of the host supporting the Nagios service testing the Newtest robot used only in passive mode

Example:

On Nagios, the host host_berlin supports the active check service of the robot named under Newtest berlin.

Passive mode execution retrieves data pertaining to the robot and all scenarios running on the robot:

```
python3 Newtest_plugin_rbt.py /usr/local/nagios/etc/newtest_config.cfg p
192.168.1.2 berlin host_berlin
```

Active mode execution retrieves data pertaining only to the robot:

```
python3 Newtest_plugin_rbt.py /usr/local/nagios/etc/newtest_config.cfg a
192.168.1.2 berlin
```

3.2.3 NEWTEST_PLUGIN_RBT_SCE.PY SCRIPT

Newtest_plugin_rbt_sce.py configuration_file *mode* NMC_server *robot_name* scenario_name	
configuration_file	configuration filename including the directory
mode	activation / deactivation of passive pooling of scenario statuses p: passive mode a: active mode (any value different from p)
NMC_server	IP address or hostname of the NMC server
robot_name	name of the Newtest robot on which the search will be run
scenario_name	name of the scenario associated with the robot used only in active mode

The script is called in passive mode by the Newtest_plugin_robot.pl: all scenario statuses for a given robot are fed back via the write mechanism of the Nagios command file.

Example:

```
python3 Newtest_plugin_rbt_sce.py /usr/local/nagios/etc/newtest_config.cfg a
192.168.1.2 berlin mailtest
```

4 DEPLOYMENT AND USE OF THE COMPONENTS

4.1 INSTALLING PLUG-IN MODULES

- Check the format of the files (if they are formatted for Windows, on Linux or Unix machines you must use the *dos2unix* command to convert the files into Unix format).
- Save Newtestlib.py, xmltodict.py, Newtest_plugin_rbt.pl, Newtest_plugin_rbt_sce.pl and Newtest_pooler.pl to the Nagios plug-in directory.
- Ensure that the Nagios user can run all plug-in files. On Linux, you can run the following command to change the owner of the files:

```
$chown nagios:nagios Newtest_plugin_rbt.py Newtest_plugin_rbt_sce.py
Newtest_pooler.py Newtestlib.py xmltodict.py
```

4.2 SETTING THE CONFIGURATION FILE

You should edit the configuration file newtest_config.cfg located in the directory /usr/local/nagios/etc.

The configuration file, "**newtest_config.cfg**", contains the following data:

- The URL corresponding to the Newtest NMC web service destination for collection requests. The format is: url=your URL.
- The NMC address to which all web services requests will be made. The format is: nmc=address of your NMC, without the preceding "http"; this corresponds to the name or IP address of your NMC.
- The HTTP(S) port destination of web services requests. The format is: port=the port number.
- The login and password used for the authentication.
- The path or location where the output files will be stored (Newtest_file_rbt_sce_@NMC and the trace files of the pooler and plug-ins). The format is: newtest_output_folder=your path and: newtest_log_folder=your path.
- The name of the Nagios command file destination for data collected in passive mode: passive_file=your path.
- The delay between two polling executions expressed in seconds (by default 30 seconds). The format is: timeout=your timeout.

This file must be readable and executable by the Nagios user.

Example:

```
url=https://nepv4.ip-label.net/nws/ManagementConsoleService.asmx

ip=nepv4.ip-label.net

port=443

login=user_login

password=user_password

newtest_output_folder=/tmp/

newtest_log_folder=/tmp/

passive_file=/usr/local/nagios/var/nagios.cmd
```

timeout = 30

A line starting with "#" is considered as a comment line and all characters must be in lowercase.

To maintain a simple syntax for the configuration file, the following examples would not be tolerated and would cause problems:

`Port=80#your comments`

In this case the port value retrieved would be "80#your comments". This would cause a problem because such a value is meaningless.

`Port= 80` would not be accepted because of the space preceding the value 80.

All the cases covered in these comments are applicable to all the lines of the file newtest_config.cfg.

