

FUJITSU Software ServerView Suite

FUJITSU Server Plug-ins V3.00 for Nagios Core

Interface Documentation

August 2014 Edition

© Copyright Fujitsu Technology Solutions GmbH 2014

All hardware and software names used are trademarks of their respective manufacturers.

All rights, including rights of translation, reproduction by printing, copying or similar methods, in part or in whole, are reserved.
Offenders will be liable for damages.

All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

Delivery subject to availability. Right of technical modification reserved.

CONTENTS

1	Introduction	8
1.1	Product Names - Conventions	8
1.2	Target Groups and Purpose of this Manual	8
1.3	Change History	8
1.4	Documentation	9
1.4.1	Documentation for Nagios/Icinga, wbemcli, OpenWSMAN Link	9
1.4.2	Documentation for ServerView Suite	9
1.5	Licensing the Software	9
1.5.1	License for ServerView Suite	9
1.5.2	License for Nagios/Icinga	9
2	Solution Overview	10
2.1	Nagios versus Icinga	10
2.2	Supported Features and Capabilities – Monitored Information	10
2.3	CIM Support: Supported Servers	11
2.3.1	Wbemcli – CIM-XML protocol	12
2.3.2	OpenWSMAN Perl Binding – WS-MAN protocol	12
2.4	SNMP Support: Supported Servers and SNMP MIBs Used	12
2.5	Monitoring via iRMC	13
3	Installation and Configuration	14
3.1	Sources	14
3.2	Requirements	14
3.3	Installation Instructions	15
3.4	Nagios Core Sample Configurations	15
3.4.1	Preliminaries (SNMP)	15
3.4.2	Host Group Definitions	16
3.4.2.1	SNMP - All Fujitsu Servers	16
3.4.2.2	SNMP - PRIMERGY Server Variants	16
3.4.2.3	SNMP - PRIMERGY Blade Variants	16
3.4.2.4	SNMP - PRIMEQUEST Variants	17
3.4.2.5	CIM - All Fujitsu Servers	17
3.4.2.6	CIM - PRIMERGY Server Variants	17
3.4.3	Command Definitions Usable for these Host Groups	18
3.4.3.1	Commands for SNMP Plug-in (check_fujitsu_server.pl)	18
3.4.3.2	Commands for CIM Plug-in (check_fujitsu_server_CIM.pl)	18
3.4.4	Service Definitions Usable for these Host Groups	19
3.4.4.1	SNMP - PRIMERGY Server	19
3.4.4.2	SNMP - PRIMERGY Blade	20
3.4.4.3	SNMP - PRIMEQUEST	22
3.4.4.4	CIM - PRIMERGY Server	23
3.4.5	Service-Dependency Definitions for these Services	25
3.4.5.1	SNMP - PRIMERGY Server	25
3.4.5.2	SNMP - PRIMERGY Blades	25
3.4.5.3	SNMP - PRIMEQUEST	26
3.4.5.4	CIM - PRIMERGY Server	27
3.4.6	Host Template for MMB Web Server Address or Option Settings	27
3.4.7	Sample Host Definitions	28
3.4.7.1	SNMP Host Samples	28
3.4.7.2	CIM Host Samples	29
3.4.8	Special Notification Command	29

4	FUJITSU SNMP Server Check Plug-in: check_fujitsu_server.pl	30
4.1	Basics	30
4.1.1	Script Name	30
4.1.2	Options around the Plug-in itself	30
4.1.3	SNMP Options around Addressing	30
4.1.4	About IPv6 Usage	31
4.1.5	Plug-in Processing Control Option	31
4.2	Common Rules for Printouts	31
4.2.1	Text Rules for Automatic Scanning of Printouts	31
4.2.2	Verbose Level Usage	32
4.3	Tool Features	33
4.3.1	Simple SNMP Uptime Check	33
4.3.2	Get Only System Information Used for Notify Information	33
4.4	Check of PRIMERGY Server or Server where ServerView SNMP Agent is Installed	34
4.4.1	Options	34
4.4.2	System Status Information	34
4.4.3	Special Information for PRIMERGY Multi-Node Servers (CX-series)	35
4.4.4	System Information in Case of WARNING or CRITICAL - for Notifications	35
4.4.5	Additional System Information	36
4.4.5.1	How to Detect a Parent Management Blade	37
4.4.6	Power Supply Information	37
4.4.7	Power Consumption Information	37
4.4.8	Fan Information	37
4.4.9	Temperature Sensor Information	38
4.4.10	Hardware – Voltage Information	39
4.4.11	Hardware – CPU Information	40
4.4.12	Hardware – Memory Module Information	40
4.4.13	System – MassStorage (RAID) Information	41
4.4.14	Driver Monitor Information	43
4.5	Check for RAID Standalone	43
4.5.1	Options	43
4.5.2	RAID Information	43
4.6	Check of PRIMERGY Management Blades	44
4.6.1	Options	44
4.6.2	System Information	44
4.6.3	System Information in the Case of WARNING or CRITICAL - for Notifications	44
4.6.4	Fan Information	44
4.6.5	Temperature Information	45
4.6.6	Power Supply Information	46
4.6.7	Power Consumption Information	46
4.7	Blades Controlled by a Primergy Management Blade	46
4.7.1	Options	46
4.7.2	Information in the Case of WARNING or CRITICAL - for Notifications	47
4.7.3	Server Blades	47
4.7.4	Additional Information about Server Blades	48
4.7.4.1	How to Detect the Host Name and IP Address of Server Blades	49
4.7.5	I/O Connection – FSIOM	49
4.7.6	I/O Connection – Switch Blades	49
4.7.7	I/O Connection – Fibre Channel Switch	50
4.7.8	I/O Connection – Serial Attached SCSI Switch (SASSwitch)	50
4.7.9	I/O Connection – LAN Pass-Through (Phy)	51
4.7.10	Keyboard/Video/Mouse Blades (KVM)	51
4.7.11	Storage Blades	51
4.8	PRIMEQUEST	52
4.8.1	Options	52
4.8.2	System Information - Overall Status and Chassis Information	52
4.8.3	System Information in the Case of WARNING or CRITICAL - for Notifications	53
4.8.4	Additional System Board and Partition Information	53
4.8.4.1	How to Detect Partition-related Host Names and Address	54
4.8.5	Additional System Information – Table of all Units	54
4.8.6	Additional System Information – Status Component Table	55
4.8.7	Additional System Information – Unit-Child Relations	56

4.8.8	Additional System Information – Management Node Table.....	57
4.8.9	Additional System Information – Server Table	57
4.8.10	Temperature Information	57
4.8.11	Fan Information	58
4.8.12	Power Supply Information	58
4.8.13	Power Consumption Information	59
4.8.14	Hardware – Voltage Information	59
4.8.15	Hardware – CPU Information	60
4.8.16	Hardware – Memory Module Information	61
4.9	<i>PRIMERGY Server – Monitoring Update Status</i>	61
4.9.1	Option.....	61
4.9.2	Summary Information Print.....	61
4.9.3	Sample Nagios Configuration.....	61
4.10	<i>Check of PRIMERGY Servers via iRMC SNMP</i>	62
4.10.1	Options	62
4.10.2	System Information in Case of WARNING or CRITICAL - for Notifications	62
4.10.3	System Status Information	63
4.10.4	Power Supply Information	63
4.10.5	Power Consumption Information	64
4.10.6	Fan Information	64
4.10.7	Temperature Sensors Information.....	64
4.10.8	Hardware – Voltage Information	65
4.10.9	Hardware – CPU Information	65
4.10.10	Hardware – Memory-Module Information	66
4.11	<i>Memory Usage - Special Performance Check for PRIMERGY Servers</i>	66
4.11.1	Option.....	66
4.11.2	Summary Information Print.....	67
4.11.3	Sample Nagios Configuration.....	67
4.12	<i>File System Checks - Special Performance Check for PRIMERGY Server</i>	67
4.12.1	Option.....	67
4.12.2	File System Status and Performance Printouts	67
4.12.3	File System Verbose or Non-OK Information	68
4.12.4	Sample Nagios Configuration.....	68
4.13	<i>Network Interface Checks - Special Performance Check for PRIMERGY Servers</i>	68
4.13.1	Option.....	68
4.13.2	Status and Performance Printouts.....	68
4.13.3	Network Interface Verbose or Non-OK Information	69
4.13.4	Sample Nagios Configuration.....	69
4.14	<i>Other Features</i>	69
4.14.1	Plug-in for PRIMERGY Server and CPU Load	69
5	FUJITSU CIM Server Plug-in: check_fujitsu_server_CIM.pl.....	70
5.1	<i>Basics</i>	70
5.1.1	Script Name.....	70
5.1.2	Options around the Plug-in itself	70
5.1.3	Select Usage Mode – CIM-XML or WS-MAN	70
5.1.4	CIM Options around Addressing	70
5.1.5	Plugin Processing Control Option.....	72
5.2	<i>Common Rules for Printouts</i>	72
5.3	<i>Tool Features</i>	72
5.3.1	Simple CIM Identify Check	72
5.3.2	Get Only System Information Used for Notify Information	72
5.4	<i>Check of PRIMERGY Servers where SV CIM Provider is installed</i>	73
5.4.1	Options	73
5.4.2	System Information in the Case of WARNING or CRITICAL - for Notifications	74
5.4.3	System Status Information	74
5.4.4	Additional System Information	74
5.4.5	Fan Information	75
5.4.6	Temperature Sensor Information.....	75
5.4.7	Power Supply Information	76
5.4.8	Power Consumption Information	76

5.4.9	Hardware – Voltage Information	77
5.4.10	Hardware – CPU Information	77
5.4.11	Hardware – Memory Module Information	78
5.5	<i>Check of PRIMERGY Servers via iRMC CIM</i>	78
5.5.1	Options	78
5.5.2	System Information in Case of WARNING or CRITICAL - for Notifications	79
5.5.3	System Status Information	79
5.5.4	Additional System Information	80
5.5.5	Fan Information	80
5.5.6	Temperature Sensors Information	81
5.5.7	Power Supply Information	81
5.5.8	Power Consumption Information	82
5.5.9	Hardware – Voltage Information	82
5.5.10	Hardware – CPU Information	83
5.5.11	Hardware – Memory-Module Information	83
6	FUJITSU SNMP Server Tool: tool_fujitsu_server.pl	84
6.1	<i>Basics</i>	84
6.1.1	Script Name	84
6.1.2	Options around the Script Itself	84
6.1.3	SNMP Options around Addressing	84
6.1.4	About IPv6 Usage	85
6.1.5	Plug-in Processing Control Option	85
6.2	<i>SNMP MIB Test – Test of Server Types</i>	85
6.2.1	Option	85
6.2.2	Common Print Rules	86
6.2.3	Initial Tests – Connection and RC1213.mib Tests	86
6.2.4	ServerView SNMP Agent – Test of Agent-Specific MIB	86
6.2.5	PRIMERGY Blade – Test of S31.mib	87
6.2.6	PRIMEQUEST – Test of MMB-COM-MIB.mib	87
6.2.7	Any other Server – RFC1213.mib only	88
6.3	<i>IPv4 Discovery</i>	88
6.3.1	Option	88
6.3.2	Common Print Rules	88
7	FUJITSU SNMP Trap Configuration Files	89
7.1	<i>Standards</i>	89
7.2	<i>Plugin Support</i>	89
7.3	<i>Brief Hints around SNMPTT</i>	90
8	FUJITSU CIM Server Tool: tool_fujitsu_server_CIM.pl	91
8.1	<i>Basics</i>	91
8.1.1	Script Name	91
8.1.2	Options around the Plug-in itself	91
8.1.3	Select Usage Mode – CIM-XML or WS-MAN	91
8.1.4	CIM Options around Addressing	91
8.1.5	Plugin Processing Control Option	92
8.2	<i>CIM Server Type Test</i>	93
8.2.1	Option	93
8.2.2	Common Print Rules	93
8.2.3	Connection Test	93
8.2.4	Server Type Information with ServerView Agent CIM Information	94
8.3	<i>IPv4 Discovery</i>	95
8.3.1	Option	95
8.3.2	Common Print Rules	95

9	Discovery – Check of Server Types and Generation of Configurations	95
9.1	Script-Hierarchy	95
9.2	Basics	96
9.2.1	Script Name.....	96
9.2.2	Options around the Plugin itself.....	96
9.2.3	Control Options for Called Scripts	96
9.2.4	Select Host Options.....	96
9.2.5	Select Usage Mode – SNMP or CIM-XML or WS-MAN.....	96
9.2.6	Additional Authentication and Connection Options.....	97
9.2.7	Output Options	97
9.3	Processing Loop Overview	97
9.4	Logging Information and Results	97
9.4.1	Central Logging	97
9.4.1.1	Checks for the Option Input Files.....	98
9.4.1.2	Central Process Logging for Each Host.....	98
9.4.1.3	Summary Information.....	99
9.4.2	Logging for Each Host Discovery	99
9.4.3	Text Information for Each Host.....	100
9.4.4	Nagios Configuration Files for Each Host.....	100

1 Introduction

FUJITSU Server Plug-ins for Nagios Core is a collection of scripts and sample configurations and more usable for a Nagios Core integration and all products based on Nagios Core.

These parts can be used for Nagios, Icinga V1.x series, SM-Box or other Nagios Core based variants for enhancements for FUJITSU servers.

Monitoring:

The scripts whose file names start with **check_fujitsu** are usable as Nagios plugins. These scripts can be used standalone but they are meant to be executed as Nagios Core plugins scheduled by the Nagios daemon. With these plugins you can see an overall hardware status of your PRIMERGY servers as well as in-depth details such as power consumption, temperatures, fan speed, and hardware issues.

Supported protocols: SNMP and CIM.

Monitoring is enabled via host address or iRMC address (see "3.2 Requirements" on page 14).

Additional Tools:

The scripts whose file names start with **tool_fujitsu** are for connection and type checks of one server. With the **discover_fujitsu** script any amount of hosts and any connection protocol type can be checked and Nagios hosts configuration files can be generated. These scripts can be used standalone.

Supported protocols: SNMP and CIM.

Sample Nagios Configurations:

A set of sample and template Nagios Core configuration files show the usage of the monitoring scripts. Concerning the configuration parts this document is based on Nagios and Icinga documentation (see "1.4.1 Documentation for Nagios/Icinga, wbemcli, OpenWSMAN Link" on page 9).

1.1 Product Names - Conventions

The FUJITSU Server Plug-ins and the FUJITSU Server Tools can be used for Nagios, Icinga V1.x, SM-Box or other Nagios Core variants. Therefore, in this document all these product variants are subsumed under the name "Nagios/Icinga".

ATTENTION: – Changes in Icinga2:

The developers of Icinga2 changed various interfaces. Therefore, the sample configurations for the FUJITSU Server Plug-ins cannot be used for Icinga2.

Any server where ServerView SNMP-agents or ServerView CIM Providers are installed will be sorted in this description under "PRIMERGY server".

1.2 Target Groups and Purpose of this Manual

This manual is intended for system administrators who already have a basic knowledge of installing and using Nagios/Icinga.

The manual provides instructions on how to use the FUJITSU Server Plug-ins and the FUJITSU Server Tools for Nagios/Icinga.

1.3 Change History

Date	Version	Comment
2012-12	1.0	Start of this document.
2013-03	1.10	RAID support, script extensions.
2013-08	1.20	Script extensions for IPv6 and performance values for file systems and network interfaces.
2013-12	2.00	Monitoring of update agent status, support of SNMP trap configuration files for snmptt, new plug-in for support of ESXi CIM provider via CIM-XML protocol.
2014-04	2.10	Support of ServerView CIM providers on any server, new tool script for the CIM access tests of servers
2014-08	3.00	new script for discovery of Servers (check of server types and generation of configurations), support in monitoring via iRMC for iRMC S4 with firmware version V7.32 or higher

1.4 Documentation

1.4.1 Documentation for Nagios/Icinga, wbemcli, OpenWSMAN Link

Reference	Document Title / Remarks	Source
[1]	Nagios Documentation	http://www.nagios.org
[2]	Icinga Version x.x Documentation	http://docs.icinga.org/
[3]	wbemcli Documentation	http://sblim.sourceforge.net/
[4]	OpenWSMAN Link Documentation	http://github.com/Openwsman/openwsman/wiki

1.4.2 Documentation for ServerView Suite

The documentation can be downloaded free of charge from the Internet. You will find the online documentation at <http://manuals.ts.fujitsu.com> under the link *Industry standard servers - Software*.

For an overview of the documentation available under *ServerView Suite* as well as the filing structure, see the *ServerView Suite* sitemap (*ServerView Suite – Site Overview*).

In this document a manual of note is the description of the *ServerView* MIBs and their events:

Reference	Document Title / Remarks	Source
[5]	PRIMERGY Server Events x.x - User Guide It can be found under: x86 Servers – Software – ServerView Suite – Operation – Event Management	manuals.ts.fujitsu.com
[6]	ServerView System Monitor - User Guide It can be found under: x86 Servers – Software – ServerView Suite – Operation – ServerView System Monitor	manuals.ts.fujitsu.com

1.5 Licensing the Software

1.5.1 License for ServerView Suite

The FUJITSU Server Plug-ins and the FUJITSU Server Tool are subject to the general terms and conditions of business from the software usage and service agreement with FUJITSU Technology Solutions. The *ServerView Suite* is a licensed software component of the PRIMERGY server hardware. It may only be used with hardware, software, or services from FUJITSU Technology Solutions. The terms of the end user license agreement (EULA) for the *ServerView Suite*, regulating copying and distribution of the software to third parties, must be complied with. Further information can be found in the EULA on the *ServerView* DVD.

1.5.2 License for Nagios/Icinga

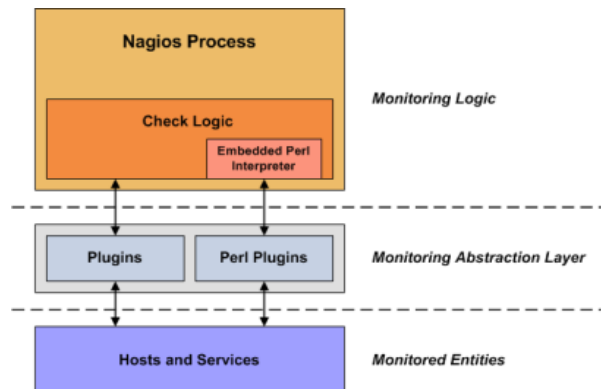
Nagios is licensed under the terms of the GNU General Public License Version 2 as published by the Free Software Foundation. This gives you legal permission to copy, distribute, and/or modify Nagios under certain conditions. Read the 'LICENSE' file in the Nagios distribution or read the online version of the license (<http://www.nagios.org>) for more details.

Icinga is licensed under the terms of the GNU General Public License Version 2 as published by the Free Software Foundation. This gives you legal permission to copy, distribute, and/or modify Icinga under certain conditions. Read the 'LICENSE' file in the Icinga distribution or read the online version of the license (<https://www.icinga.org>) for more details.