4.3 SCHEDULING THE POOLER

- Schedule Newtest_pooler.pl for every *x* minutes so that data in the file Newtest_file_rbt_sce_@NMC is meaningful. It is advisable to use the crontab (on Linux) with the following commands:

# crontab -u nagios –e

*/5 * * * * /usr/bin/python3/usr/local/nagios/libexec/Newtest_pooler.py
/usr/local/nagios/etc/newtest_config.cfg 0   2>&1

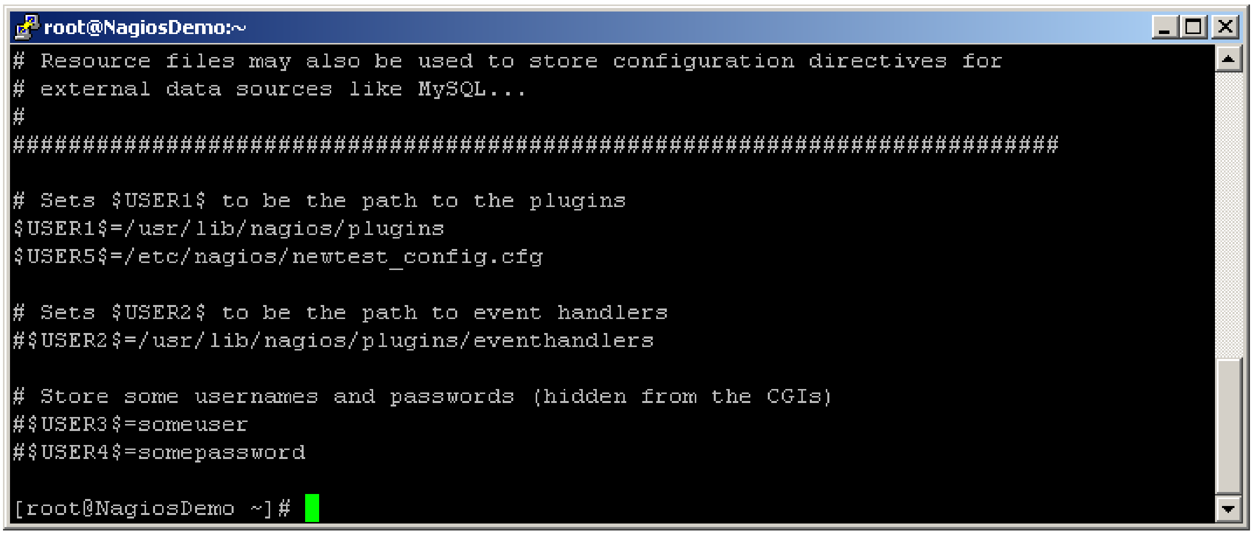
Note: All scripts must be executable by the Nagios user.

5 INTEGRATION WITH NAGIOS

5.1 ADDING THE NEWTEST CONFIGURATION FILEPATH

Add a $USER$ variable to the Nagios ***resource.cfg*** file and assign to it the path to access the Newtest configuration file.

Example: $USER5$=/usr/local/nagios/etc//newtest_config.cfg



```
# Resource files may also be used to store configuration directives for
# external data sources like MySQL...
#
###########################################################################

# Sets $USER1$ to be the path to the plugins
$USER1$=/usr/lib/nagios/plugins
$USER5$=/etc/nagios/newtest_config.cfg

# Sets $USER2$ to be the path to event handlers
#$USER2$=/usr/lib/nagios/plugins/eventhandlers

# Store some usernames and passwords (hidden from the CGIs)
#$USER3$=someuser
#$USER4$=somepassword

[root@NagiosDemo ~]#
```

5.2 ADDING COMMANDS FOR THE NEWTEST_PLUGIN FILES

Commands for both files are added to the same file named ***commands.cfg***, usually located in the directory */etc/nagios/objects*, as follows:

```
define command{

command_name      plugin_rbt
command_line      /usr/bin/python3 $USER1$/Newtest_plugin_rbt.py $USER5$
$ARG1$ $ARG2$ $ARG3$ $HOSTNAME$
# If you use the same name for the probe in Nagios as in NMC, the command
line could become:
#command_line     /usr/bin/python3 $USER1$/Newtest_plugin_rbt.py $USER5$
$ARG1$ $ARG2$ $HOSTNAME$ $HOSTNAME$
}
```

```
define command{

command_name      plugin_rbt_sce
command_line      /usr/bin/python3 $USER1$/Newtest_plugin_rbt_sce.py
$USER5$ $ARG1$ $ARG2$ $ARG3$ $ARG4$
# If you use the same name for the probe in Nagios as in NMC, the command
line could become:
#command_line     /usr/bin/python3 $USER1$/Newtest_plugin_rbt_sce.py
$USER5$ $ARG1$ $ARG2$ $HOSTNAME$ $ARG3$

}
```

5.3 ADDING HOSTS AND SERVICES

Hosts and services are added into the **windows.cfg** file usually located in the directory */etc/nagios/objects.*

Comment: it is advisable to add one host per robot.

This section provides several examples:

Adding a host:

```
define host{
     use          windows-server    ; Inherit default values from a template
     host_name    Madrid            ; The name we're giving to this host
     alias        Probe madrid      ; A longer name associated with the host
     address      192.168.2.211     ; IP address of the host
 }
```

Adding a service to retrieve data about robots (Newtest_plugin_rbt.pl)

```
define service{
    use                      generic-service
    host_name                Madrid
    service_description      newtest_Madrid
    normal_check_interval    1
    check_command            plugin_rbt!a!192.168.1.2!Madrid
 }
```

Adding a service to retrieve data about scenarios (Newtest_plugin_rbt_sce.pl)

The following pertains only to active-mode execution of the file Newtest_plugin_rbt_sce.pl

```
define service{
    use                      generic-service
    host_name                Madrid
    service_description      MADRID_RESERVATION
    active_checks_enabled    1
    check_command            plugin_rbt_sce!a!192.168.1.2!Madrid!reservation
 }
```

Comments:

- For the services associated with the plug-in for scenarios running in passive mode, no value is supplied to the parameters because this file runs with the same arguments as those given for the Newtest_plugin_rbt.pl file.
- If you wish to use your own template for the host and/or services, you can add it to the templates.cfg file.

APPENDIX A : STATUS CONVERSION

The following tables set forth the conversion of NMC robot statuses to Nagios:

ROBOT STATUS	NAGIOS VALUE	PERFORMANCE (PERF)	NAGIOS RESULT VALUE
OK	OK	10	0
SUSPENDED	WARNING	5	1
STOPPED	WARNING	5	1
UNKNOWN	CRITICAL	0	2
UNINSTALLED	CRITICAL	0	2
INOPERATIVE	CRITICAL	0	2
RESUME	WARNING	5	1

SCENARIO	NAGIOS VALUE	NAGIOS RESULT VALUE
Available	OK	0
Warning	WARNING	1
Failed	CRITICAL	2
Canceled	CRITICAL	2
Unknown	UNKNOWN	3
OutOfRange	UNKNOWN	3

APPENDIX B: NEWTEST_FILE_RBT_@NMC

This file is the output of pooler execution (using the library Newtestlib.pm) and is used by the plug-ins Newtest_plugin_rbt.pl and Newtest_plugin_rbt_sce.pl.

The file named file_rbt_sce_hostname (@NMC) contains the data collected by web services requests from robots and scenarios (parent and child measurements) written in the following format:

Robots, whose structure in the file is:

- Line prefix "RBT": first element (header) followed by the separator " || ".
- Robot name: second element followed by the separator " || ".
- Newtest robot status: third element followed by the separator " || ".
- Last message sent by the robot (Last message): fourth element followed by the separator.

Example:

```
RBT||PARIS||OK||2019-09-05T17:28:22||
```

Scenarios (parent measurement), whose structure in the file is:

- Prefix "SCE": first element (header) followed by the separator " || ".
- Robot name: second element followed by the separator " || ".
- Name of the scenario on the robot: third element followed by the separator “ = ”.
- Scenario web services ID: fourth element followed by the separator " || ".
- Newtest scenario status: fifth element followed by the separator " || ".
- Last message sent by the scenario: sixth element followed by the separator " || ".

Example:

```
SCE||MADRID||AGENCIES_ACCESS=194412048||OUTOFRANGE||2019-09-05T17:26:12||
```

Child measurements, whose record structure is:

- Separator " !! " differentiates parent scenarios from child scenarios.
- Child measurement name followed by the separator “ = ”.
- Value of the child measurement followed by the separator " || ".
- Threshold (if any, other than -1).