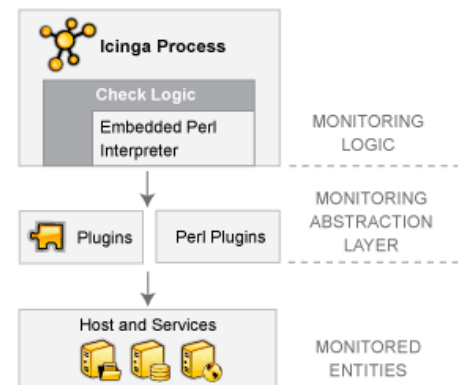
2 Solution Overview

2.1 Nagios versus Icinga

In 2002, Ethan Galstad renamed his existing project “Nagios”. Nagios monitors hosts and services on your network. Actual host and service checks are performed by separate plugins which return the host or service status to Nagios. As the architecture of Nagios grew, some people became dissatisfied with direction and progression of its further development – therefore, in 2009 Icinga was created as a fork of Nagios.



(Source: www.nagios.org)



(Source: docs.icinga.org)

Due to its nature as a fork, Icinga offers all features of Nagios with some additions. Icinga also maintains configuration and plugin compatibility with Nagios to allow for migration.

2.2 Supported Features and Capabilities – Monitored Information

The FUJITSU Server Plug-ins are specially designed to monitor FUJITSU servers. They can be used as Nagios plugins within Nagios, Icinga V1.x series, SM-Box, or other Nagios Core variants. With these plugins you can see an overall hardware status of your PRIMERGY servers as well as in-depth details such as power consumption, temperatures, fan speed, and hardware issues.

For SNMP and CIM following information is available:

- Environment (fans and temperature sensors)
- Power supply and power consumption
- Systemboard parts like voltage, CPU, memory modules

(On some systems not all component information is available.)

Following component information depends on system type:

- DriverMonitor status
- RAID status
- Update agent status

Via PRIMERGY management blade following sub-blade types can be monitored (only SNMP):

- Server blades
- IO-connection blades (e.g. the switch variants)
- KeyVideoMouse blades
- Storage blades

SNMP performance data: The FUJITSU Server Plug-in supports performance values and their thresholds if they are available in corresponding SNMP data.

- PRIMEQUEST MMB, PRIMERGY Blade MMB and PRIMERGY server temperature sensor (value and threshold)
- PRIMEQUEST power consumption (value and max-value)
- PRIMERGY blade MMB power consumption (value)

- PRIMERGY server power consumption (value and threshold)
- PRIMERGY server "Physical Memory Usage" – special: the thresholds can be set as simple percent options
- PRIMERGY server "File System" – special: the thresholds can be set as simple percent options
- PRIMERGY server "Network Interface" – special: the thresholds can be set as simple KB/sec values

CIM performance data:

- Temperature sensor (value and threshold)
- Server power consumption (value and threshold if available)

2.3 CIM Support: Supported Servers

CIM support is available for PRIMERGY servers with ServerView agents and ServerView CIM providers installed.

The following servers can be monitored with **check_fujitsu_server_CIM.pl**:

- Servers where ServerView CIM providers are available via iRMC connection (iRMC firmware version 7.32 or higher)
- Linux or Windows with ServerView SNMP agents and ServerView CIM providers V6.30 (or higher)
- VMware ESXi 5.x with ServerView SNMP agents and ServerView CIM providers V6.21 (or higher)
- VMware ESX 4.1 with ServerView SNMP agents and ServerView CIM providers V6.20

For CIM usage wbemcli or OpenWSMAN can be used as "clients".

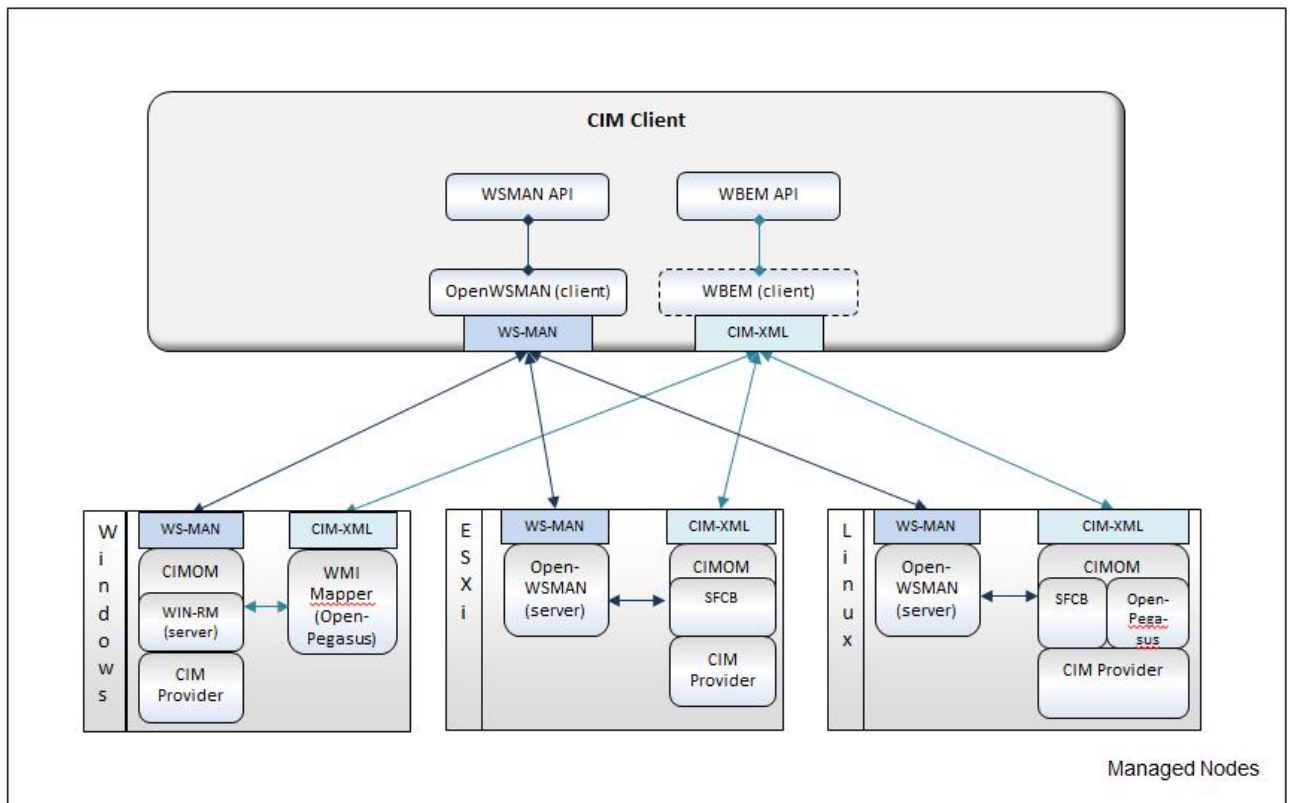


Figure 1: Protocol stacks

2.3.1 Wbemcli – CIM-XML protocol

The command *wbemcli* is used to access the CIMOM CIM-XML service. ESXi systems come with Small Footprint CIM Broker (SFCB) as the CIMOM, which is usually already configured such that it can be accessed via basic authentication (user and password) on port 5989 (*https*) - access to port 5988 for *http* is disabled by default. Other configurations are possible. For a general description see e.g. *Sfcb The Book*.

With OpenPegasus an alternative CIMOM CIM-XML service is available. This service can run on the same port numbers as *sfcf*.

Please note: *wbemcli* and *sfcf* resp. *OpenPegasus* are used as they are. For any problems to run, configure or use these tools, please refer to the originator.

This Plug-in relies on a configuration which is already set up for remote access via *wbemcli* executed on a Nagios Core system.

You can check if **wbemcli** can access your system correctly by executing the following command lines:

```
wbemcli ei -nl -t -noverify 'https://root:<password>@<host>:5989/root/svs:CIM_ComputerSystem'
```

With this call you can also check if the given user and password authentication are accepted by the target CIM service.

Call **wbemcli** to get a SVS_PGY class to test if the ServerView CIM provider information is available:

```
wbemcli ei -nl -t -noverify 'https://root:<password>@<host>:5989/root/svs:SVS_PGYComputerSystem'
```

2.3.2 OpenWSMAN Perl Binding – WS-MAN protocol

The Perl binding of OpenWSMAN is used to communicate with WS-MAN protocol services like OpenWSMAN service on ESXi or LINUX or Windows Remote Management (WinRM) on Windows.

The default ports are 5985 for *http* calls and 5986 for *https* calls.

The WinRM "Listener" must be configured and activated. It is required that WinRM listener is configured to enable basic authentication (user and password).

You can check if OpenWSMAN can access your system correctly by executing the following command lines:

```
wsman identify -h host-P 5985 -u administrator -p password -y basic -V -v
```

OR for SSL usage:

```
wsman identify -b https://host:5986 -u administrator -p password -y basic -V -v
```

Call *wsman* to get a SVS_PGY class to test if the ServerView CIM Provider information is available.

```
wsman enumerate  
http://schemas.microsoft.com/wbem/wsman/1/wmi/root/svs/SVS_PGYComputerSystem -h host -P  
5985 -u administrator -p password -y basic -V -v
```

With this call you can also check if the given user and password authentication is accepted by the target CIM service.

2.4 SNMP Support: Supported Servers and SNMP MIBs Used

The following servers can be monitored with **check_fujitsu_server.pl** (for further information on the SNMP MIBs see 1.4.2 "Documentation for ServerView Suite" on page 9):

- Servers where ServerView SNMP agents are installed – meaning PRIMERGY servers or others.
The following SNMP MIBs are used:
 - SC2.mib
 - Status.mib
 - INVENT.mib
 - RAID.mib
 - SVUpdate.mib

- FUJITSU multi-node servers where ServerView SNMP agents are installed.
The following SNMP MIBs are used:
 - SC2.mib
 - Status.mib
 - INVENT.mib
- PRIMERGY blades – The firmware contains an agent for the following SNMP MIB:
 - S31.mib
- PRIMEQUEST – The firmware contains an agent for the following SNMP MIBs:
 - MMB-COM-MIB.mib and PSA-COM-MIB.mib
- Servers where RAID is used independently on the ServerView SNMP agent:
 - RAID.mib

For all servers to be monitored, some useful administrative information is collected via the RFC1213.mib. This is used for long service output in case of a non-OK status.

Supported ServerView SNMP agent version: V5 or higher.

2.5 Monitoring via iRMC

There are three sources of information to be monitored which can be fetched using the iRMC address for connection:

- iRMC/ IPMI data independent on each installation on the server to be monitored
- Additional information from the ServerView Agent installed on the server to be monitored
- Additional information from the ServerView Agentless Service installed on the server to be monitored

With the FUJITSU Server Plug-ins a support is enabled for iRMC S4 with firmware version V7.32 or higher.

The amount of available data depends on the version of the firmware and on the existence of additional information from ServerView Agent or ServerView Agentless Service.

ATTENTION:

By default SNMP is disabled on iRMC.

You have to enable SNMP on iRMC to monitor the server using the iRMC address and SNMP.

CIM is enabled by default.

3 Installation and Configuration

3.1 Sources

You can obtain the FUJITSU Server Plug-ins and the FUJITSU Server Tools in the following ways:

- Download from the ServerView Suite DVD (version 11.14.02 and later)
- Download from the FUJITSU website:
http://support.ts.fujitsu.com/prim_supported/SVSSoftware
ServerView – Integration Solutions
- Download from the Monitoring Exchange website:
<https://www.monitoringexchange.org>
Check Plugins – Hardware – Server (Manufacturer) – Fujitsu
- Download from the Exchange Nagios website:
exchange.nagios.org
Below *Project Categories* select *Plugins - Hardware - Server Hardware - Fujitsu - check_fujitsu_server.pl*
(or directly call exchange.nagios.org/directory/Plugins/Hardware/Server-Hardware/Fujitsu)

3.2 Requirements

- The FUJITSU Server Plug-ins are specially designed to monitor FUJITSU servers. They can be used as Nagios plugins within Nagios, Icinga, SM-Box, or other Nagios Core variants.
- SNMP:
On the Nagios/Icinga server, **Perl Net::SNMP** must be available.
For IPv6: Perl Net::SNMP V5.2 or higher must be available together with Perl Socket6 V0.23 or higher.
- SNMP:
The PRIMERGY servers to be monitored must have ServerView SNMP agents installed.
Supported ServerView SNMP agent version: V5 or higher. Monitoring of ServerView update status is only available for ServerView agent V6.20 or higher.
These SNMP agents must be configured for read access via SNMP by the Nagios/Icinga server.
- CIM:
For CIM usage wbemcli or OpenWSMAN can be used:
 - **sblim-wbemcli**
The WBEM command line interface (CLI) is a standalone, convenient, systems management utility for CIMOM access which does not require a CIM client library.

RHEL V6.x: included in distribution
RHEL V5 and other operating systems: not included but build version can be downloaded from:
<http://download.opensuse.org/repositories/systemsmanagement/wbem>
 - **OpenWSMAN**
OpenWSMAN supports client and server features. For the Nagios-Core usage the so-called "client" installation should be installed.

Required packages for the plug-in usage:

libwsman1, wsmancli, openwsman-perl
OpenWSMAN Link:
<https://github.com/Openwsman/openwsman/wiki>

OpenWSMAN Link:
<http://download.opensuse.org/repositories/Openwsman>

Additional installation and configuration hints see:
<http://en.community.de.com/techcenter/systems-management/w/wiki/4139.web-services-management-wsman-linux-client-installation-setup.aspx>

3.3 Installation Instructions

1. To add the FUJITSU Server Plug-ins to your Nagios installation, find **resource.cfg** within the Nagios/Icinga installation.
In this the path the variable **\$USER1\$** is defined as the path to your plugin directory.

2. Copy the FUJITSU Server Plug-ins into this path, e.g.

```
cp <somepath>/check_fujitsu_server.pl /usr/local/nagios/libexec
```

3. Change the file permissions (the same access rights as the other plugins), e.g.

```
cd /usr/local/nagios/libexec
chown nagios check_fujitsu_server.pl
chgrp nagios check_fujitsu_server.pl
chmod 755 check_fujitsu_server.pl
```

4. Check with:

```
./check_fujitsu_server.pl -H <hostname_or_address> -C <read_community_string>
```

The above command should not return any errors.

3.4 Nagios Core Sample Configurations

These are sample configurations which can be copied or imported into the existing configurations. Not all Nagios-Core systems monitor all FUJITSU server types, so the administrator should be free to integrate only the required parts.

ATTENTION: – Changes in Icinga2:

The developers of Icinga2 changed various interfaces. Therefore, the sample configurations for the FUJITSU Server Plug-ins cannot be used for Icinga2.

3.4.1 Preliminaries (SNMP)

- To make the Nagios configurations, search the main configuration file of Nagios or Icinga

Icinga sample:

```
ps -edalf | fgrep 'icinga.cfg'
```

This shows the Nagios daemon process and, as a parameter, the main configuration file used.

- In the Nagios main configuration file, check the **cfg_file** or **cfg_dir** directives.

There are two ways to do this: Expand existing configurations or add a new directory for the new configurations.

- The following description assumes that in **\$USER1\$** is the path of the plugins.

Please note:

Check the access rights – they should have the same rights as other configuration files.

If configuration tools such as NagiosQL are in use, the configurations must be imported.

(NagiosQL cannot handle decentralized configurations well (seen in tests).)

In this description some predefined **Nagios/Icinga configurations** are used. These are shown in **blue**. If there are none, other similar definitions should be used.

3.4.2 Host Group Definitions

There are host groups with services attached which print collected summary information (status, relevant component names and performance data). (E.g. a collection of environment, power supply and system board status values).

There are 'detail' host groups which are of interest if one or more servers is to have a monitoring service for each component group instead one overall summary service (e.g. component groups are 'environment', 'power', 'systemboard', etc.).

The 'services' described above are assigned to these host groups.

The customer can choose which host group the specific host should be assigned to.

Of course a single host can be assigned directly to a service.

3.4.2.1 *SNMP - All Fujitsu Servers*

```
define hostgroup {
    hostgroup_name      all-fujitsu-servers
    alias               Fujitsu Servers
    hostgroup_members   primequest-servers,primequest-detail-
servers,primergy-blade-servers,primergy-blade-detail-servers,primergy-blade-with-
KVM,primergy-blade-with-Storage,primergy-detail-servers,primergy-servers-with-
drvmonitor,primergy-servers
    register            1
}
```

3.4.2.2 *SNMP - PRIMERGY Server Variants*

```
define hostgroup {
    hostgroup_name      primergy-servers
    alias               Fujitsu PRIMERGY Servers
    register            1
}

define hostgroup {
    hostgroup_name      primergy-detail-servers
    alias               Fujitsu PRIMERGY Servers
    register            1
}

define hostgroup {
    hostgroup_name      primergy-with-drvmonitor
    alias               Fujitsu PRIMERGY Servers with Driver-Monitor
    register            1
}
```

3.4.2.3 *SNMP - PRIMERGY Blade Variants*

```
define hostgroup {
    hostgroup_name      primergy-blade-servers
    alias               Fujitsu BX Management Blades
    register            1
}

define hostgroup {
    hostgroup_name      primergy-blade-detail-servers
    alias               Fujitsu BX Management Blades
    register            1
}
```



```

define hostgroup {
    hostgroup_name    primergy-blade-with-KVM
    alias             Fujitsu BX Management Blades with KVM Blades
    register           1
}

define hostgroup {
    hostgroup_name    primergy-blade-with-storage
    alias             Fujitsu BX Management Blades with Storage Blades
    register           1
}

```

3.4.2.4 *SNMP - PRIMEQUEST Variants*

```

define hostgroup {
    hostgroup_name    primequest-servers
    alias             Fujitsu PRIMEQUEST Servers
    register           1
}

define hostgroup {
    hostgroup_name    primequest-detail-servers
    alias             Fujitsu PRIMEQUEST Servers
    register           1
}

```

3.4.2.5 *CIM - All Fujitsu Servers*

```

define hostgroup {
    hostgroup_name    all-fujitsu-servers-CIM
    alias             Fujitsu Servers
    hostgroup_members primergy-servers-CIM,primergy-servers-CIM-
component-group
    register           1
}

```

3.4.2.6 *CIM - PRIMERGY Server Variants*

```

define hostgroup {
    hostgroup_name    primergy-servers-CIM
    alias             Fujitsu PRIMERGY Servers CIM Monitoring
    register           1
}

define hostgroup {
    hostgroup_name    primergy-servers-CIM-component-group
    alias             Fujitsu PRIMERGY Servers CIM Component Group
    Monitoring
    register           1
}

```

3.4.3 Command Definitions Usable for these Host Groups

To use NAGIOS plugins for each CLI script or program, one or more 'commands' should be defined.