Comment: This principle is repeated (on the same line) for all of the scenario’s child measurements.

- At the end, the separator " || " is added to finish the line.

Example:

```
SCE||MADRID||CORPORATE_MAIN=194412049||OUTOFRANGE||2019-09-
05T17:26:12!!port_test=0.00000,70.00000!!port1_dns=1.00000,-
1!!port2_connect=4.00000,-1||
```

Comment:

If a scenario returns nothing (no value in the xml result), the corresponding line is structured as follows:

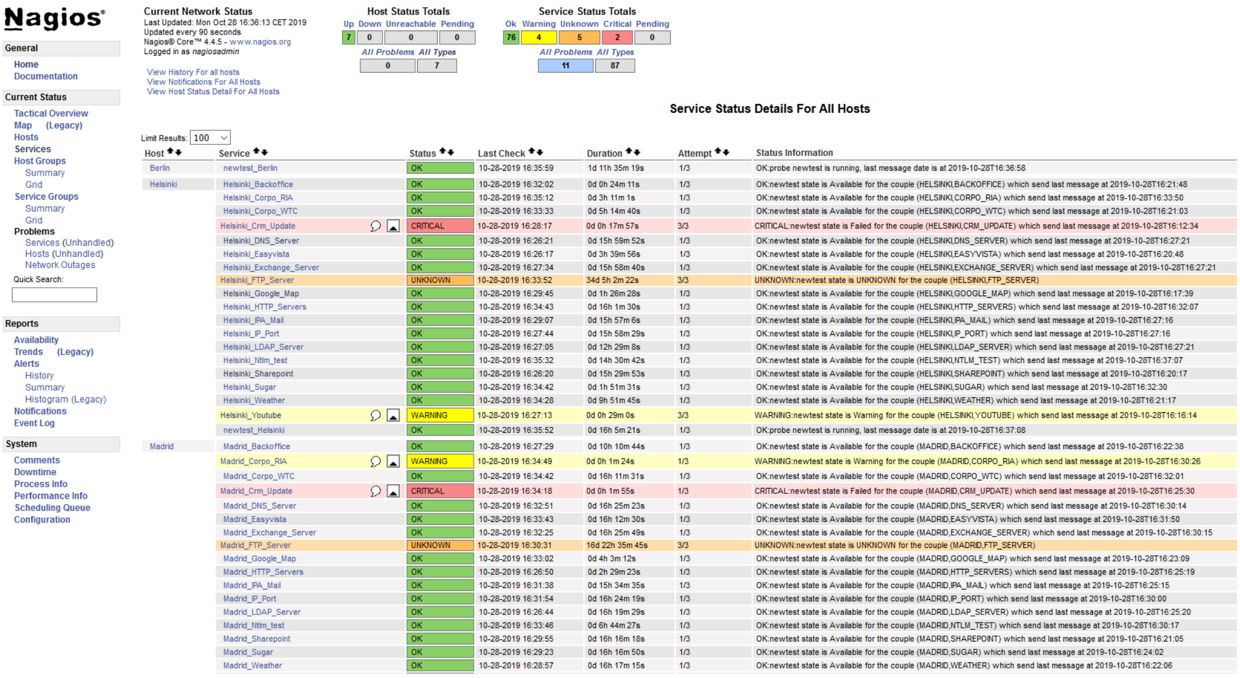
```
SCE||robot name||scenario name
```

Example:

```
SCE||BERLIN||CARGO
```

APPENDIX C: NAGIOS CONSOLE SCREENSHOTS

The figures below show some screenshots of Nagios consoles.



Nagios®

General
- Home
- Documentation

Current Status
- Tactical Overview
- Map (Legacy)
- Hosts
- Services
- Host Groups
  - Summary
  - Grid
- Service Groups
  - Summary
  - Grid
- Problems
  - Services (Unhandled)
  - Hosts (Unhandled)
  - Network Outages

Quick Search:

Reports
- Availability
- Trends (Legacy)
- Alerts
  - History
  - Summary
  - Histogram (Legacy)
- Notifications
- Event Log

System
- Comments
- Downtime
- Process Info
- Performance Info
- Scheduling Queue
- Configuration

Current Network Status

Last Updated: Mon Oct 28 16:36:13 CET 2019
Updated every 90 seconds
Nagios® Core™ 4.4.5 - www.nagios.org
Logged in as *nagiosadmin*

View History For all hosts
View Notifications For All Hosts
View Host Status Detail For All Hosts

Host Status Totals

Up	Down	Unreachable	Pending
7	0	0	0

All Problems	*All Types*
0	7

Service Status Totals

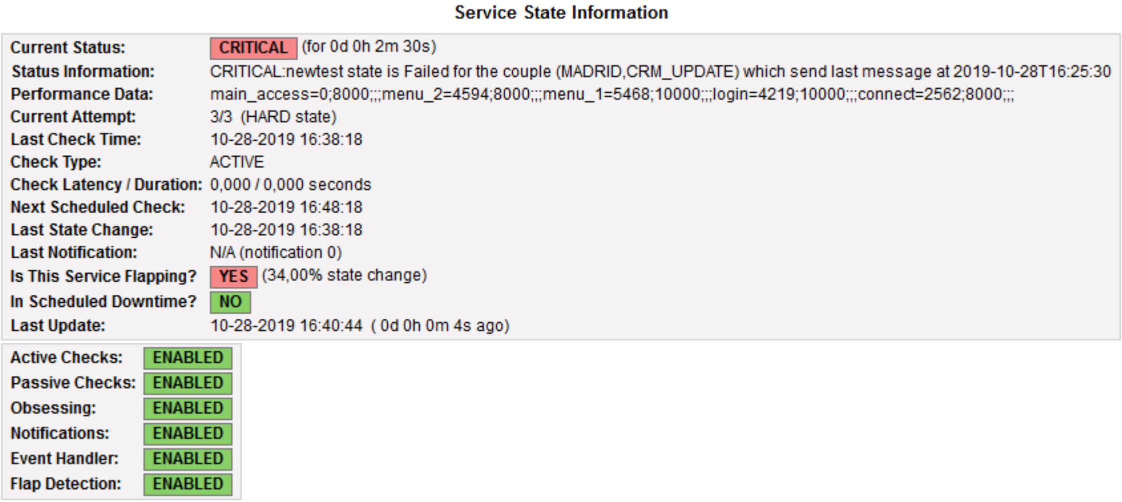
Ok	Warning	Unknown	Critical	Pending
76	4	5	2	0