3.4.3.1 Commands for SNMP Plug-in (*check_fujitsu_server.pl*)

```
define command {
    command_name          check_fujitsu_server
    command_line          $USER1$/check_fujitsu_server.pl -H $HOST ADDRESS$ $_HOSTSV_OPTIONS$ $ARG1$
    register              1
}
```

The above-named command could be enough for the use for all services. However, additional commands can be used if the blade or PRIMEQUEST-specific option is to be set beforehand:

```
define command {
    command_name          check_primergy
    command_line          $USER1$/check_fujitsu_server.pl -H $HOST ADDRESS$ $_HOSTSV_OPTIONS$ $ARG1$
    register              1
}
```

```
define command {
    command_name          check_blade
    command_line          $USER1$/check_fujitsu_server.pl -H $HOST ADDRESS$ --blade $_HOSTSV_OPTIONS$
    register              1
}
```

```
define command {
    command_name          check_primequest
    command_line          $USER1$/check_fujitsu_server.pl -H $HOST ADDRESS$ --primequest
    register              1
}
```

ATTENTION: – No default SNMP community setting:

The default of the plugin is the SNMP community **public**. To enable different communities or other authentications à la SNMPv3 and SNMP ports different from the default 161, the macro `$_HOSTSV_OPTIONS$` is used. Specify `_SV_OPTIONS` in the host definition with the following syntax (as a sample with default values):

```
_SV_OPTIONS -C public -p 161
```

3.4.3.2 Commands for CIM Plug-in (*check_fujitsu_server_CIM.pl*)

```
define command {
    command_name          check_fujitsu_server_CIM
    command_line          $USER1$/check_fujitsu_server_CIM.pl -H
    register              1
}
```

ATTENTION: – This call needs user/password authentications. To enable this, use the macro `$_HOSTSV_CIM_OPTIONS$`.

Specify `_SV_CIM_OPTIONS` in the host definition with the following syntax:
`_SV_CIM_OPTIONS -I <file_with_userpassword_options_inside>`

3.4.4 Service Definitions Usable for these Host Groups

Services can be defined around command definitions. Each service must/should be assigned to a host or host groups.

In this description the predefined service-templates `generic-service` and `perfddata-service` are used. In each Nagios variant there are service templates where the call frequency of scheduled calls and event handling and, in the case of performance data, the connection to tools to show graphs for the performance data, are assigned.

(See PnpNagios as a very often used performance tool.)

3.4.4.1 SNMP - PRIMERGY Server

```
define service {
    hostgroup_name      primergy-servers
    service_description  SV Server
    servicegroups        serverview
    use                  generic-service,perfddata-service
    check_command         check_primergy
    flap_detection_enabled 0
    register             1
}
```

User interface sample in Icinga Classic V1.5:

EC200S2C		OK	10-17-2012 13:30:24	27d 22h 52m 35s	1/3		
SV Server		OK	10-17-2012 13:27:43	9d 4h 31m 28s	1/3	OK - ID=YKXT012345 - Environment(ok) PowerSupply(ok) Systemboard(ok)	
SV UpTime		OK	10-17-2012 13:36:27	6d 8h 30m 50s	1/3	OK UpTime = 6 days, 08:32:05.43	

```
define service {
    hostgroup_name      primergy-detail-servers
    service_description  SV Environment
    servicegroups        serverview
    use                  generic-service,perfddata-service
    check_command         check_fujitsu_server!--chkenv
    flap_detection_enabled 0
    register             1
}

define service {
    hostgroup_name      primergy-detail-servers
    service_description  SV Power
    servicegroups        serverview
    use                  generic-service,perfddata-service
    check_command         check_fujitsu_server!--chkpower
    flap_detection_enabled 0
    register             1
}
```

```

define service {
    hostgroup_name      primergy-detail-servers
    service_description SV System
    servicegroups
    use                 generic-service
    check_command       check_fujitsu_server! --chkssystem
    flap_detection_enabled 0
    register            1
}

define service {
    hostgroup_name      primergy-with-drvmmonitor
    service_description SV Driver Monitor
    servicegroups
    use                 generic-service
    check_command       check_primergy! --chkdrvmonitor
    flap_detection_enabled 0
    register            1
}

```

User interface sample in Icinga Classic V1.5:

TX120S22		OK	10-17-2012 13:37:14	8d 4h 22m 0s	1/3		
	SV Driver Monitor	OK	10-17-2012 13:39:16	7d 23h 57m 36s	1/3	OK DrvMonitor(ok)	
	SV Environment	OK	10-17-2012 13:31:18	8d 4h 28m 31s	1/3	OK Environment(ok)	
	SV Memory Usage	OK	10-17-2012 13:37:20	6d 12h 53m 49s	1/3	OK Physical-Memory=88% Physical-Memory=1814MB Virtual-Memory=1846MB	
	SV Power	OK	10-17-2012 13:35:22	8d 4h 28m 52s	1/3	OK PowerSupply(ok)	
	SV System	WARNING	10-17-2012 13:26:02	8d 4h 29m 3s	3/3	WARNING - ID=YKHINT500101 - MassStorage(degraded) Systemboard(ok)	
	SV UpTime	OK	10-17-2012 13:39:21	8d 4h 29m 22s	1/3	OK UpTime = 25 days, 22:17:06.00	

3.4.4.2 SNMP - PRIMERGY Blade










```

define service {
    hostgroup_name      primergy-blade-servers
    service_description SV Blades
    servicegroups
    use                 generic-service
    check_command       check_fujitsu_server! --bladeinside
    flap_detection_enabled 0
    register            1
}

define service {
    hostgroup_name      primergy-blade-servers
    service_description SV MMB
    servicegroups
    use                 generic-service,perfdata-service
    check_command       check_blade
    flap_detection_enabled 0
    register            1
}

```

User interface sample in Icinga Classic V1.5:

BX900-5			OK	10-17-2012 13:33:24	27d 22h 13m 41s	1/3	SNMP OK - Temperature (40.15583) / 1 / 1 2015-06-17	
	SV Blades		CRITICAL	10-17-2012 13:40:06	11d 23h 58m 47s	3/3	CRITICAL Server-critical(1)-standby(11) Switch-ok(4) LAN Pass Through Blades-ok(1) Fibre Channel Switch- unkown(2) Storage Blades-standby(2)	
	SV MMB 		CRITICAL	10-17-2012 13:37:25	1d 20h 45m 16s	3/3	CRITICAL - ID=System_223 - Fans(ok) - Temperature(ok) - PowerSupply(degraded) - SystemControl(critical)	
	SV UpTime		OK	10-17-2012 13:34:50	11d 23h 58m 38s	1/3	OK UpTime = 1 day, 20:47:23.96	

```

define service {
    hostgroup_name      primergy-blade-detail-servers
    service_description SV IO Connection Blades
    servicegroups
    use                  serverview
    check_command        generic-service
    flap_detection_enabled check_fujitsu_server! --bladeio
    register             0
}

define service {
    hostgroup_name      primergy-blade-detail-servers
    service_description SV MMB Environment
    servicegroups
    use                  serverview
    check_command        generic-service,perfdata-service
    flap_detection_enabled check_blade! --chkenv
    register             0
}

define service {
    hostgroup_name      primergy-blade-detail-servers
    service_description SV MMB Power
    servicegroups
    use                  serverview
    check_command        generic-service,perfdata-service
    flap_detection_enabled check_blade! --chkpower
    register             0
}

define service {
    hostgroup_name      primergy-blade-detail-servers
    service_description SV MMB System
    servicegroups
    use                  serverview
    check_command        generic-service
    flap_detection_enabled check_blade! --chkssystem
    register             0
}

define service {
    hostgroup_name      primergy-blade-detail-servers
    service_description SV Server Blades
    servicegroups
    use                  serverview
    check_command        generic-service
    flap_detection_enabled check_fujitsu_server! --bladesrv
    register             0
}

```


```

define service {
    hostgroup_name      primergy-blade-with-KVM
    service_description  SV KVM Blades
    servicegroups
    use                  generic-service
    check_command        check_fujitsu_server! --bladekvm
    flap_detection_enabled 0
    register             1
}

define service {
    hostgroup_name      primergy-blade-with-storage
    service_description  SV Storage Blades
    servicegroups
    use                  generic-service
    check_command        check_fujitsu_server! --bladestore
    flap_detection_enabled 0
    register             1
}

```

User interface sample in Icinga Classic V1.5:

Host ^v	Service ^v	Status ^v	Last Check ^v	Duration ^v	Attempt ^v	Status Information	
BX600-3 		OK	10-17-2012 13:38:40	27d 22h 39m 31s	1/3		
	SV IO Connection Blades	OK	10-17-2012 13:42:15	8d 0h 51m 16s	1/3	OK Switch-ok(2)	
	SV KVM Blades	OK	10-17-2012 13:39:57	8d 1h 6m 42s	1/3	OK KVM-ok(1)	
	SV MMB Environment 	OK	10-17-2012 13:38:34	8d 0h 33m 13s	1/3	OK - Fans(ok) - Temperature(ok)	
	SV MMB Power 	OK	10-17-2012 13:42:09	8d 0h 49m 11s	1/3	OK - PowerSupply(ok)	
	SV MMB System	OK	10-17-2012 13:39:33	8d 1h 11m 8s	1/3	OK - ID=00000003	
	SV Server Blades	OK	10-17-2012 13:40:49	8d 0h 45m 6s	1/3	OK Server-ok(1)-standby(5)	
	SV UpTime	OK	10-17-2012 13:42:03	8d 0h 52m 51s	1/3	OK UpTime = 263 days, 01:43:18.98	

3.4.4.3 SNMP - PRIMEQUEST

```

define service {
    hostgroup_name      primequest-servers
    service_description  SV Chassis
    servicegroups
    use                  generic-service,perfddata-service
    check_command        check_primequest
    flap_detection_enabled 0
    register             1
}

define service {
    hostgroup_name      primequest-detail-servers
    service_description  SV Chassis Environment
    servicegroups
    use                  generic-service,perfddata-service
    check_command        check_primequest! --chkenv
    flap_detection_enabled 0
    register             1
}

```

```

define service {
    hostgroup_name      primequest-detail-servers
    service_description SV Chassis Hardware
    servicegroups
    use                 generic-service
    check_command       check_primequest! --chkhardware
    flap_detection_enabled 0
    register            1
}

define service {
    hostgroup_name      primequest-detail-servers
    service_description SV Chassis Power
    servicegroups
    serverview
    use                 generic-service,perfddata-service
    check_command       check_primequest! --chkpower
    flap_detection_enabled 0
    register            1
}

define service {
    hostgroup_name      primequest-detail-servers
    service_description SV Chassis System
    servicegroups
    serverview
    use                 generic-service
    check_command       check_primequest! --chkssystem
    flap_detection_enabled 0
    register            1
}

```

User interface sample in Icinga Classic V1.5:

FTS5			OK	10-17-2012 13:34:17	23d 2h 28m 52s	1/3		
			OK	10-17-2012 13:43:18	23d 4h 9m 21s	1/3		
	SV Chassis Environment		OK	10-17-2012 13:40:20	9d 4h 52m 26s	1/3	OK temperatures(ok) fans(ok)	
	SV Chassis Hardware		CRITICAL	10-17-2012 13:37:27	9d 4h 52m 38s	3/3	CRITICAL voltages(ok) cpus(failed) memory-modules(ok)	
	SV Chassis Power		CRITICAL	10-17-2012 13:36:29	9d 4h 52m 52s	3/3	CRITICAL power-supplies(error)	
	SV Chassis System		CRITICAL	10-17-2012 13:40:33	9d 4h 53m 5s	3/3	CRITICAL ID=1480937002 - All=error - Partitions:-ok(1)-failed(2) - SystemBoard:-ok(1)-failed(3)	
	SV UpTime		OK	10-17-2012 13:38:43	12d 0h 2m 8s	1/3	OK UpTime = 2 days, 00:36:38.05	

3.4.4.4 CIM - PRIMERGY Server

```

define service {
    hostgroup_name      all-fujitsu-servers-CIM
    service_description SV CIM Identify
    servicegroups
    use                 generic-service
    check_command       check_fujitsu_server_CIM! --chkidentify
    flap_detection_enabled 1
    register            1
}

```

```

define service {
    hostgroup_name      primergy-servers-CIM
    service_description  SV Server
    servicegroups        serverview
    use                  generic-service,perfddata-service
    check_command        check_fujitsu_server_CIM
    flap_detection_enabled 0
    register             1
}

```

User interface sample in Icinga Classic V1.7:

Host ▾	Service ▾	Status ▾	Last Check ▾	Duration ▾	Attempt ▾	Status Information	
PDB-ESXi-139	SV CIM Identify	OK	10-16-2013 09:07:00	0d 1h 49m 56s	1/3	OK - Type=ESXi	<input type="checkbox"/>
	SV Server	OK	10-16-2013 09:09:00	1d 21h 56m 25s	1/3	OK - ID=920S3063 - Environment(ok) PowerSupply(notdefined) Systemboard(ok)	<input type="checkbox"/>

```

define service {
    hostgroup_name      primergy-servers-CIM-component-group
    service_description  SV Environment
    servicegroups        serverview
    use                  generic-service,perfddata-service
    check_command        check_fujitsu_server_CIM! --chkenv
    flap_detection_enabled 0
    register             1
}

define service {
    hostgroup_name      primergy-servers-CIM-component-group
    service_description  SV Power
    servicegroups        serverview
    use                  generic-service,perfddata-service
    check_command        check_fujitsu_server_CIM! --chkpower
    flap_detection_enabled 0
    register             1
}

define service {
    hostgroup_name      primergy-servers-CIM-component-group
    service_description  SV System
    servicegroups        serverview
    use                  generic-service
    check_command        check_fujitsu_server_CIM! --chkssystem
    flap_detection_enabled 0
    register             1
}

```

User interface sample in Icinga Classic V1.7:

Host ▾	Service ▾	Status ▾	Last Check ▾	Duration ▾	Attempt ▾	Status Information	
QA3-ESXi-221	SV CIM Identify	OK	10-16-2013 09:13:00	0d 2h 14m 19s	1/3	OK - Type=ESXi	<input type="checkbox"/>
	SV Environment	OK	10-16-2013 09:13:00	0d 2h 4m 19s	1/3	OK - Environment(ok)	<input type="checkbox"/>
	SV Power	OK	10-16-2013 09:13:00	0d 2h 4m 19s	1/3	OK - PowerSupply(ok)	<input type="checkbox"/>
	SV System	OK	10-16-2013 09:13:00	0d 2h 4m 19s	1/3	OK - ID=YLAM000811 - Systemboard(ok)	<input type="checkbox"/>

3.4.5 Service-Dependency Definitions for these Services

If there is a predefined service which checks whether SNMP is active or not, it is **recommended** to make the above - named services dependent on this service. In the following sample the service is **SV SNMP Uptime**.

To do this, the hostgroup **'all-primergy-servers'** should be added to this SNMP check and the following dependencies should be added:

3.4.5.1 SNMP - PRIMERGY Server

```

define servicedependency {
    dependent_service_description    SV Server
    hostgroup_name                  primergy-servers
    service_description              SV SNMP Uptime
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

define servicedependency {
    dependent_service_description    SV System
    hostgroup_name                  primergy-detail-servers
    service_description              SV SNMP Uptime
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

define servicedependency {
    dependent_service_description    SV Environment
    hostgroup_name                  primergy-detail-servers
    service_description              SV SNMP Uptime
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

define servicedependency {
    dependent_service_description    SV Power
    hostgroup_name                  primergy-detail-servers
    service_description              SV SNMP Uptime
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

```

3.4.5.2 SNMP - PRIMERGY Blades

```

define servicedependency {
    dependent_service_description    SV MMB
    hostgroup_name                  primergy-blade-servers
    service_description              SV SNMP Uptime
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

```

```

define servicedependency {
    dependent_service_description
    hostgroup_name
    service_description
    execution_failure_criteria
    notification_failure_criteria
    inherits_parent
}

define servicedependency {
    dependent_service_description
    hostgroup_name
    service_description
    execution_failure_criteria
    notification_failure_criteria
    inherits_parent
}

define servicedependency {
    dependent_service_description
    hostgroup_name
    service_description
    execution_failure_criteria
    notification_failure_criteria
    inherits_parent
}

```

SV MMB System
primergy-blade-detail-servers
SV SNMP Uptime
w,u,c
w,u,c
1

SV MMB Environment
primergy-blade-detail-servers
SV SNMP Uptime
w,u,c
w,u,c
1

SV MMB Power
primergy-blade-detail-servers
SV SNMP Uptime
w,u,c
w,u,c
1

3.4.5.3 SNMP - PRIMEQUEST

```

define servicedependency {
    dependent_service_description
    hostgroup_name
    service_description
    execution_failure_criteria
    notification_failure_criteria
    inherits_parent
}

define servicedependency {
    dependent_service_description
    hostgroup_name
    service_description
    execution_failure_criteria
    notification_failure_criteria
    inherits_parent
}

define servicedependency {
    dependent_service_description
    hostgroup_name
    service_description
    execution_failure_criteria
    notification_failure_criteria
    inherits_parent
}

define servicedependency {
    dependent_service_description
    hostgroup_name
    service_description
    execution_failure_criteria
    notification_failure_criteria
    inherits_parent
}

```

SV Chassis
primequest-servers
SV SNMP Uptime
w,u,c
w,u,c
1

SV Chassis System
primequest-detail-servers
SV SNMP Uptime
w,u,c
w,u,c
1

SV Chassis Environment
primequest-detail-servers
SV SNMP Uptime
w,u,c
w,u,c
1

SV Chassis Power
primequest-detail-servers
SV SNMP Uptime
w,u,c
w,u,c
1

OPTIONAL

```

define servicedependency {
    dependent_service_description    SV Chassis Hardware
    hostgroup_name                  primequest-detail-servers
    service_description              SV SNMP Uptime
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

```

3.4.5.4 CIM - PRIMERGY Server

```

define servicedependency {
    dependent_service_description    SV Server
    hostgroup_name                  primergy-servers-CIM
    service_description              SV CIM Identify
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

define servicedependency {
    dependent_service_description    SV System
    hostgroup_name                  primergy-servers-CIM-component-group
    service_description              SV CIM Identify
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

define servicedependency {
    dependent_service_description    SV Environment
    hostgroup_name                  primergy-servers-CIM-component-group
    service_description              SV CIM Identify
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

define servicedependency {
    dependent_service_description    SV Power
    hostgroup_name                  primergy-servers-CIM-component-group
    service_description              SV CIM Identify
    execution_failure_criteria       w,u,c
    notification_failure_criteria    w,u,c
    inherits_parent                  1
}

```

3.4.6 Host Template for MMB Web Server Address or Option Settings

For PRIMERGY blade servers and PRIMEQUEST servers there is a web address for calling the web server interface for MMB. To add this, a host template can be used:

```

define host {
    name                            mmb-webaddress
    use                             generic-host
    notes_url                       http://$HOSTADDRESS$:80
    register                         0
}

```

```

define host {
    name                primequest-webaddress
    use                 generic-host
    notes_url           http://$HOSTADDRESS$:8081
    register            0
}

```

SNMP - The following host template can be used to specify default or non-default options for the called **check_fujitsu_server** Plug-in:

```

define host {
    name                fujitsu-snmp-defaults
    #hostgroups         all-fujitsu-servers
    use                 generic-host
    _SV_OPTIONS         -Cpublic -p161
    # These are defaults - usable for SNMP2 or SNMP3 options
    register            0
}

```

3.4.7 Sample Host Definitions

These configurations can be used for a host if its definition contains the assignment to the corresponding host group.

3.4.7.1 SNMP Host Samples

Here a sample for a PRIMERGY blade server:

```

define host {
    host_name           BX600-3
    alias               BX600-3
    display_name        BX600-3
    address             xxx.xxx.xxx.xxx
    parents             servware.abg.fsc.net
    hostgroups          primergy-blade-detail-servers
    use                 windows-server, mmb-webserver
    register            1
}

```

Here a sample for a PRIMERGY server (single node):

```

define host {
    host_name           EC200S2C
    alias               PRIMERGY Econel 200 S2
    address             xxx.xxx.xxx.xxx
    parents             servware.abg.fsc.net
    hostgroups          primergy-servers
    use                 windows-server
    register            1
}

```

3.4.7.2 CIM Host Samples

For CIM-XML usage:

```
define host {
    host_name          PDB-ESXi-139
    address            nnn.nnn.nnn.139
    hostgroups         primergy-servers-CIM,linux-servers
    use                linux-server
    #_SV_CIM_OPTIONS   -uroot -p*****
    _SV_CIM_OPTIONS    -I/etc/authent/AuthentPDB.txt
    register           1
}
```

For WS-MAN usage:

```
define host {
    host_name          QA3-WIN-CIM-103
    address            nnn.nnn.nnn.103
    hostgroups         primergy-servers-CIM
    use                windows-server
    SV_CIM_OPTIONS     -UW -P5985 -IAuthentQA3Win.txt
    register           1
}
```

3.4.8 Special Notification Command

The notification commands seen in Icinga V1.5 only send \$SERVICEOUTPUT\$. The script for Fujitsu servers writes detailed information in the \$LONGSERVICEOUTPUT\$. Because of this it is **recommended** that this output is sent along with the summary information from the script in \$SERVICEOUTPUT\$.

Configuration sample:

```
define command {
    command_name        notify-service-by-email-detail
    command_line         /usr/bin/printf "%b" "***** Icinga
*****\n\nNotification Type: $NOTIFICATIONTYPE$\n\nService: $SERVICEDESC$\nHost:
$HOSTALIAS$\nAddress: $HOSTADDRESS$\nState: $SERVICESTATE$\n\nDate/Time:
$LONGDATETIME$\n\nAdditional
Info:\n\n$SERVICEOUTPUT$\n\nDetails:\n\n$LONGSERVICEOUTPUT$\n" | /bin/mail -s "***
$NOTIFICATIONTYPE$ Service Alert: $HOSTALIAS$/ $SERVICEDESC$ is $SERVICESTATE$ ***"
$CONTACTEMAIL$
    register            1
}
```

4 FUJITSU SNMP Server Check Plug-in: check_fujitsu_server.pl

Nagios Core uses command line interface (CLI) programs or scripts to retrieve state and performance data. These so-called Nagios plugins should return four return values for **OK**, **WARNING**, **ERROR** or **UNKNOWN** states and can print data on standard output.

These plugins are called as scheduled by the Nagios daemon. Schedule values can be set via the configuration.

Currently the above-named script uses Perl SNMP abilities. If the managed node does not support corresponding SNMP/MIB values, an error message will be generated.

4.1 Basics

4.1.1 Script Name

Name of the script

check_fujitsu_server.pl

This is a Perl script which uses Net::SNMP calls.

It checks a FUJITSU server using SNMP. Currently this script is able to check PRIMERGY servers (any server where ServerView SNMP Agent is installed and running), PRIMERGY blade servers (BX series,) and PRIMEQUEST servers, as well as PRIMERGY multi-node servers (CX series).

4.1.2 Options around the Plug-in itself

-V|--version

Print version information and usage.

-h|--help

Print help text.

4.1.3 SNMP Options around Addressing

-H|--host=<name-or-ip>

Host address as DNS name or IP address of the server.

[-A|--admin=<adminaddress>]

Specify administration address with which all data for the host should be retrieved. This might be the iRMC address if a corresponding SNMP agent is enabled on iRMC.

HINT: The two addresses are meant for Nagios administrators who want to define one host and Nagios services which should use the origin \$HOSTADDRESS\$ and parallel to these other Nagios services which should use the iRMC address e.g. specified in \$HOSTSV_ADMIN_OPTION\$ or \$HOSTSV_OPTIONS\$

-P|-p|--port=<portnumber>

SNMP port number. Default is 161.

-T|--transport=<type>

SNMP transport domain type. A full description of available values can be found in Net::SNMP->session parameters and parameter '-domain'.

Default is "udp", meaning UDP service for IPv4 addresses.

Samples for other values:

- **tcp** – for TCP connection instead of UDP
- **udp6** or **tcp6** – for IPv6 addresses

ATTENTION: For IPv6 requirements, see sections 3.2 "Requirements" on page 14 and 2.4 "SNMP Support: Supported Servers and SNMP MIBs" on page 12.

-C|--community=<SNMP community string>

SNMP community of the server. Usable for SNMPv1 and SNMPv2. Default is **public**.

-u|--user=<username> [--authpassword=<pwd>] [--authkey=<key>] [--authprot=<prot>] [--privpassword=<pwd>] [--privkey=<key>] [--privprot=<prot>]

SNMPv3 authentication credentials.

-I|--inputfile=<file> [--inputdir=<directory>]

File in which host specific options like the above named ones beside the host option can be stored. The usage of this is **recommended** for security relevant options like -u and the other SNMPv3 credentials.

This file must be readable for the owner of the plug-in script used by the Nagios scheduler out of the Nagios Plug-in directory!

With inputdir a directory path of the input option file can be specified. The directory path is ignored if the input file starts with '/'.

4.1.4 About IPv6 Usage

For IPv6 requirements, see sections 3.2 "Requirements" on page 14 and 2.4 "SNMP Support: Supported Servers and SNMP MIBs" on page 12.

Before using an IPv6 address, make sure that on the host to be monitored there is an SNMP which is able to handle IPv6 addresses:

- Check SNMP configuration settings
- Check firewalls
- Check IPv6 kernel abilities and configurations of the operating system itself

ATTENTION: On SLES10 there is an SNMP which does not fully support IPv6 and is unstable if called with IPv6!

If an IPv6 address is entered, the Plug-in sets automatically sets the transport type to UDP-IPv6 (if not specified).

If an IPv4 address is entered and no '-T' option is set, the default transport type of local Perl Net::SNMP is used.

For DNS names as host address, the default transport type of local Perl Net::SNMP is used.

4.1.5 Plug-in Processing Control Option

-t|--timeout=<timeout in seconds>

Timeout for the script processing.

-v|--verbose=<verbose mode level>

Enable verbose mode (levels: 0,1,2,3). Generates multi-line output with inventory information or other additional information.

4.2 Common Rules for Printouts

This script always prints the **nagios_status** string at the beginning. This status string is printed in upper case letters.

The Nagios status is followed by additional information (data, component and status values).

SNMP or CIM-specific status values are printed in lower case. Most components have more than the four Nagios status values.

Details are printed after the first line – in Nagios this is part of LONG_SERVICE_OUTPUT. This long output contains detailed information which is especially relevant in the case of non-OK responses ("notifications").

It is suggested to send the LONG_SERVICE_OUTPUT along with notifications.

4.2.1 Text Rules for Automatic Scanning of Printouts

The following rules are used for printouts. The samples in this document show this.

Here is a short description for those who want to scan and analyse the text automatically.

- 1st line – the SERVER_OUTPUT line
 - Always starts with
<nagios_status> OK, WARNING, CRITICAL, UNKNOWN

Rules for all lines:

- All status values which are NOT the Nagios status are printed in lower case!

- Key-Value-Pair print:
`<field>=<value>[<unit>]`
- Value Print
 If a value contains blanks then quotation marks are used.
 All other values may have quotations marks or not.
- Status-Counter-Prints
`[<topic>-]<realstatus>(<counter>)-...`

 Sample: CRITICAL Server-critical(1)-standby(11)
 ... Service for checking server blades via MMB

In the subsequent lines, LONG_SERVICE_OUTPUT might be single-line units or multi-line units (table contents):

- For Single Line Unit Prints
`[<title> -] <key-value-pairs> ...`

 Sample:
 AgentInfo - Ident="ServerView ServerControl 2 hardware monitoring agent"
 Version=6.10.01.05 Company="Fujitsu"
- For Multi-Line-Unit Prints – Header
 * <title>
- For Multi-Line-Unit Prints – Rows
`[<status>:] <item>['<index>'] - <key-value-pairs> ...`

 Multi-Line-Sample
 * Temperature Sensors:
 ok: Sensor[1] Name=Ambient Temperature=24C Warning=37C Critical=42C
 ok: Sensor[2] Name=Systemboard Temperature=33C Warning=60C Critical=65C
 ok: Sensor[3] Name=CPU1 Temperature=30C Warning=85C Critical=90C
 ok: Sensor[4] Name=CPU2 Temperature=30C Warning=85C Critical=90C
 ok: Sensor[5] Name=DIMM-1A Temperature=31C Warning=78C Critical=82C
 ok: Sensor[7] Name=DIMM-1B Temperature=32C Warning=78C Critical=82C

 ok: Sensor[11] Name=DIMM-1D Temperature=29C Warning=78C Critical=82C
 ok: Sensor[13] Name=DIMM-1E Temperature=29C Warning=78C Critical=82C

Performance data

- The performance data is printed in a new line after the above-mentioned lines starting with ' | '.
 The syntax is compliant with Nagios Core.

4.2.2 Verbose Level Usage

The Plug-in should print the output as small as possible by default. Any output is stored by Nagios systems in databases.

The standard – Verbose Level 0 – is to print the output as small as necessary.

Verbose Level 1 – Print system information usable by administrators in the case of notifications.

If the Plug-in check has to print a WARNING or CRITICAL status, all available system information is printed automatically.

If this is to be done independently of the status, Verbose Level 1 can be used.

The output depends on the server type – the only common part is the data of the standard RFC1213.mib

- See PRIMERGY server and 4.4.4 "System Information in Case of WARNING or CRITICAL - for Notifications" on page 35.
- See PRIMERGY blade MMB and 4.6.3 "System Information in the Case of WARNING or CRITICAL - for Notifications" on page 44.

- See PRIMEQUEST and 4.8.3 "System Information in the Case of WARNING or CRITICAL - for Notifications" on page 53.

Verbose Level 2 – Print all analyzed data regardless of the status

By default only the summary status is checked and printed. In the case of a non-OK status, the status of each sub-component is checked and the data of the non-OK components is printed.

With Verbose Level 2, all sub-components and their information are checked and printed, regardless of status.

This is of course dependent on the check options used and the server type.

This level can be used (unscheduled) for diagnostics and to analyze check-option-dependent resources of the server.

Verbose Level 3 – Print status-independent information for the "System" check to analyze resources of the server.

This is very much server-type-dependent. With this level, "unit" prints, agent versions and so on can be read.

This level is only used to analyze resources of the server.

- See PRIMERGY server and 4.4.5 "Additional System Information" on page 36.
- See PRIMEQUEST and 4.8.4 "Additional System Board and Partition Information" on page 53.

Verbose Level greater than 3 and less than 100 --- for internal use only.

Verbose Level greater than 100 – Special table printouts.

For some tables there are special numbers used to read and print this data. The available numbers are described in the document sections below. These tables might be useful for analyzing resources and for diagnostics.

4.3 Tool Features

4.3.1 Simple SNMP Uptime Check

Option

--chkuptime

ATTENTION: This option cannot be combined with other check options.

With this option the script checks if SNMP is running and if the local server is allowed to receive SNMP responses from the host to be monitored.

The result is the RFC1213 SNMP uptime of this system.

HINT: If this call fails, check the following potential reasons on the server to be monitored:

- Is the server powered-on and SNMP running
- Check the SNMP community data or SNMPv3 authentication data
- Check other permission or restriction rules on the server to be monitored

4.3.2 Get Only System Information Used for Notify Information

Option

--systeminfo

ATTENTION: This option cannot be combined with other check options.

If one of the following checks returns WARNING or CRITICAL and if Verbose Level 1 is used, system information usable by administrators is collected (if available) and printed.

To achieve this without a check, the above-named option can be used in combination with the PRIMERGY blade options (--blade or --bladeinside) or with the PRIMEQUEST option (--primequest) or with no extra option for PRIMERGY servers or other hosts.

4.4 Check of PRIMERGY Server or Server where ServerView SNMP Agent is Installed

The checks depend on the existence and use of **SC2.mib**. Each server that supports this MIB can be monitored with this script.

HINT: Depending on the server type and version of SC2 support, it may occur that not all data can be retrieved.

ServerView SNMP Agent supports this MIB.

4.4.1 Options

There is no special option to for selecting the type "Primergy Server". This is the default for the script.

By default a global check is performed for the specific server. Single checks can be selected using the following options:

--chkssystem

Checks anything other than "Environment", "Power" or "Driver Monitor".

The system information can be split with the following options.

--chkstorage

Check only "MassStorage" (RAID) data.

This is an option for those who want to monitor only this component.

--chkhardware

"Hardware" (voltage, CPU and memory modules).

This is an option for those who want to monitor only these components.

The system hardware information can be split with the following options.

--chkcpu CPU information

--chkvoltage Voltage information

--chkmemmodule Memory module information

--chkenv

Checks fan and temperature sensors

The environment information can be split using the following options instead of chkenv.

--chkenv-fan Fan

--chkenv-temp Temperature sensor, includes threshold data

--chkpower

Checks: Power supply units and power consumption with threshold data if available.

--chkdrvmonitor

Print driver monitor information. In the case of a non-OK status, the detailed information on the driver monitor will only be visible with this option.

4.4.2 System Status Information

- Summary line
 - Nagios status string
 - Serial number (ID)
 - Status of each subcomponent.
If `chkssystem` is selected, only subcomponents other than "Environment" and "Power" are printed.

HINT: the components mentioned in the first output line are part of *Status.mib*.

If the option `chkssystem` is selected than "Hardware" components assigned to the system board such as CPU, voltage and memory modules, are monitored and if "MassStorage" exists the RAID information is scanned.

For more on these, see the following sections:

- 4.4.11 "Hardware – CPU Information" on page 40
 - 4.4.10 "Hardware – Voltage Information" on page 39
 - 4.4.12 "Hardware – Memory Module Information" on page 40
 - 4.4.13 "System – MassStorage (RAID) Information" on page 41
 - With Verbose Level 1 or in case of Non-OK the **additional** status components are printed, e.g. "Network". Currently there is no detailed information available on these components. If only some of them are not OK, check the server information via the administrative URL.
- Sample
OK - ID=YKHJXXXXXX - MassStorage(ok) SystemBoard(ok)

4.4.3 Special Information for PRIMERGY Multi-Node Servers (CX-series)

If the host is a multi-node server where additional information about the chassis is available, this additional information will be printed in the case of WARNING or CRITICAL before the information described below.

Multi node information:

- Serial number (assigned to chassis)
 - Name of multi node
 - Location (if defined)
 - Contact (if defined)
 - Model
- Sample (Verbose Level 1):
OK - **Chassis-ID=CX420S1014** ID=CX272S1024 - Environment(ok) PowerSupply(ok)
MassStorage(ok) Systemboard(ok) Network(ok) DrvMonitor(ok)
ID=CX272S1024 Systemname=H49025-CX272S1.servware.abg.fsc.net Description="Hardware:
Intel64 Family 6 Model 45 Stepping 7 AT/AT COMPATIBLE - Software: Windows Version
6.2 (Build 9200 Multiprocessor Free)" AdminURL=http://xxx.xxx.xxx.xxx
Model="PRIMERGY CX272 S1"
Multi Node System - ID=CX420S1014 Name=CX420S1 Model="PRIMERGY CX420 S1"

4.4.4 System Information in Case of WARNING or CRITICAL - for Notifications

- Serial identification - Additional information in the case of errors or for Verbose Level 1:
- RFC1213 information - Additional information in the case of errors or for Verbose Level 1:
 - Server system name
 - Description string
 - Location (if defined)
 - Contact (if defined)
- System unit information - Additional information in the case of errors or for Verbose Level 1:
 - Model
 - Administrative URL (if available)

The following is only printed for Verbose Level 2:

 - Server unit name
 - Location (if defined)
 - Contact (if defined)
- Additional inventory information
 - Operating system (OS) and OS revision
 - Fully Qualified DNS Name (FQDN) if available and not identical with server system name

- Notify sample or Verbose Level 1 sample:
 OK - ID=YBWC000000 - Environment(ok) PowerSupply(ok) MassStorage(ok) Systemboard(ok)
 ID=YBWC000000 Systemname=GF-RX300S3 Description="Hardware: EM64T Family 15 Model 6
 Stepping 4 AT/AT COMPATIBLE - Software: Windows Version 5.2 (Build 3790
 Multiprocessor Free)" AdminURL=http://xxx.xxx.xxx.xxx:80 Model="PRIMERGY RX300 S3"
 OS="Windows Server 2003 R2 Enterprise Edition (x64) Service Pack 2" OS-
 Revision="5.2" FQDN="..."

4.4.5 Additional System Information

With the additional option verbose=3, additional information is available for diagnostic or inventory reasons.

SC2 information:

- Status component table
 - SC2 components and their status
 (In the first output line are Status.mib component names which bundle parts of these components)
- Agent information group
 - Identification string
 - Version
 - Company
- Management node table
 - IP Address
 - Name (if not identical with IP address)
 - Controller type (model)
 - Class (if not "unknown" or "primary")
 - MAC
- Server Table
 - UUID
 - Assigned memory
 - Boot status

Sample:

```
...
* Status Component Table:
[1] - Boot(ok) PowerSupplies(ok) Temperatures(ok) Fans(ok) Voltages(ok) CPUs(ok)
MemoryModules(ok)
...
AgentInfo - Ident="ServerView ServerControl 2 hardware monitoring agent"
Version=6.10.01.05 Company="Fujitsu"
* Management Nodes:
Node[1] - IP=xxx.xxx.xxx.xxx ControllerType="Local Area Connection"
MAC=0x000ae48950ee
Node[2] - IP=xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx ControllerType="Local Area
Connection" MAC=0x000ae48950ee
Node[3] - IP=xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx ControllerType="Local Area Connection"
MAC=0x000ae48950ee
Node[4] - IP=xxx.xxx.xxx.xxx ControllerType="iRMC S2" Class=baseboard-controller
MAC=0x000ae48950f0
Node[5] - IP=xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx ControllerType="iRMC S2"
Class=baseboard-controller MAC=0x000ae48950f0
Node[6] - IP=xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx ControllerType="iRMC S2" Class=baseboard-
controller MAC=0x000ae48950f0
* Server Table:
Server[1] - UUID=00000000-0000-0000-0000-000AE48950EE Memory=4096MB BootStatus=os-
running
```

4.4.5.1 How to Detect a Parent Management Blade

To detect the management blade of a server blade, use the above-mentioned --verbose=3.