All Problems	*All Types*
11	87

Service Status Details For All Hosts

Limit Results: 100

Host	Service	Status	Last Check	Duration	Attempt	Status Information
Berlin	newtest_Berlin	OK	10-28-2019 16:35:59	1d 11h 35m 19s	1/3	OK:probe newtest is running, last message date is at 2019-10-28T16:36:58
Helsinki	Helsinki_Backoffice	OK	10-28-2019 16:32:02	0d 0h 24m 11s	1/3	OK:newtest state is Available for the couple (HELSINKI,BACKOFFICE) which send last message at 2019-10-28T16:21:48
	Helsinki_Corpo_RIA	OK	10-28-2019 16:35:12	0d 3h 11m 1s	1/3	OK:newtest state is Available for the couple (HELSINKI,CORPO_RIA) which send last message at 2019-10-28T16:33:50
	Helsinki_Corpo_WTC	OK	10-28-2019 16:33:33	0d 5h 14m 40s	1/3	OK:newtest state is Available for the couple (HELSINKI,CORPO_WTC) which send last message at 2019-10-28T16:21:03
	Helsinki_Crm_Update	CRITICAL	10-28-2019 16:28:17	0d 0h 17m 57s	3/3	CRITICAL:newtest state is Failed for the couple (HELSINKI,CRM_UPDATE) which send last message at 2019-10-28T16:12:34
	Helsinki_DNS_Server	OK	10-28-2019 16:26:21	0d 15h 59m 52s	1/3	OK:newtest state is Available for the couple (HELSINKI,DNS_SERVER) which send last message at 2019-10-28T16:27:21
	Helsinki_Easyvista	OK	10-28-2019 16:26:17	0d 3h 39m 56s	1/3	OK:newtest state is Available for the couple (HELSINKI,EASYVISTA) which send last message at 2019-10-28T16:20:48
	Helsinki_Exchange_Server	OK	10-28-2019 16:27:34	0d 15h 58m 40s	1/3	OK:newtest state is Available for the couple (HELSINKI,EXCHANGE_SERVER) which send last message at 2019-10-28T16:27:21
	Helsinki_FTP_Server	UNKNOWN	10-28-2019 16:33:52	34d 5h 2m 22s	3/3	UNKNOWN:newtest state is UNKNOWN for the couple (HELSINKI,FTP_SERVER)
	Helsinki_Google_Map	OK	10-28-2019 16:29:45	0d 1h 26m 28s	1/3	OK:newtest state is Available for the couple (HELSINKI,GOOGLE_MAP) which send last message at 2019-10-28T16:17:39
	Helsinki_HTTP_Servers	OK	10-28-2019 16:34:43	0d 16h 1m 30s	1/3	OK:newtest state is Available for the couple (HELSINKI,HTTP_SERVERS) which send last message at 2019-10-28T16:32:07
	Helsinki_IPA_Mail	OK	10-28-2019 16:29:07	0d 15h 57m 6s	1/3	OK:newtest state is Available for the couple (HELSINKI,IPA_MAIL) which send last message at 2019-10-28T16:27:16
	Helsinki_IP_Port	OK	10-28-2019 16:27:44	0d 15h 58m 29s	1/3	OK:newtest state is Available for the couple (HELSINKI,IP_PORT) which send last message at 2019-10-28T16:27:16
	Helsinki_LDAP_Server	OK	10-28-2019 16:27:05	0d 12h 29m 8s	1/3	OK:newtest state is Available for the couple (HELSINKI,LDAP_SERVER) which send last message at 2019-10-28T16:27:21
	Helsinki_Ntlm_test	OK	10-28-2019 16:35:32	0d 14h 30m 42s	1/3	OK:newtest state is Available for the couple (HELSINKI,NTLM_TEST) which send last message at 2019-10-28T16:37:07
	Helsinki_Sharepoint	OK	10-28-2019 16:26:20	0d 15h 29m 53s	1/3	OK:newtest state is Available for the couple (HELSINKI,SHAREPOINT) which send last message at 2019-10-28T16:20:17
	Helsinki_Sugar	OK	10-28-2019 16:34:42	0d 1h 51m 31s	1/3	OK:newtest state is Available for the couple (HELSINKI,SUGAR) which send last message at 2019-10-28T16:32:30
	Helsinki_Weather	OK	10-28-2019 16:34:28	0d 9h 51m 45s	1/3	OK:newtest state is Available for the couple (HELSINKI,WEATHER) which send last message at 2019-10-28T16:21:17
	Helsinki_Youtube	WARNING	10-28-2019 16:27:13	0d 0h 29m 0s	3/3	WARNING:newtest state is Warning for the couple (HELSINKI,YOUTUBE) which send last message at 2019-10-28T16:16:14
	newtest_Helsinki	OK	10-28-2019 16:35:52	0d 16h 5m 21s	1/3	OK:probe newtest is running, last message date is at 2019-10-28T16:37:08
Madrid	Madrid_Backoffice	OK	10-28-2019 16:27:29	0d 10h 10m 44s	1/3	OK:newtest state is Available for the couple (MADRID,BACKOFFICE) which send last message at 2019-10-28T16:22:38