If the following is part of the output, this 'host' is a server blade and the parent IP address is printed in the management nodes information

IP=... Class=management-blade

4.4.6 Power Supply Information

This is part of the "Power" information (chkpower).

- Summary line

PowerSupplies(<status>)

with summary status of all power supplies.

- Long output in case of WARNING or CRITICAL or verbose=2 output:

For each power supply unit:

- PSU status
- PSU[<idx>] with <idx> as number of power supply unit
- Name
- Current load and maximum load in watts

Some parts might not be available.

- Verbose sample

* Power Supplies:

ok: PSU[1] Name=PSU1 CurrentLoad=128Watt Max=768Watt

4.4.7 Power Consumption Information

This is part of the "Power" information (chkpower).

- Performance data

PowerConsumption=<average>Watt[;<warn>]

- Long output for verbose=2:

- Power consumption status
- Current and maximum value in watts (if specified)
- Warning level in percent (if specified)
- Critical level limit in watts (if specified)

This information is only available if the customer or corresponding tools have enabled the PowerConsumption thresholds. If the <pmstatus> is unknown or disabled, this information is not printed and used!

- Verbose sample

ok: PowerConsumption Current=128Watt Max=200Watt Warning=80%
| PowerConsumption=128Watt;160

4.4.8 Fan Information

This is part of the "Environment" information (chkenv).

- Summary line for the summary monitoring:

Only the combined information on all fans and temperature sensors is available:

Environment(<state>)

- Summary line for the chkfan usage

Fans(<state>)

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each fan:

- Fan status
- Fan[<idx>] with <idx> as number of fan
- Name
- Speed in rpm (rotations per minute)

- Verbose sample

* Fans:

```
ok: Fan[1] Name=FAN1_SYS Speed=1980rpm
ok: Fan[2] Name=FAN2_SYS Speed=2040rpm
ok: Fan[3] Name=FAN_PSU Speed=1440rpm
```

- User interface sample in Icinga Classic 1.5:

Service State Information

Current Status:	WARNING (for 1d 12h 53m 50s)
Status Information:	WARNING Environment(degraded) Systemname=TX150S42 Description="Hardware: x86 Family 15 Model 4 Stepping 1 AT/AT COMPATIBLE - Software: Windows Version 5.2 (Build 3790 Multiprocessor Free)" Location=" " Contact=" " Model="PRIMERGY TX150 S4" AdminURL=http:// redundant-fan-failed: Fan[2] Name=FAN1_PSI redundant-fan-failed: Fan[3] Name=FAN2_PSI
Performance Data:	Systemboard=29C;60;65 CPU=35C;80;85 Ambient=23C;38;43
Current Attempt:	3/3 (HARD state)
Last Check Time:	10-17-2012 14:07:25
Check Type:	ACTIVE
Check Latency / Duration:	0.155 / 3.959 seconds
Next Scheduled Check:	10-17-2012 14:17:25
Last State Change:	10-16-2012 01:17:25
Last Notification:	10-17-2012 13:47:38 (notification 166)
Is This Service Flapping?	N/A
In Scheduled Downtime?	NO
Last Update:	10-17-2012 14:11:08 (0d 0h 0m 7s ago)

4.4.9 Temperature Sensor Information

This is part of the "Environment" information (chkenv).

- Summary line for the summary monitoring:

Only the combined information on all fans and temperature sensors is available:
Environment(<state>)

- Summary line for the chktemp usage:

Temperatures(<state>)

- Long output in case of WARNING or CRITICAL or verbose=2 output:

For each sensor:

- Temperature sensor status
 - Sensor[<idx>] with <idx> as number of sensor
 - Name
 - Current value and WARNING level and CRITICAL level in Celsius
- Sensors with the status "not-available" will be ignored.

- Performance data:

<identification of sensor>=<current>C;<warn>;<critical>

- Verbose sample:
 - * Temperature Sensors:
 - ok: Sensor[1] Name=Ambient Temperature=24C Warning=37C Critical=42C
 - ok: Sensor[2] Name=Systemboard Temperature=33C Warning=60C Critical=65C
 - ok: Sensor[3] Name=CPU1 Temperature=30C Warning=85C Critical=90C
 - ok: Sensor[4] Name=CPU2 Temperature=30C Warning=85C Critical=90C
 - ok: Sensor[5] Name=DIMM-1A Temperature=31C Warning=78C Critical=82C
 - ok: Sensor[7] Name=DIMM-1B Temperature=32C Warning=78C Critical=82C
 - ok: Sensor[11] Name=DIMM-1D Temperature=29C Warning=78C Critical=82C
 - ok: Sensor[13] Name=DIMM-1E Temperature=29C Warning=78C Critical=82C
 - | Ambient=24C;37;42 Systemboard=33C;60;65 CPU1=30C;85;90 CPU2=30C;85;90 DIMM-1A=31C;78;82 DIMM-1B=32C;78;82 DIMM-1D=29C;78;82 DIMM-1E=29C;78;82

4.4.10 Hardware – Voltage Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkssystem:
 - ... **SystemBoard(<combinedstatus>)** ...
 - The status includes at least all CPUs, voltages and memory modules.
- Summary line for chkhardware or chkvoltage:
 - ... **Voltages(<all_voltage_unit_Status>)** ...
- Performance data
 - ATTENTION:** Currently no performance data is generated for the Nagios tool.
This data is only visible with verbose data.
- Long output in the case of WARNING or CRITICAL or verbose=2 output:
 - For each voltage unit:
 - Voltage status
 - Voltage[idx] with <idx> as number of voltage
 - Name
 - Current, minimum and maximum values in mV
- Verbose sample

```
...
* Voltages:
ok: Voltage[1] Name=Main_+12V Current=12000mV Min=11050mV Max=13000mV
ok: Voltage[2] Name=Main_-12V Current=4294955246mV Min=-13760mV Max=-10530mV
ok: Voltage[3] Name=Main_+5V Current=5070mV Min=4650mV Max=5410mV
ok: Voltage[4] Name=Main_+3.3V Current=3360mV Min=3050mV Max=3570mV
ok: Voltage[5] Name=Battery_+3V Current=3100mV Min=2010mV Max=3500mV
ok: Voltage[6] Name=AUX_1.8V Current=1750mV Min=1710mV Max=1890mV
ok: Voltage[7] Name=AUX_1.2V Current=1190mV Min=1120mV Max=1280mV
ok: Voltage[8] Name=LAN_1.2V Current=1190mV Min=1120mV Max=1280mV
ok: Voltage[9] Name=Stdbby_+3.3V Current=3310mV Min=3090mV Max=3570mV
ok: Voltage[10] Name=Mem1_+1.5V Current=1520mV Min=1400mV Max=1600mV
ok: Voltage[11] Name=Mem1_+1.8V Current=1820mV Min=1670mV Max=1930mV
not-available: Voltage[12] Name=Mem2_+1.5V Min=1400mV Max=1600mV
not-available: Voltage[13] Name=Mem2_+1.8V Min=1670mV Max=1930mV
ok: Voltage[14] Name=Core_+1.5V Current=1540mV Min=1400mV Max=1600mV
ok: Voltage[15] Name=1.8V Current=1800mV Min=1670mV Max=1920mV
ok: Voltage[16] Name=VTT_BMC Current=880mV Min=800mV Max=1000mV
```

4.4.11 Hardware – CPU Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkssystem:
... SystemBoard(<combinedstatus>) ...
The status includes at least all CPUs, voltages and memory modules.
- Summary line for chkhardware or chkcpu:
...CPUs(<cpus_status>)...
- Long output in the case of WARNING or CRITICAL or verbose=2 output:
For each CPU:
 - CPU status
 - CPU[<idx>] with <idx> as number of CPU
 - Name
 - Model
 - Speed in MHz
- Verbose sample
* CPU Table:
ok: CPU[1] Name=CPU_1 Model="CPU 1" Speed=2133MHz
ok: CPU[2] Name=CPU_2 Model="CPU 2" Speed=2133MHz

4.4.12 Hardware – Memory Module Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkssystem:
... SystemBoard(<combinedstatus>) ...
The status includes at least all CPUs, voltages and memory modules.
- Summary line for chkhardware or chkmemmodule:
...MemoryModules(<status>)...
- Long output in the case of WARNING or CRITICAL or verbose=2 output
For each memory module:
 - Memory status
 - Memory[<idx>] with <idx> as number of memory module
 - Name
 - Type
 - Capacity in Mbytes
 - Frequency and maximum frequency in MHzMemory modules with the SNMP status "not-present" will be ignored.
- Verbose sample
* Memory Modules Table:
ok: Memory[1] Name=DIMM-1A Type="DDR2 / RDIMM" Capacity=1024MB Frequency=533MHz
Frequency-Max=667
ok: Memory[2] Name=DIMM-1B Type="DDR2 / RDIMM" Capacity=1024MB Frequency=533MHz
Frequency-Max=667

- User interface sample in Icinga Classic 1.5

Service State Information

Current Status:	CRITICAL (for 11d 23h 54m 48s)
Status Information:	CRITICAL - ID=YLAM009999 - MassStorage(ok) Systemboard(error) Systemname=H5019-CX122S1M.servware.abg.fsc.net Description="Hardware: Intel64 Family 6 Model 44 Stepping 2 AT/AT COMPATIBLE - Software: Windows Version 6.2 (Build 9200 Multiprocessor Free)" Location=" " Contact=" " Model="PRIMERGY CX122 S1" AdminURL=http:// :80 failed: Memory[4] Name=DIMM-1B Type="DDR3 / UDIMM" Capacity=2048MB Frequency=1066MHz Frequency-Max=1333MHz
Performance Data:	
Current Attempt:	3/3 (HARD state)
Last Check Time:	10-17-2012 13:59:58
Check Type:	ACTIVE
Check Latency / Duration:	0.211 / 2.639 seconds
Next Scheduled Check:	10-17-2012 14:09:58
Last State Change:	10-05-2012 14:11:06
Last Notification:	10-17-2012 13:24:29 (notification 280)
Is This Service Flapping?	N/A
In Scheduled Downtime?	NO
Last Update:	10-17-2012 14:05:48 (0d 0h 0m 6s ago)

4.4.13 System – MassStorage (RAID) Information

This is part of the "System" information (chkssystem). It is dependent on the existence of the subcomponent "MassStorage".

- Summary line:

...**MassStorage(<status>)**...

- Long output in the case of WARNING or CRITICAL or verbose=2 output

RAID overall component status

- RAID status
- Status for each subcomponent (controller, physical device and logical drive))

For each RAID controller:

- Controller status
- RAIDCtrl[<idx>] with <idx> as number of controller
- Name
- Description
- Host name
- BBU status
- Model

For verbose only:

- Cache in Mbytes
- Interface
- Driver

For each RAID physical device:

- Smart status (if enabled)
- Device status
- PhysicalDevice[] – no simple index number because there is a only a combination of four numbers available
- Name
- Serial number (ID)

- Error count
- Bad block count
- Enclosure number (if available)
- Slot number
- Interface
- Type
- Model

For verbose only:

- Power status
- Capacity in Gbytes

For each logical drive

- Drive status
- LogicalDrive[<idx>] with <idx> as combination of driver control number and driver number
- Name
- OS device name
- Level
- Total size in Gbytes

HINT: Some parts are omitted if no corresponding information is available.

- Error print sample

```
prefailure: RAID - Controller(prefailure) PhysicalDevice(prefailure)
LogicalDrive(ok)
prefailure: RAIDCtrl[1] Name="LSI 1064SASIME-2550 (0)" Description="PCI Bus 1,
Device 0, Function 0"
failurePredicted:online: PhysicalDevice[] Name="SEAGATE ST9146802SS (0)"
ID=3NM1J4DE CntErrors=1 SlotNr=0 Interface=sas Type=disk
```

- User interface sample in Icinga Classic 1.5:

Service State Information

Current Status:	WARNING (for 8d 4h 57m 58s)
Status Information:	WARNING - ID=YKHNT500101 - MassStorage(degraded) Systemboard(ok) Systemname=Tx120S22 Description="Hardware: Intel64 Family 6 Model 23 Stepping 6 AT/AT COMPATIBLE - Software: Windows Version 6.1 (Build 7601 Multiprocessor Free)" Model="PRIMERGY TX120 S2" AdminURL=http://10.10.10.1:80 prefailure: RAID - Controller(prefailure) PhysicalDevice(prefailure) LogicalDrive(ok) prefailure: RAIDCtrl[1] Name="FTS RAID 5/6 SAS based on LSI MegaRAID (0)" Description="PCI Bus 16, Device 0, Function 0" failurePredicted:ready: PhysicalDevice[] Name="FUJITSU MBB2073RC (2)" ID=BSA4P7B001N9 CntErrors=1 SlotNr=2 Interface=sas Type=disk
Performance Data:	
Current Attempt:	3/3 (HARD state)
Last Check Time:	10-17-2012 14:01:24
Check Type:	ACTIVE
Check Latency / Duration:	0.139 / 0.334 seconds
Next Scheduled Check:	10-17-2012 14:11:24
Last State Change:	10-09-2012 09:12:06
Last Notification:	10-17-2012 13:16:09 (notification 189)
Is This Service Flapping?	N/A
In Scheduled Downtime?	NO
Last Update:	10-17-2012 14:09:58 (0d 0h 0m 6s ago)

4.4.14 Driver Monitor Information

The driver monitor information can be monitored using the option *chkdrvmonitor*.

- Summary line:

...DrvMonitor(<status>)...

- Long output in the case of WARNING or CRITICAL or verbose=2 output

Driver monitor component table:

- Status
- DrvMon[<idx>] with <idx> as a combination of unit and number of entries
- Name
- Class
- Type
- Driver name
- Location

- Verbose sample

* Driver Monitor Component Table:

```
ok: DrvMon[1.2] Name="Intel(R) 82575EB Gigabit Ethernet Controller" Class=network
Type=pci Driver=elqexpress Location="onboard"
ok: DrvMon[1.3] Name="Intel(R) 82575EB Gigabit Ethernet Controller" Class=network
Type=pci Driver=elqexpress Location="onboard"
ok: DrvMon[1.1] Name="LSI RAID 0 1 SAS 8P" Class=storage Type=pci Driver=LSI_SAS
Location="Slot 3"
ok: DrvMon[1.4] Name="ICH10 SATA Controller 1" Class=storage Type=pci
Driver=pciide Location="onboard"
ok: DrvMon[1.5] Name="ICH10 SATA Controller 2" Class=storage Type=pci
Driver=pciide Location="onboard"
```

4.5 Check for RAID Standalone

This checks the status in **Status.mib (SV SNMP Agent)**, if it exists, and checks existence and status data in **RAID.mib**.

4.5.1 Options

Optional additional options:

--chkstorage

Select only "MassStorage" data.

HINT: If this option is combined with the above-mentioned options for PRIMERGY, the existence of ServerView SNMP Agent MIB support is required! If the Status.mib information cannot be read an error message will be printed.

4.5.2 RAID Information

- Summary line:

...MassStorage(<RAID status or overall status>)

The rest of the data is identical with the descriptions in "System – MassStorage (RAID) Information".

- Additional line if only RAID data is available

Hint="RAID Only Check - status is RAID status"

4.6 Check of PRIMERGY Management Blades

The check depends on the existence and use of S31.mib.

4.6.1 Options

For these servers enter the option:

--blade

This can be combined with:

--chkssystem

Anything other than

"Environment" (fan, temperature) or "Power"

--chkenv

Fan and temperature sensors

The environment information can be split using the following options instead of chkenv.

--chkenv-fan

Fan

--chkenv-temp

Temperature sensor, including threshold data

--chkpower

Check: Power supply units and power consumption, including threshold data if available.

If none of these check options are set, all checks will be performed.

These options can be combined.

4.6.2 System Information

- Summary line:
 - Nagios status
 - Serial number (ID) on the management blade
 - If chkssystem is not used, than all subcomponents and their summary status are printed
- Verbose sample:
OK - ID=sqywwn00000-...

4.6.3 System Information in the Case of WARNING or CRITICAL - for Notifications

- Serial identification - Additional information in the case of errors or for Verbose Level 1:
- RFC1213 Information - Additional information in the case of errors or for Verbose Level 1:
 - Server system name
 - Description string
 - Location (if defined)
 - Contact (if defined)
- S31 agent information - Additional information in the case of errors or for Verbose Level 1:
 - Administrative URL (if available)
- Notify sample:
ID=SQYWWN00000 Systemname=CF-BX600 Description="BX600" Location="Xxxx"
Contact="Xxxx Xxxx" AdminURL=http://xxx.xxx.xxx.xxx:80

4.6.4 Fan Information

This is part of the "Environment" information (chkenv).

- Summary line for chkenv:
Environment(<fan_and_temperature>)
- Summary line for chkenv-fan:
Fans(<all_fans_status>)

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each fan:

- Fan status
- Fan[<idx>] with <idx> as number of fan
- Name
- Speed in rpm (rotation per minute)

- Verbose sample:

```
...
* Fans:
ok: Fan[1] Name=RearFanUnit1-Fan-1 Speed=7968rpm
ok: Fan[2] Name=RearFanUnit1-Fan-2 Speed=6592rpm
ok: Fan[3] Name=RearFanUnit1-Fan-3 Speed=8032rpm
ok: Fan[4] Name=RearFanUnit1-Fan-4 Speed=6528rpm
ok: Fan[5] Name=RearFanUnit1-Fan-5 Speed=7968rpm
ok: Fan[6] Name=RearFanUnit1-Fan-6 Speed=6528rpm
ok: Fan[7] Name=RearFanUnit1-Fan-7 Speed=7968rpm
ok: Fan[8] Name=RearFanUnit1-Fan-8 Speed=6528rpm
ok: Fan[9] Name=RearFanUnit2-Fan-1 Speed=13312rpm
ok: Fan[10] Name=RearFanUnit2-Fan-2 Speed=10496rpm
...
```

4.6.5 Temperature Information

This is part of the "Environment" information (chkenv).

- Summary line for chkenv:

Environment(<fan_and_temperature>)

- Summary line:

Temperatures(<all_temperature_sensor_status>)

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each temperature sensor

- Sensor status
- Sensor[<idx>] with <idx> as number of sensor
- Name
- Temperature values in Celsius: CURRENT, WARNING level and CRITICAL level

- Performance data:

<identification of sensor>=<current>C;<warn>;<critical>

- Verbose sample:

```
...
* Temperature Sensors:
ok: Sensor[1] Name=Housing-Left Temperature=33C Warning=70C Critical=85C
ok: Sensor[2] Name=Housing-Center Temperature=35C Warning=70C Critical=85C
ok: Sensor[3] Name=Housing-Right Temperature=30C Warning=70C Critical=85C
ok: Sensor[4] Name=Ambient Temperature=24C Warning=40C Critical=45C
ok: Sensor[5] Name=Front-Side_I/O-Module Temperature=24C Warning=55C Critical=60C
ok: Sensor[6] Name=MMB-1 Temperature=45C Warning=76C Critical=80C
ok: Sensor[7] Name=MMB-2 Temperature=44C Warning=76C Critical=80C
ok: Sensor[8] Name=Connection_Blade-1 Temperature=44C Warning=84C Critical=88C
ok: Sensor[9] Name=Connection_Blade-2 Temperature=42C Warning=60C Critical=65C
ok: Sensor[10] Name=Connection_Blade-3 Temperature=37C Warning=65C Critical=68C
ok: Sensor[11] Name=Connection_Blade-4 Temperature=38C Warning=65C Critical=68C
ok: Sensor[12] Name=PSU-1 Temperature=29C Warning=90C Critical=95C
ok: Sensor[13] Name=PSU-2 Temperature=29C Warning=90C Critical=95C
ok: Sensor[14] Name=Rear_Fan_Unit-3 Temperature=24C Warning=90C Critical=95C
ok: Sensor[15] Name=Rear_Fan_Unit-4 Temperature=24C Warning=90C Critical=95C
| Housing-Left=33C;70;85 Housing-Center=35C;70;85 Housing-Right=30C;70;85
Ambient=24C;40;45 Front-Side_I/O-Module=24C;55;60 MMB-1=45C;76;80 MMB-2=44C;76;80
Connection_Blade-1=44C;84;88 Connection_Blade-2=42C;60;65 Connection_Blade-
3=37C;65;68 Connection_Blade-4=38C;65;68 PSU-1=29C;90;95 PSU-2=29C;90;95
Rear_Fan_Unit-3=24C;90;95 Rear_Fan_Unit-4=24C;90;95
```

4.6.6 Power Supply Information

This is part of the "Power" information (chkpower).

- Summary line:

PowerSupplies(<all_PowerSupplyUnit_Status>)

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each PSU:

- PSU status
- PowerSupplyUnit[<idx>] with <idx> as number of PSU
- Serial number (ID)
- Product
- Model (always verbose print – printed for notify only if this is no internal boarding string like A3C*)

- Verbose sample:

```
* Power Supplies:
ok: PowerSupplyUnit[1] ID=G743DN002P0EF Product="AA25370L" Model="A3C40094088"
ok: PowerSupplyUnit[2] ID=G743DN005J0EF Product="AA25370L" Model="A3C40094088"
ok: PowerSupplyUnit[3] ID=G743DN005X0EF Product="AA25370L" Model="A3C40094088"
ok: PowerSupplyUnit[4] ID=G743DN00510EF Product="AA25370L" Model="A3C40094088"
fanmodule: PowerSupplyUnit[5] ID=G818DG00110CF Product="AA25370L-RFN"
Model="A3C40094164"
fanmodule: PowerSupplyUnit[6] ID=G818DG00280CF Product="AA25370L-RFN"
Model="A3C40094164"
```

4.6.7 Power Consumption Information

This is part of the "Power" information (chkpower).

- Long output for verbose=2:
 - Time stamp of information (seconds)
 - Average, minimum and maximum in watts
- Performance data:

PowerConsumption=<average>Watt

- Verbose sample:

```
PowerConsumption - TimeStamp=1348833000 Average=2079Watt Min=325Watt Max=2411Watt
| PowerConsumption=2079Watt
```

4.7 Blades Controlled by a Primergy Management Blade

This depends on the existence and use of S31.mib

4.7.1 Options

As an alternative to --blade, which checks the overall state and performance of the management blade with the following options, additional checks for blades assigned to this management blade can be retrieved.

The subblades are grouped into four types:

- Server blades
- I/O connection blades, including:
 - Switch, Fibre Channel switch, LAN pass-through, FSIOM, Serial-Attached SCSI switch (SAS)
- Keyboard/video/mouse (KVM) blades
- Storage blades

Options:

--bladeinside

This is a combination of the four following options with the advantage that error messages concerning unknown blade types are suppressed.
(Do not combine with --blade).

--bladesrv

Check server blade status values on a PRIMERGY blade server.

--bladeio

Check all I/O connection blade status values on a PRIMERGY blade server.

The *bladeio* information can be split with one or more of the following options instead of *--bladeio*.

--bladeio-switch Switch
--bladeio-fcswitch Fibre Channel switch
--bladeio-phy LAN pass-through
--bladeio-fsiom FSIOM
--bladeio-sasswitch SAS switch

--bladekvm

Check keyboard/video/mouse blade status values on a PRIMERGY blade server.

--bladestore

Check storage blade status values on a PRIMERGY blade server.

4.7.2 Information in the Case of WARNING or CRITICAL - for Notifications

- RFC1213 information - Additional information in the case of errors or for Verbose Level 1
 - Server system name
 - Description string
 - Location (if defined)
 - Contact (if defined)
- S31 agent information - Additional information in the case of errors or for Verbose Level 1:
 - Serial identification - Additional information in the case of errors or for Verbose Level 1:
 - Administrative URL (if available)
- Notify sample (Verbose Level 1):
 MMB System Information - Systemname=BX900-6 Description="PRIMERGY BX900 S1
 A3C40119934" Location="Xxx" Contact="Xxx" ID=SQ1010P00117
 AdminURL=http://xxx.xxx.xxx.xxx:80

4.7.3 Server Blades

- Summary line:
 Server-<one status>(<count_of_server_with_this_status>)[-<status>(<count>[...])]
- ATTENTION:** The <status> is not just limited to the four-value system of Nagios - there are additional specific status values available.
- Long output in the case of WARNING or CRITICAL or Verbose Level 2:
 For each server blade:
 - Server blade status
 - Server[<idx>] with <idx> as server blade number identical with slot number
 - Serial number (ID)
 - Host name (if available)
 - Administrative URL
 - Product and model (always verbose print – printed for notify only if these are no internal boarding strings)
- Notify sample:
 CRITICAL Server-critical(1)-standby(11)
 critical: Server[4] ID=QTFMQK84200076 ID2=System_429
 AdminURL=http://xxx.xxx.xxx.xxx Product=D2860 Model=A3C40094854

- User interface sample in Icinga Classic 1.5:

Service State Information

Current Status:	CRITICAL (for 1d 0h 6m 0s)
Status Information:	CRITICAL Server-critical(2)-standby(4) critical: Server[1] ID=QTFMQK83900186 ID2=System_422 AdminURL=http:// critical: Server[11] ID=SQ1008MR00076 ID2=System_117 AdminURL=http://
Performance Data:	
Current Attempt:	3/3 (HARD state)
Last Check Time:	10-17-2012 14:01:50
Check Type:	ACTIVE
Check Latency / Duration:	0.159 / 2.801 seconds
Next Scheduled Check:	10-17-2012 14:11:50
Last State Change:	10-16-2012 13:57:50
Last Notification:	10-17-2012 13:11:59 (notification 24)
Is This Service Flapping?	N/A
In Scheduled Downtime?	NO
Last Update:	10-17-2012 14:03:48 (0d 0h 0m 2s ago)

4.7.4 Additional Information about Server Blades

With additional Verbose Level 2 you can get the server blade table entries mentioned above and additionally the NIC table information.

With this information you can get the host name and IP addresses (available in NIC table if the system is on) and management IP of each server blade.

Sample:

```
* Server Blades:
ok: Server[1] ID=SQ948MS00112 ID2=System_053 Hostname=localhost
AdminURL=http://nnn.nnn.nnn.nnn Product="D3030" Model="A3C40114042"
ok: Server[2] ID=SQ1124MV00162 ID2=920S3063 Hostname=SW1-BX920S3-2.v
AdminURL=http://nnn.nnn.nnn.nnn Product="D3142" Model="A3C40125624"
ok: Server[3] ID=SQ948MS00002 ID2=System_125 Hostname=pdbsm-bx924s3-2
AdminURL=http://nnn.nnn.nnn.nnn Product="D3030" Model="A3C40114042"
ok: Server[4] ID=SQ1143MW00050 ID2=924S3116 Hostname=SW1-BX924S3
AdminURL=http://nnn.nnn.nnn.nnn Product="D3143" Model="A3C40125625"
unknown: Server[5] ID=QTFMQK83900390 ID2=System_178 Hostname=SW1-BX920S1-1
Product="D2860" Model="A3C40094854"
ok: Server[6] ID=SQ1146MV00171 ID2=920S3106 Hostname=SW1-BX920S3-1
AdminURL=http://nnn.nnn.nnn.nnn Product="D3142" Model="A3C40125624"
* Server Blade NIC Table:
ServerNicInfo[1.1] - MAC=00:26:9E:82:96:9A Type=on-board-lan-controller
ServerNicInfo[3.1] - MAC=00:26:9E:82:96:36 Type=on-board-lan-controller
ServerNicInfo[4.1] - MAC=04:7D:7B:06:D5:F8 IP=xxx.xxx.xxx.xxx Type=on-
board-lan-c ontroller
ServerNicInfo[4.2] - MAC=04:7D:7B:06:D5:FC IP=xxx.xxx.xxx.xxx Type=on-
board-lan-c ontroller
ServerNicInfo[5.1] - MAC=00:1E:68:DA:55:BC IP=xxx.xxx.xxx.xxx Type=on-
board-lan-c ontroller
ServerNicInfo[6.1] - MAC=04:7D:7B:11:26:F8 Type=on-board-lan-controller
```


4.7.4.1 How to Detect the Host Name and IP Address of Server Blades

Use the above-mentioned options --bladesrv and --verbose=2, the host name from the first table print and, (if available,) detect with the server index the related IP addresses in the NIC table.

In the above mentioned example it is

```
ok: Server[4] ID=SQ1143MW00050 ID2=924S3116 Hostname=SW1-BX924S3
AdminURL=http://nnn.nnn.nnn.nnn Product="D3143" Model="A3C40125625"
```

and

```
ServerNicInfo[4.1] - MAC=04:7D:7B:06:D5:F8 IP=xxx.xxx.xxx.xxx
Type=on-board-lan-controller
ServerNicInfo[4.2] - MAC=04:7D:7B:06:D5:FC IP=xxx.xxx.xxx.xxx
Type=on-board-lan-controller
```

4.7.5 I/O Connection – FSIOM

- Summary line:
FSIOM(<fsiom_status>)
- Long output in the case of WARNING or CRITICAL or verbose=2 output:
 - FSIOM status
 - Serial number (ID)
 - Product and model (always verbose print – printed for notify only if these are no internal boarding strings)
- Sample
ok: FSIOM ID=QTFFZT103500046 Product="D3080" Model="A3C40120510"

4.7.6 I/O Connection – Switch Blades

- Summary line:
Switch-<one status>(<count_of_switch_with_this_status>)[-<status>(<count>[...])]
- ATTENTION:** The <status> is not just limited to the four-value system of Nagios - there are additional specific status values available.
- Long output in the case of WARNING or CRITICAL or verbose=2 output:
For each switch blade:
 - Switch status
 - Switch[<idx>] with <idx> as number of switch
 - Serial number (ID)
 - Second identification string
 - IP address (if available)
 - Name – customer-assigned name (if available)
 - Product and model (always verbose print of Model– Model printed for notify only if this is no internal boarding string)
- Verbose sample:
OK Switch-ok(3)
Switch Blades:
ok: Switch[2] ID=SQ719LS00263 Product=BX600 GbE Switch Blade 30/12
Model=A3C40089238
ok: Switch[3] ID=SQ838LS02221 Product=BX600 GbE Switch Blade 30/12
Model=A3C40089238
ok: Switch[4] ID=SQ719LS00250 IP=xxx.xxx.xxx.xxx Product=BX600 GbE Switch Blade 30/12
Model=A3C40089238

4.7.7 I/O Connection – Fibre Channel Switch

- Summary line:

Fibre Channel Switch-(<one status>(<count_of_switch_with_this_status>)[-<status>(<count>[...])])]

ATTENTION: The <status> is not just limited to the four-value system of Nagios - there are additional specific status values available.

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each FC switch:

- FC switch status
- FCSwitch[<idx>] with <idx> as number of FC switch
- Serial number (ID)
- Second identification string
- Slot number
- IP address (if available)
- Product and model (always verbose print of Model– Model printed for notify only if this is no internal boarding string)

- Verbose sample:

* Fiber Channel Switch Blades:

ok: FCSwitch[1] ID=AHQ0437D01K Slot=3 IP=xxx.xxx.xxx.xxx Product="PRIMERGY BX600 4Gb/4Gb FC SW" Model="A3C40085440 "

ok: FCSwitch[2] ID=AHQ0446B00G Slot=4 IP=xxx.xxx.xxx.xxx Product="PRIMERGY BX600 4Gb/4Gb FC SW" Model="A3C40085440 "

4.7.8 I/O Connection – Serial Attached SCSI Switch (SASSwitch)

- Summary line:

Serial Attached SCSI Switch-(<one status>(<count_of_switch_with_this_status>)[-<status>(<count>[...])])]

ATTENTION: The <status> is not just limited to the four-value system of Nagios - there are additional specific status values available.

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each SAS switch

- SAS status
- SASSwitch[<idx>] with <idx> as number of SAS switch
- Serial number (ID)
- Second identification string
- Slot number
- IP address (if available)
- Product and model (always verbose print of Model– Model printed for notify only if this is no internal boarding string)

- Verbose sample:

* Serial Attached SCSI Switch Blades:

ok: SASSwitch[3] ID=35279450 Slot=3 IP=xxx.xxx.xxx.xxx Product="PY CB SAS Switch 6Gb 18/6" Model="A3C40121295"

ok: SASSwitch[4] ID=35279444 Slot=4 IP=xxx.xxx.xxx.xxx Product="PY CB SAS Switch 6Gb 18/6" Model="A3C40121295"

4.7.9 I/O Connection – LAN Pass-Through (Phy)

- Summary line:

LAN Pass Through Blades-`<one status>(<count_of_switch_with_this_status>)[-<status>(<count>[...])]`]

ATTENTION: The `<status>` is not just limited to the four-value system of Nagios - there are additional specific status values available.

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each LPT blade

- o LPT status
- o LANPT[`<idx>`] with `<idx>` as number of blade
- o Serial number (ID)
- o Second identification string
- o Product

- Verbose sample:

* LAN Pass Through Blades:

ok: LANPT[8] ID=FJ1038X00014 Product="PY CB Eth Pass Thru 10Gb 18/18"

4.7.10 Keyboard/Video/Mouse Blades (KVM)

- Summary line:

KVM-`<one status>(<count_of_switch_with_this_status>)[-<status>(<count>[...])]`]

ATTENTION: The `<status>` is not just limited to the four-value system of Nagios - there are additional specific status values available.

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each KVM:

- o KVM status
- o KVM[`<idx>`] with `<idx>` as number of blade
- o Serial number (ID)
- o IP address (if available)
- o Product and model (always verbose print of Model– Model printed for notify only if this is no internal boarding string)

- Verbose sample:

OK KVM-ok((1)

Key-Video-Mouse Blades:

ok: KVM[1] IP=xxx.xxx.xxx.xxx Product="BX600 Digital KVM Switch"

Model="A3C40083767"

4.7.11 Storage Blades

- Summary line:

Storage Blades-`<one status>(<count_of_switch_with_this_status>)[-<status>(<count>[...])]`]

ATTENTION: The `<status>` is not just limited to the four-value system of Nagios - there are additional specific status values available.

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each storage blade

- o Storage blade status
- o Storage[`<idx>`] with `<idx>` as number of blade
- o Serial number (ID)
- o Second identification string (if available)
- o Administrative URL (if available)
- o Product and model (always verbose print –printed for notify only if these are no internal boarding strings)

- Verbose sample:
 OK Storage-Blades-ok(3)
 Storage Blades:
 ok: Storage[2] ID=SQ934MK00029 ID2=System_330 Product=D2714 Model=A3C40100456
 ok: Storage[4] ID=35305583 ID2=System_025 Product=D3103 Model=A3C40121258
 ok: Storage[6] ID=35305576 ID2=System_027 Product=D3103 Model=A3C40121258

4.8 PRIMEQUEST

Based on **MMB-COM-MIB.mib** and **PSA-COM-MIB.mib**.

4.8.1 Options

--pq or --primequest

This can be combined with:

--chkssystem

Check anything other than

"Environment" (fan, temperature), "Power" or "Hardware" parts.

--chkenv

Check fan and temperature sensors

The environment information can be split using the following options instead of chkenv.

--chkenv-fan Fan

--chkenv-temp Temperature sensor, includes performance data

--chkpower

Check power supply units and power consumption, including threshold data if available.

If none of these are set, the system, environment and power ranges will be checked.

--chkhardware

"Hardware" means voltage, CPU or memory modules.

Tests have shown that in case of non-OK status the collection of details need sometimes much time.

The system hardware information can be split with the following options.

--chkcpu CPU information

--chkvoltage Voltage information

--chkmemmodule Memory modules information

These options can be combined.

ATTENTION: If one of the environment, power or hardware units has the status WARNING or CRITICAL the assigned "partitions" and "system boards" might also have this status.

Because of this, chkssystem will return a non-OK status if one of the partition summary status are not OK!

4.8.2 System Information - Overall Status and Chassis Information

- Summary line:
 - Nagios status
 - Serial number (ID)
 - Summary status in SNMP-PRIMEQUEST nomenclature
 - Chassis subcomponent status values
 - Partition status count
 - System board status count

If chkssystem is used, the environment, power and hardware subcomponent status values are suppressed.

HINT: Partitions and system board will show a non-OK status if they are connected to non-OK CPUs, power supplies or other components.

- Sample

```
CRITICAL ID=1480932004 - All=failed - Chassis: temperatures(ok) fans(ok) power-  
supplies(error) voltages(ok) cpus(failed) memory-modules(ok) - Partitions:-ok(1)-  
failed(2) - SystemBoard:-ok(1)-failed(3)
```

4.8.3 System Information in the Case of WARNING or CRITICAL - for Notifications

- Serial identification - Additional information in the case of errors or for Verbose Level 1:
- RFC1213 information - Additional information in the case of errors or for Verbose Level 1:
 - Server system name
 - Description string
 - Location (if defined)
 - Contact (if defined)
- MMB-COM first unit information - Additional information in the case of errors or for Verbose Level 1:
 - Chassis unit name
 - Administrative URL (if available)
 - Chassis model
- Notify sample:
 ID=1480932004 Systemname=FTS4 Description="PRIMEQUEST Management Board"
 Location="Xxxx" Contact="Xxx Xxx" AdminURL=http://xxx.xxx.xxx.xxx:8081
 Model="PRIMEQUEST 1800L" Designation=MCD3AC111

4.8.4 Additional System Board and Partition Information

With Verbose Level 3 additional information will be printed:

- Partition component status (verbose=2)
 - Subcomponents and their status
- System board overall status (verbose=2)
- Management node table (verbose=3)
 - For more see section 4.8.8 "Additional System Information – Management Node Table" on page 57.
- Server NodeTable (verbose=3)
 - For more see section 4.8.9 "Additional System Information – Server Table" on page 57.