	Madrid_Corpo_RIA	WARNING	10-28-2019 16:34:49	0d 0h 1m 24s	1/3	WARNING:newtest state is Warning for the couple (MADRID,CORPO_RIA) which send last message at 2019-10-28T16:30:26
	Madrid_Corpo_WTC	OK	10-28-2019 16:34:42	0d 16h 11m 31s	1/3	OK:newtest state is Available for the couple (MADRID,CORPO_WTC) which send last message at 2019-10-28T16:32:01
	Madrid_Crm_Update	CRITICAL	10-28-2019 16:34:18	0d 0h 1m 55s	1/3	CRITICAL:newtest state is Failed for the couple (MADRID,CRM_UPDATE) which send last message at 2019-10-28T16:25:30
	Madrid_DNS_Server	OK	10-28-2019 16:32:51	0d 16h 25m 23s	1/3	OK:newtest state is Available for the couple (MADRID,DNS_SERVER) which send last message at 2019-10-28T16:30:14
	Madrid_Easyvista	OK	10-28-2019 16:33:43	0d 16h 12m 30s	1/3	OK:newtest state is Available for the couple (MADRID,EASYVISTA) which send last message at 2019-10-28T16:31:50
	Madrid_Exchange_Server	OK	10-28-2019 16:32:25	0d 16h 25m 49s	1/3	OK:newtest state is Available for the couple (MADRID,EXCHANGE_SERVER) which send last message at 2019-10-28T16:30:15
	Madrid_FTP_Server	UNKNOWN	10-28-2019 16:30:31	16d 22h 35m 45s	3/3	UNKNOWN:newtest state is UNKNOWN for the couple (MADRID,FTP_SERVER)
	Madrid_Google_Map	OK	10-28-2019 16:33:02	0d 4h 3m 12s	1/3	OK:newtest state is Available for the couple (MADRID,GOOGLE_MAP) which send last message at 2019-10-28T16:23:09
	Madrid_HTTP_Servers	OK	10-28-2019 16:26:50	0d 2h 29m 23s	1/3	OK:newtest state is Available for the couple (MADRID,HTTP_SERVERS) which send last message at 2019-10-28T16:25:19
	Madrid_IPA_Mail	OK	10-28-2019 16:31:38	0d 15h 34m 35s	1/3	OK:newtest state is Available for the couple (MADRID,IPA_MAIL) which send last message at 2019-10-28T16:25:15
	Madrid_IP_Port	OK	10-28-2019 16:31:54	0d 16h 24m 19s	1/3	OK:newtest state is Available for the couple (MADRID,IP_PORT) which send last message at 2019-10-28T16:30:00
	Madrid_LDAP_Server	OK	10-28-2019 16:26:44	0d 16h 19m 29s	1/3	OK:newtest state is Available for the couple (MADRID,LDAP_SERVER) which send last message at 2019-10-28T16:25:20
	Madrid_Ntlm_test	OK	10-28-2019 16:33:46	0d 6h 44m 27s	1/3	OK:newtest state is Available for the couple (MADRID,NTLM_TEST) which send last message at 2019-10-28T16:30:17
	Madrid_Sharepoint	OK	10-28-2019 16:29:55	0d 16h 16m 18s	1/3	OK:newtest state is Available for the couple (MADRID,SHAREPOINT) which send last message at 2019-10-28T16:21:05
	Madrid_Sugar	OK	10-28-2019 16:29:23	0d 16h 16m 50s	1/3	OK:newtest state is Available for the couple (MADRID,SUGAR) which send last message at 2019-10-28T16:24:02
	Madrid_Weather	OK	10-28-2019 16:28:57	0d 16h 17m 15s	1/3	OK:newtest state is Available for the couple (MADRID,WEATHER) which send last message at 2019-10-28T16:22:06



Service State Information

Current Status:	CRITICAL (for 0d 0h 2m 30s)
Status Information:	CRITICAL:newtest state is Failed for the couple (MADRID,CRM_UPDATE) which send last message at 2019-10-28T16:25:30
Performance Data:	main_access=0;8000;;;menu_2=4594;8000;;;menu_1=5468;10000;;;login=4219;10000;;;connect=2562;8000;;;
Current Attempt:	3/3 (HARD state)
Last Check Time:	10-28-2019 16:38:18
Check Type:	ACTIVE
Check Latency / Duration:	0,000 / 0,000 seconds
Next Scheduled Check:	10-28-2019 16:48:18
Last State Change:	10-28-2019 16:38:18
Last Notification:	N/A (notification 0)
Is This Service Flapping?	YES (34,00% state change)
In Scheduled Downtime?	NO
Last Update:	10-28-2019 16:40:44 (0d 0h 0m 4s ago)

Active Checks:	ENABLED
Passive Checks:	ENABLED
Obsessing:	ENABLED
Notifications:	ENABLED
Event Handler:	ENABLED
Flap Detection:	ENABLED