The following are dependent on the OS installation on the partitions:

- Partition management information (verbose=2)
 For each partition:
 - Partition number
 - Partition name
 - Assigned host name
 - Assigned IP address for host name
- Partition Related Physical Components Tables – System board and CPU assignments (verbose=2)
 - ... For more see sample below:

Sample:

```
* StatusComponentTable - Partitions:  
Partition[0] - temperatures(ok) voltages(ok) cpus(failed) memory-  
modules(ok) total(failed)  
Partition[1] - temperatures(ok) voltages(ok) cpus(failed) memory-  
modules(ok) total(failed)  
Partition[2] - temperatures(ok) voltages(ok) cpus(ok) memory-modules(ok)  
total(ok)  
* StatusComponentTable - SystemBoards:  
SystemBoard[0] - total(failed)  
SystemBoard[1] - total(failed)  
SystemBoard[2] - total(failed)  
SystemBoard[3] - total(ok)  
* Management Node:  
Node[136.1] - Name=SIRIUS1480937002 Address=xxx.xxx.xxx.xxx  
MAC=00:17:42:9B:D8:96 Class=management-blade
```

```

Node[136.2] - MAC=00:17:42:9B:D8:96 Class=management-blade
Node[137.1] - MAC=00:17:42:9B:D8:94 Class=secondary-management-blade
* Server Table:
Server[2] - Partition=0 BootStatus=os-running ManagementIP=xxx.xxx.xxx.xxx
UUID=9AB0C000-F668-11DE-8000-001742DE7C58
Server[3] - Partition=1 BootStatus=os-running ManagementIP=xxx.xxx.xxx.xxx
UUID=9AB0C000-F668-11DE-8000-001742DE7C59
Server[4] - Partition=2 BootStatus=reset ManagementIP=xxx.xxx.xxx.xxx
UUID=9AB0C000-F668-11DE-8000-001742DE7C5A
* Partition Management Info:
Partition[0] - Name=Partition#0 Hostname=FTS5P0 IP=xxx.xxx.xxx.xxx
Partition[1] - Name=Partition#1 Hostname=FTS5P1 IP=xxx.xxx.xxx.xxx
* Partition Related Physical Components Tables:
Partition[0] -
  System Board Table:
    SystemBoard[0] - ID=PP09370655 Name=SB#0 Model="CA07125-D001 B2"
  CPU Table:
    CPU[0.0] - Name=SB#0-CPU#0
    CPU[0.1] - Name=SB#0-CPU#1
Partition[1] -
  System Board Table:
    SystemBoard[1] - ID=PP09370656 Name=SB#1 Model="CA07125-D001 B2"
    SystemBoard[2] - ID=PP09370653 Name=SB#2 Model="CA07125-D001 B2"
  CPU Table:
    CPU[1.0] - Name=SB#1-CPU#0
    CPU[1.1] - Name=SB#1-CPU#1
    CPU[2.0] - Name=SB#2-CPU#0
    CPU[2.1] - Name=SB#2-CPU#1
Partition[2] -
Partition[3] -

```

4.8.4.1 How to Detect Partition-related Host Names and Address

Use the above-mentioned --pq --chkssystem --verbose=2 or 3. The "Partition Management Info" contains the host name and the IP address of the assigned server. "Server Table" (verbose=3) contains the management IP address of the assigned server.

This information is not always available.

4.8.5 Additional System Information –Table of all Units

HINT: The PRIMEQUEST 2800 system has additional unit classes.

- Long output for Verbose Level 231 - **ATTENTION:** This requires several seconds.

Unit table

Fields inside:

- Serial number
- Class
- Name
- Location
- Contact
- Administrative URL
- Model

Sample:

```

UnitTable:
[2] - Class=partition Name=Partition#0
[3] - Class=partition Name=Partition#1
[4] - Class=partition Name=Partition#2
[5] - Class=partition Name=Partition#3
[18] - Class=free-pool
[19] - ID=PP09370655 Class=sb Name=SB#0 Model="SB"
[20] - ID=PP09370656 Class=sb Name=SB#1 Model="SB"
[21] - ID=PP09370653 Class=sb Name=SB#2 Model="SB"
[22] - ID=PP094304PB Class=sb Name=SB#3 Model="SB"
[43] - ID=PP0934046P Class=iou Name=IOB#0 Model="IOB"

```

```

[44] - ID=PP0934046N Class=iou Name=IOB#1 Model="IOB"
[51] - Class=iou-divided Name=IOB#0A
[52] - Class=iou-divided Name=IOB#0B
[53] - Class=iou-divided Name=IOB#1A
[54] - Class=iou-divided Name=IOB#1B
[67] - Class=pci-box Name=PCI_Box#0
[68] - Class=pci-box Name=PCI_Box#1
[118] - Class=fan-tray Name=FAN#0
[119] - Class=fan-tray Name=FAN#1
[120] - Class=fan-tray Name=FAN#2
[121] - Class=fan-tray Name=FAN#3
[122] - Class=fan-tray Name=FAN#4
[123] - Class=fan-tray Name=FAN#5
[124] - Class=fan-tray Name=FAN#6
[125] - Class=fan-tray Name=FAN#7
[136] - ID=PP0935031G Class=mmb Name=MMB#0 Model="MMB"
[137] - ID=PP0935031K Class=mmb Name=MMB#1 Model="MMB"
[143] - ID=PP093404UA Class=gspb Name=GSPB#0 Model="GSPB"
[144] - ID=PP093404UB Class=gspb Name=GSPB#1 Model="GSPB"
[145] - Class=gspb-divided Name=GSPB#0A
[146] - Class=gspb-divided Name=GSPB#0B
[147] - Class=gspb-divided Name=GSPB#1A
[148] - Class=gspb-divided Name=GSPB#1B
[149] - ID=PP0935040M Class=sas-unit Name=SASU#0 Model="SASU"
[150] - ID=PP0935040N Class=sas-unit Name=SASU#1 Model="SASU"
[151] - ID=PP0935040Q Class=sas-unit Name=SASU#2 Model="SASU"
[152] - ID=PP0935040R Class=sas-unit Name=SASU#3 Model="SASU"
[153] - ID=FA09300134 Class=psu Name=PSU#0 Model="FH2000U1"
[154] - Class=psu Name=PSU#1
[155] - ID=FA09300152 Class=psu Name=PSU#2 Model="FH2000U1"
[156] - ID=FA09300149 Class=psu Name=PSU#3 Model="FH2000U1"
[157] - ID=PP0934046R Class=dvdb Name=DVDB Model="DVDB"
[158] - Class=lpci-box Name=PCI_Box#0
[159] - Class=lpci-box Name=PCI_Box#1
[160] - Class=lpci-box Name=PCI_Box#2
[161] - Class=lpci-box Name=PCI_Box#3

```

4.8.6 Additional System Information – Status Component Table

HINT: The PRIMEQUEST 2800 system has additional unit classes.

- Long output for Verbose Level 821 - **ATTENTION:** This requires several seconds.

Status component table

Fields inside:

- Unit ID
- Status for each status type.

Sample:

```

CRITICAL ID=1480937002 - All=error - Partitions:-ok(1)-failed(2) -
SystemBoard:-ok(1)-failed(3)
StatusComponentTable - Partitions:
Partition[0] - temperatures(ok) voltages(ok) cpus(failed) memory-modules(ok)
total(failed)
Partition[1] - temperatures(ok) voltages(ok) cpus(failed) memory-modules(ok)
total(failed)
Partition[2] - temperatures(ok) voltages(ok) cpus(ok) memory-modules(ok)
total(ok)
StatusComponentTable - SystemBoards:
SystemBoard[0] - total(failed)
SystemBoard[1] - total(failed)
SystemBoard[2] - total(failed)
SystemBoard[3] - total(ok)
Other Units:
[43]total(ok) [44]total(ok) [51]total(ok) [52]total(ok) [53]total(ok)
[54]total(ok) [118]total(ok) [119]total(ok) [120]total(ok) [121]total(ok)
[122]total(ok) [123]total(ok) [124]total(ok) [125]total(ok) [136]total(ok)

```

```
[137]total(degraded) [143]total(ok) [144]total(ok) [149]total(ok)
[150]total(ok) [151]total(ok) [152]total(ok) [153]total(ok) [155]total(ok)
[156]total(error) [157]total(ok)
```

4.8.7 Additional System Information – Unit-Child Relations

HINT: The PRIMEQUEST 2800 system has additional unit classes.

- Long output for Verbose Level 261 - **ATTENTION:** This requires several seconds.
The counterpart is available with 251 'Unit-Parent table'. The Unit-Child table is easier to understand.

Unit-Child table

Fields inside:

- ParentUnitId ---> ChildId and Child-Class.

Sample:

UnitChildTable:

```
1 ---> 2 Class=partition
1 ---> 3 Class=partition
1 ---> 4 Class=partition
1 ---> 18 Class=free-pool
1 ---> 19 Class=sb
1 ---> 20 Class=sb
1 ---> 21 Class=sb
1 ---> 22 Class=sb
1 ---> 43 Class=iou
1 ---> 44 Class=iou
1 ---> 118 Class=fan-tray
1 ---> 119 Class=fan-tray
1 ---> 120 Class=fan-tray
1 ---> 121 Class=fan-tray
1 ---> 122 Class=fan-tray
1 ---> 123 Class=fan-tray
1 ---> 124 Class=fan-tray
1 ---> 125 Class=fan-tray
1 ---> 136 Class=mmb
1 ---> 137 Class=mmb
1 ---> 143 Class=gspb
1 ---> 144 Class=gspb
1 ---> 149 Class=sas-unit
1 ---> 150 Class=sas-unit
1 ---> 151 Class=sas-unit
1 ---> 152 Class=sas-unit
1 ---> 153 Class=psu
1 ---> 155 Class=psu
1 ---> 156 Class=psu
1 ---> 157 Class=dvdb
2 ---> 19 Class=sb
2 ---> 51 Class=iou-divided
2 ---> 145 Class=gspb-divided
3 ---> 20 Class=sb
3 ---> 21 Class=sb
3 ---> 52 Class=iou-divided
3 ---> 146 Class=gspb-divided
4 ---> 22 Class=sb
4 ---> 53 Class=iou-divided
4 ---> 147 Class=gspb-divided
18 ---> 54 Class=iou-divided
18 ---> 148 Class=gspb-divided
43 ---> 51 Class=iou-divided
43 ---> 52 Class=iou-divided
44 ---> 53 Class=iou-divided
44 ---> 54 Class=iou-divided
143 ---> 145 Class=gspb-divided
143 ---> 146 Class=gspb-divided
```



```
144 ---> 147 Class=gspb-divided
144 ---> 148 Class=gspb-divided
```

4.8.8 Additional System Information – Management Node Table

- Long output for Verbose Level 311 or Level 3

Management node table

Fields inside:

- Name
- Address
- MAC
- Class

Sample:

```
Management Node:
Node[136.1] - Name=SIRIUS1480937002 Address=xxx.xxx.xxx.xxx
MAC=00:17:42:9B:D8:96 Class=management-blade
Node[136.2] - MAC=00:17:42:9B:D8:96 Class=management-blade
Node[137.1] - MAC=00:17:42:9B:D8:94 Class=secondary-management-blade
```

4.8.9 Additional System Information – Server Table

- Long output for Verbose Level 411 or Level 3

Server table

Fields inside:

- Partition relation
- Boot status
- Management IP address
- UUID

Sample:

```
Server Table:
Server[2] - Partition=0 BootStatus=os-running ManagementIP=xxx.xxx.xxx.xxx
UUID=9AB0C000-F668-11DE-8000-001742DE7C58
Server[3] - Partition=1 BootStatus=os-running ManagementIP=xxx.xxx.xxx.xxx
UUID=9AB0C000-F668-11DE-8000-001742DE7C59
Server[4] - Partition=2 BootStatus=reset ManagementIP=xxx.xxx.xxx.xxx
UUID=9AB0C000-F668-11DE-8000-001742DE7C5A
```

4.8.10 Temperature Information

This is part of the "Environment" information (chkenv).

- Summary line:

```
... temperatures(<all_sensor_status>)
```

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each temperature sensor:

- Sensor status
- Sensor [<idx>] with <idx> as a combination of unit number and sensor number
- Name
- Temperature values in Celsius: CURRENT, WARNING level and CRITICAL level

- Performance data:

```
<identification of sensor>=<current>C;<warn>;<critical>
```

- Verbose sample:

```
* Temperature Sensors:
none: Sensor[1.83] Name=Inlet Temperature=23C Warning=38C
```

```

ok: Sensor[19.16] Name=SB#0_CPU#0 Temperature=41C Warning=89C Critical=90C
ok: Sensor[19.17] Name=SB#0_CPU#1 Temperature=44C Warning=89C Critical=90C
ok: Sensor[20.16] Name=SB#1_CPU#0 Temperature=40C Warning=89C Critical=90C
ok: Sensor[20.17] Name=SB#1_CPU#1 Temperature=40C Warning=89C Critical=90C
ok: Sensor[21.16] Name=SB#2_CPU#0 Temperature=43C Warning=89C Critical=90C
ok: Sensor[21.17] Name=SB#2_CPU#1 Temperature=46C Warning=89C Critical=90C
ok: Sensor[22.16] Name=SB#3_CPU#0 Temperature=44C Warning=89C Critical=90C
ok: Sensor[22.17] Name=SB#3_CPU#1 Temperature=46C Warning=89C Critical=90C
| Inlet=23C;38 SB#0_CPU#0=41C;89;90 SB#0_CPU#1=44C;89;90 SB#1_CPU#0=40C;89;90
SB#1_CPU#1=40C;89;90 SB#2_CPU#0=43C;89;90 SB#2_CPU#1=45C;89;90
SB#3_CPU#0=44C;89;90 SB#3_CPU#1=46C;89;90

```

4.8.11 Fan Information

This is part of the "Environment" information (chkenv).

- Summary line

... fans(<state>)

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each fan:

- Fan status
- Fan[<idx>] with <idx> as number of fan
- Name
- Speed in rpm (rotations per minute)

- Verbose sample

```

OK temperatures(ok) fans(ok)
* Fans:
ok: Fan[119] Name=FAN#1 Speed=5065rpm
ok: Fan[120] Name=FAN#2 Speed=5239rpm
ok: Fan[121] Name=FAN#3 Speed=5150rpm
ok: Fan[122] Name=FAN#4 Speed=5378rpm
ok: Fan[123] Name=FAN#5 Speed=5194rpm
ok: Fan[124] Name=FAN#6 Speed=5425rpm
ok: Fan[125] Name=FAN#7 Speed=5239rpm

```

4.8.12 Power Supply Information

This is part of the "Power" information (chkpower).

- Summary line:

power-supplies(<all_PowerSupplyUnit_Status>))

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each power supply unit:

- PSU status
- PSU[<idx>] with <idx> as number of PSU
- Name

- Verbose sample:

```

CRITICAL power-supplies(error)
* Power Supplies:
none: PSU[0] Name=PSU#0
ok: PSU[2] Name=PSU#2
failed: PSU[3] Name=PSU#3

```

4.8.13 Power Consumption Information

This is part of the "Power" information (chkpower).

HINT: With PRIMEQUEST 2800 or later the unit can be watts or Btu/h.

- Long output in case of verbose=2 output:
 - Current value
 - Warning limit (SNMP OID defined but not supported yet by PRIMEQUEST)
 - Redundancy critical Level (SNMP OID defined but not supported yet by PRIMEQUEST)
 - Nominal value
 - Max. limit threshold (if set and PRIMEQUEST 2800 or later)
 - Control mode (PRIMEQUEST 2800 or later)

- Performance data

PowerConsumption=<average>Watt;<min>;<critical>;<max>

Note about critical value:

This is only set if the "Max Limit Threshold" or "Redundancy Critical Level" are set. The last one overwrites the Max Limit Threshold setting (if available).

- Verbose sample
| PowerConsumption=1320Watt;;;4000

4.8.14 Hardware – Voltage Information

This is part of the "Hardware" information (chkhardware). This requires several seconds.

- Summary line:
voltages(<all_voltage_unit_Status>)

- Performance data

ATTENTION: Currently there will be no performance data generated for the Nagios tool.
It is only visible with verbose data.

- Long output in the case of WARNING or CRITICAL or verbose=2 or verbose=631:

For each voltage unit:

- Voltage status
- Voltage[<idx>] with <idx> as a combination of unit number and voltage number
- Name
- Volt values in mV: Current, Minimum and Maximum

- Verbose sample

```
...
* Voltages:
ok: Voltage[19.96] Name=SB#0_+5.0VL Current=497mV Min=459mV Max=540mV
ok: Voltage[19.97] Name=SB#0_+3.3VL Current=328mV Min=300mV Max=357mV
ok: Voltage[19.98] Name=SB#0_+1.8VL Current=176mV Min=165mV Max=194mV
ok: Voltage[19.99] Name=SB#0_+1.2VL Current=117mV Min=110mV Max=129mV
ok: Voltage[19.100] Name=SB#0_+12V#0 Current=1193mV Min=1101mV Max=1295mV
ok: Voltage[19.101] Name=SB#0_+12V#1 Current=1183mV Min=1101mV Max=1295mV
ok: Voltage[19.102] Name=SB#0_+5.0V Current=501mV Min=459mV Max=540mV
ok: Voltage[19.103] Name=SB#0_+3.3V Current=326mV Min=300mV Max=357mV
ok: Voltage[19.104] Name=SB#0_+1.8V Current=178mV Min=165mV Max=194mV
ok: Voltage[19.105] Name=SB#0_+1.5V#0 Current=149mV Min=138mV Max=162mV
ok: Voltage[19.106] Name=SB#0_+1.5V#1 Current=149mV Min=138mV Max=162mV
ok: Voltage[19.107] Name=SB#0_+1.5V#2 Current=148mV Min=138mV Max=162mV
ok: Voltage[19.108] Name=SB#0_+1.1V#0 Current=111mV Min=94mV Max=130mV
ok: Voltage[19.109] Name=SB#0_+1.1V#1 Current=110mV Min=101mV Max=118mV
ok: Voltage[19.110] Name=SB#0_+1.0V_CACHE#0 Current=102mV Min=56mV Max=139mV
ok: Voltage[19.111] Name=SB#0_+1.0V_CACHE#1 Current=95mV Min=56mV Max=139mV
ok: Voltage[19.112] Name=SB#0_+1.0V_CORE#0 Current=80mV Min=45mV Max=145mV
ok: Voltage[19.113] Name=SB#0_+1.0V_CORE#1 Current=80mV Min=45mV Max=145mV
ok: Voltage[19.114] Name=SB#0_+12V#2 Current=0mV Min=1101mV Max=1295mV
ok: Voltage[20.96] Name=SB#1_+5.0VL Current=497mV Min=459mV Max=540mV
ok: Voltage[20.97] Name=SB#1_+3.3VL Current=328mV Min=300mV Max=357mV
ok: Voltage[20.98] Name=SB#1_+1.8VL Current=177mV Min=165mV Max=194mV
```

```
ok: Voltage[20.99] Name=SB#1_+1.2VL Current=117mV Min=110mV Max=129mV
...
```

4.8.15 Hardware – CPU Information

This is part of the "Hardware" information (chkhardware). This requires several seconds.

- Summary line:

cpus(<cpu_summary_status>)

- Long output in the case of WARNING or CRITICAL or verbose=2 or verbose=641:

For each CPU:

- o CPU status
- o CPU[<idx>] with <idx> as a combination of system board number and CPU number
- o Model
- o Speed in MHz

- Verbose sample

* CPU Table:

```
ok: CPU[19.0] Name=SB#0-CPU#0 Model="Intel(R) Xeon(R) Processor " Speed=2266MHz
prefailed-warning: CPU[19.1] Name=SB#0-CPU#1 Model="Intel(R) Xeon(R) Processor "
Speed=2266MHz
prefailed-warning: CPU[20.0] Name=SB#1-CPU#0 Model="Intel(R) Xeon(R) Processor "
Speed=2266MHz
ok: CPU[20.1] Name=SB#1-CPU#1 Model="Intel(R) Xeon(R) Processor " Speed=2266MHz
ok: CPU[21.0] Name=SB#2-CPU#0 Model="Intel(R) Xeon(R) Processor " Speed=2266MHz
prefailed-warning: CPU[21.1] Name=SB#2-CPU#1 Model="Intel(R) Xeon(R) Processor "
Speed=2266MHz
ok: CPU[22.0] Name=SB#3-CPU#0 Model="Intel(R) Xeon(R) Processor " Speed=2266MHz
ok: CPU[22.1] Name=SB#3-CPU#1 Model="Intel(R) Xeon(R) Processor " Speed=2266MHz
```

- User interface sample in Icinga Classic 1.5:

Service State Information

Current Status:	CRITICAL (for 9d 5h 16m 21s)
Status Information:	CRITICAL voltages(ok) cpus(failed) memory-modules(ok) Systemname=FTS5 Description="PRIMEQUEST Management Board" Location="██████████" "██████████" Contact="F██████████" Name=MCD3AC111 AdminURL=http://1██████████5:8081 Model="PRIMEQUEST 1800E" prefailed-warning: CPU[19.1] Name=SB#0-CPU#1 Model="Intel(R) Xeon(R) Processor " Speed=2266MHz prefailed-warning: CPU[20.0] Name=SB#1-CPU#0 Model="Intel(R) Xeon(R) Processor " Speed=2266MHz prefailed-warning: CPU[21.1] Name=SB#2-CPU#1 Model="Intel(R) Xeon(R) Processor " Speed=2266MHz
Performance Data:	
Current Attempt:	3/3 (HARD state)
Last Check Time:	10-17-2012 14:07:27
Check Type:	ACTIVE
Check Latency / Duration:	0.050 / 6.799 seconds
Next Scheduled Check:	10-17-2012 14:17:27
Last State Change:	10-08-2012 08:51:35
Last Notification:	10-17-2012 13:17:39 (notification 212)
Is This Service Flapping?	N/A
In Scheduled Downtime?	NO
Last Update:	10-17-2012 14:07:48 (0d 0h 0m 8s ago)

4.8.16 Hardware – Memory Module Information

This is part of the "Hardware" information (chkhardware).

- Summary line:

memory-modules(<status>)

- Long output for verbose level 651
ATTENTION: This requires more than 20 seconds. Memory modules with the status "not-present" will be ignored.
 Because this check needs too much time, it is not done via simple verbose=2 or in non-OK cases!

Memory module table.

Sample:

```
* Memory Modules Table:
ok: MemMod[19.0] Name=SB#0-DIMM#0A0 Capacity=2048MB
ok: MemMod[19.4] Name=SB#0-DIMM#0B0 Capacity=2048MB
ok: MemMod[19.8] Name=SB#0-DIMM#0C0 Capacity=2048MB
ok: MemMod[19.12] Name=SB#0-DIMM#0D0 Capacity=2048MB
ok: MemMod[19.16] Name=SB#0-DIMM#1A0 Capacity=2048MB
ok: MemMod[19.20] Name=SB#0-DIMM#1B0 Capacity=2048MB
ok: MemMod[19.24] Name=SB#0-DIMM#1C0 Capacity=2048MB
ok: MemMod[19.28] Name=SB#0-DIMM#1D0 Capacity=2048MB
ok: MemMod[20.0] Name=SB#1-DIMM#0A0 Capacity=2048MB
ok: MemMod[20.4] Name=SB#1-DIMM#0B0 Capacity=2048MB
ok: MemMod[20.8] Name=SB#1-DIMM#0C0 Capacity=2048MB
ok: MemMod[20.12] Name=SB#1-DIMM#0D0 Capacity=2048MB
ok: MemMod[20.16] Name=SB#1-DIMM#1A0 Capacity=2048MB
ok: MemMod[20.20] Name=SB#1-DIMM#1B0 Capacity=2048MB
ok: MemMod[20.24] Name=SB#1-DIMM#1C0 Capacity=2048MB
ok: MemMod[20.28] Name=SB#1-DIMM#1D0 Capacity=2048MB
...
```

4.9 PRIMERGY Server – Monitoring Update Status

Manual about Update Agent Status: "ServerView System Monitor"

4.9.1 Option

Special option of the script 'check_fujitsu_server.pl' script:

--chkupdate

With this option the update server status can be obtained via the ServerView Update Agent MIB (SVUpdate.mib).

4.9.2 Summary Information Print

This script only returns the status if ServerView Update Agent SNMP information is available (ServerView Agent V6.20 or higher):

For systems without this SNMP data, an error note on missing information will be printed beside the "UNKNOWN" result.

4.9.3 Sample Nagios Configuration

```
define hostgroup {
    hostgroup_name    primergy-update-monitor
    alias              Fujitsu PRIMERGY Servers Update Monitoring
    register          1
}
```

```

define service {
    hostgroup_name      primergy-update-monitor
    service_description  SV Update Status
    servicegroups        serverview
    use                  generic-service
    check_command         check_primergy! --chkupdate
    flap_detection_enabled 1
    register              1
}

```

4.10 Check of PRIMERGY Servers via iRMC SNMP

There are three known sources of information to be monitored which can be fetched using the iRMC address for connection

- iRMC / IPMI data independent on each installation on the server to be monitored
- Additional information from the ServerView Agent installed on the server to be monitored
- Additional information from the ServerView Agentless Service installed on the server to be monitored

4.10.1 Options

About addressing via iRMC see `-A|--admin=` option.

Optional additional options to select the range of information:

--chkssystem

"System" means anything other than "Environment" or "Power"

This system information can be split with following options.

ATTENTION: `--chkstorage` is ignored currently because there is no information available

--chkhardware

"Hardware" (voltage, CPU and memory modules). This is now part of "System".

This is only an option for those who want to monitor only these components only.

The system hardware information can be split with following options.

--chkcpu	CPU information
--chkvoltage	Voltage information
--chkmemmodule	Memory modules information

--chkenv

"Environment" range - meaning fan and temperature sensors

The environment information can be split with following options instead of `chkenv`.

--chkenv-fan	Fan
--chkenv-temp	Temperature sensor, includes performance data

--chkpower

"Power" range: Monitoring power supply units and get power consumption summary if available.

If none of these are set than these three ranges will be checked.

ATTENTION: `--chkdrvmonitor` is ignored currently because there is no information available

4.10.2 System Information in Case of WARNING or CRITICAL - for Notifications

- Serial Identification - Additional information In case of errors or for Verbose-Level 1:
- RFC1213 Information - Additional information In case of errors or for Verbose-Level 1:
 - Server System Name
 - Description String
 - Location (if defined)
 - Contact (if defined)

- System Unit Information - Additional information In case of errors or for Verbose-Level 1:
 - Model (if available)
 - Administrative URL (if available)
 - Management IP (if available)

The following is only printed for Verbose Level 2:

- Server Unit Name
 - Location (if defined)
 - Contact (if defined)
- Notify Sample or Verbose-Level 1 sample:
 ID=YLNT000533 Systemname=RX300-S82-iRMC Description="Linux RX300-S82-iRMC
 2.6.28.10-ami #1 Thu Mar 13 18:12:17 CET 2014 armv5tejl" Location="Unknown"
 Contact="root@localhost" AdminURL=http://RX300-S82-iRMC.svsnet.qanet:80
 Model="PRIMERGY RX300 S8" SpecifiedAddress=nnn.nnn.nnn.41
 AdminAddress=nnn.nnn.nnn.141

4.10.3 System Status Information

- Summary Line
 - Nagios Status String
 - Serial Number (ID)
 - Status of each Subcomponent.
 If chkssystem is selected only subcomponents other than "Environment" and "Power" are printed

If option chkssystem is selected than "Hardware components assigned to the system board such as CPU, Voltage and Memory-Modules are monitored.

For more about these see in chapters below.

- See "4.10.8 Hardware – Voltage Information" on page 65
 - See "4.10.9 Hardware – CPU Information" on page 65
 - See "4.10.10 Hardware – Memory-Module Information" on page 66
- Sample
 OK - ID=YLSK000406 - SystemBoard(ok)

4.10.4 Power Supply Information

This is part of the "Power" information (chkpower).

- Summary Line
PowerSupplies(<status>)
 with summary status of all power supplies.
- Long output in case of WARNING or CRITICAL or verbose=2 output:
 For each power supply unit:
 - PSU status
 - PSU[<idx>] with <idx> as number of the power supply unit
 - Name
 - Current load and maximum load in watts

Some parts might not be available.

- Verbose sample
 * Power Supplies:
 ok: PSU[1] Name=PSU1 CurrentLoad=80Watt Max=448Watt

4.10.5 Power Consumption Information

This is part of the "Power" information (chkpower).

- Performance data
PowerConsumption=<average>Watt
- Sample
...
| PowerConsumption=80Watt

4.10.6 Fan Information

This is part of the "Environment" information (chkenv).

- Summary line for the summary monitoring:
There is only the combined information of all Fans and Temperature Sensors:
Environment(<state>)
- Summary line for the chkfan usage
Fans(<state>)
- Long output in case of WARNING or CRITICAL or verbose=2 output:
For each Fan:
 - Fan status
 - Fan[<idx>] with <idx> as number of the fan.
 - Name
 - Speed in rpm ("rotation per minute")
- Verbose sample
* Fans:
ok: Fan[1] Name=FAN1_SYS Speed=5700rpm
ok: Fan[2] Name=FAN2_SYS Speed=5700rpm
ok: Fan[3] Name=FAN3_SYS Speed=6120rpm
ok: Fan[4] Name=FAN4_SYS Speed=5700rpm
ok: Fan[5] Name=FAN5_SYS Speed=5820rpm
ok: Fan[6] Name=FAN_PSU1 Speed=1440rpm

4.10.7 Temperature Sensors Information

This is part of the "Environment" information (chkenv).

- Summary line for the summary monitoring:
There is only the combined information of all Fans and Temperature Sensors:
Environment(<state>)
- Summary line for the chktemp usage:
Temperatures(<state>)
- Long output in case of WARNING or CRITICAL or verbose=2 output:
For each Sensor:
 - Temperature sensor status
 - Sensor[<idx>] with <idx> as number of the sensor
 - Name
 - Current value and warning level and critical level in Celsius

Sensors with status "not-available" will be ignored.
- Performance data:
<identification of sensor>=<current>C;<warn>;<critical>

- Verbose Sample:
 - * Temperature Sensors:
 - ok: Sensor[1] Name=Ambient Temperature=23C Warning=37C Critical=42C
 - ok: Sensor[2] Name=Systemboard_1 Temperature=24C Warning=75C Critical=80C
 - ok: Sensor[3] Name=Systemboard_2 Temperature=27C Warning=75C Critical=80C
 - ok: Sensor[4] Name=CPU1 Temperature=27C Warning=90C Critical=91C
 - ok: Sensor[5] Name=CPU2 Temperature=30C Warning=90C Critical=91C
 - ok: Sensor[6] Name=MEM_A Temperature=26C Warning=78C Critical=82C
 - ok: Sensor[9] Name=MEM_D Temperature=28C Warning=78C Critical=82C
 - ok: Sensor[12] Name=PSU1_Inlet Temperature=24C Warning=57C Critical=61C
 - ok: Sensor[14] Name=PSU1 Temperature=49C Warning=102C Critical=107C
 - | Ambient=23C;37;42 Systemboard_1=24C;75;80 Systemboard_2=27C;75;80
 - CPU1=27C;90;91 CPU2=30C;90;91 MEM_A=26C;78;82 MEM_D=28C;78;82 PSU1_Inlet=24C;57;61
 - PSU1=49C;102;107

4.10.8 Hardware – Voltage Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkssystem:
 - ... **SystemBoard(<combinedstatus>)** ...
 - The status is the combination of at least all CPU, Voltage and MemoryModules
- Summary line for chkhardware or chkvoltage:
 - ... **Voltages(<all_voltage_unit_Status>)** ...
- Performance data
 - ATTENTION:** Currently no performance data is generated for the Nagios tool.
This data is only visible with verbose mode.
- Long output in case of WARNING or CRITICAL or verbose=2:
 - For each Voltage Unit:
 - Voltage status
 - Voltage[idx] with <idx> as number of voltage
 - Name
 - Current, minimum and maximum values in mV
- Verbose sample
 - ...
 - * Voltages:
 - ok: Voltage[1] Name=BATT_3.0V Current=3075mV Min=2010mV Max=3495mV
 - ok: Voltage[2] Name=STBY_12V Current=11594mV Min=11304mV Max=12986mV
 - ok: Voltage[3] Name=STBY_5V Current=5074mV Min=4628mV Max=5421mV
 - ok: Voltage[4] Name=STBY_3.3V Current=3218mV Min=3040mV Max=3575mV
 - ok: Voltage[5] Name=iRMC_1.8V_STBY Current=1750mV Min=1672mV Max=1927mV
 - ok: Voltage[6] Name=iRMC_1.5V_STBY Current=1450mV Min=1392mV Max=1607mV
 - ok: Voltage[7] Name=MAIN_12V Current=12159mV Min=11311mV Max=12901mV
 - ok: Voltage[8] Name=MAIN_5V Current=4876mV Min=4628mV Max=5421mV
 - ok: Voltage[9] Name=MAIN_3.3V Current=3283mV Min=3040mV Max=3575mV
 - ok: Voltage[10] Name=PCH_1.5V Current=1431mV Min=1401mV Max=1607mV
 - ok: Voltage[11] Name=PCH_1.1V Current=1070mV Min=1021mV Max=1178mV

4.10.9 Hardware – CPU Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkssystem:
 - ... **SystemBoard(<combinedstatus>)** ...
 - The status is the combination of at least all CPU, Voltage and MemoryModules

- Summary line for chkhardware or chkcpu:
...CPUs(<cpus_status>)...
- Long output in case of WARNING or CRITICAL or verbose=2:
For each CPU:
 - CPU status
 - CPU[<idx>] with <idx> as number of CPU
 - Name
 - Model (if available)
 - Speed in MHz
- Verbose sample
* CPU Table:
unknown: CPU[1] Name=CPU1 Speed=2500MHz
unknown: CPU[2] Name=CPU2 Speed=2500MHz

4.10.10 Hardware – Memory-Module Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkssystem:
... SystemBoard(<combinedstatus>) ...
The status is the combination of at least all CPU, Voltage and MemoryModules
- Summary line for chkhardware or chkmemmodule:
...MemoryModules(<status>)...
- Long output in case of WARNING or CRITICAL or verbose=2
For each MemoryModule
 - Memory status
 - Memory[<idx>] with <idx> as number of Memory Module
 - Name
 - Type
 - Capacity in Mbytes
 - Frequency and maximum frequency in MHz
 MemoryModules with SNMP status "not-present" will be ignored.
- Verbose sample
* Memory Modules Table:
ok: Memory[1] Name=DIMM-1A Type="DDR3 / RDIMM" Capacity=8192MB Frequency=1600MHz
Frequency-Max=1600MHz
ok: Memory[7] Name=DIMM-1D Type="DDR3 / RDIMM" Capacity=8192MB Frequency=1600MHz
Frequency-Max=1600MHz

4.11 Memory Usage - Special Performance Check for PRIMERGY Servers

4.11.1 Option

Special option of the 'check_fujitsu_server.pl' script:

--chkmemperf

With this option the physical and virtual memory usage data can be obtained via the "PerformanceTable" of SC2.

-w <percent> -c <percent>

Add WARNING or CRITICAL level for file systems where the usage level can be monitored. These should be simple percentage values (0..100).

ATTENTION: This script sets no default values for this.

ATTENTION: This option cannot be combined with other check options.

4.11.2 Summary Information Print

This script prints the following data:

- Physical memory in percent – send also as performance data
- Physical memory in Mbytes
- Virtual memory in Mbytes

4.11.3 Sample Nagios Configuration

```
define service {
    hostgroup_name      primergy-detail-servers,primergy-servers
    service_description  SV Memory Usage
    servicegroups        serverview
    use                  generic-service,perfdata-service
    check_command        check_fujitsu_server!-chkmemperf $_HOSTSV_MEMPERF_OPTIONS$
    flap_detection_enabled 1
    register             1
}
```

In the host definition, **_SV_MEMPERF_OPTIONS** could be set to '-w90 -c95' as a sample for WARNING level 90% and CRITICAL level 95%.

4.12 File System Checks - Special Performance Check for PRIMERGY Server

4.12.1 Option

Special option of the 'check_fujitsu_server.pl' script:

--chkfsp

Check INVENT.mib file system entries and get performance data.

-w <percent> -c <percent>

Add WARNING or CRITICAL level for file systems where the usage level can be monitored. These should be simple percentage values (0..100).

ATTENTION: This script sets no default values for this.

ATTENTION: This option cannot be combined with other check options.

4.12.2 File System Status and Performance Printouts

The status will be OK if file systems can be detected and no limits are set or if no file system exceeds the limits.

The non-OK file system information will be added to the long output.

The usage percentage performance data will be set for any file system with monitorable values.

HINT: For names used in performance data:

For empty file system names, "FS_<index>" will be used.

For LINUX the name "/" will be renamed "ROOT".

Special characters like '(', ') and ':' will be changed in the performance output.

For Windows names the file system character will be set as the prefix (where possible).

4.12.3 File System Verbose or Non-OK Information

The following information is available in the case of a non-OK status or for Verbose Level 2.

- FS[<idx>] with <idx> as index of the file system
- Name of the file system (as-is in SNMP)
- Type
- Use in percent – send also as performance data
- Size in KB or MB depending on SNMP settings
- Available space in KB or MB depending on SNMP settings
- Mount mode

4.12.4 Sample Nagios Configuration

```
define service {
    hostgroup_name      primergy-detail-servers,primergy-servers
    service_description  SV File Systems
    servicegroups        serverview
    use                  generic-service,perfddata-service
    check_command         check_fujitsu_server!-chkfsperf
    $_HOSTSV_FSPERF_OPTIONS$
    flap_detection_enabled 1
    register              1
}
```

In the host definition, **_SV_FSPERF_OPTIONS** could be set to '-w90 -c95' as a sample for WARNING level 90% and CRITICAL level 95%.

4.13 Network Interface Checks - Special Performance Check for PRIMERGY Servers

4.13.1 Option

Special option of the 'check_fujitsu_server.pl' script:

--chknetperf

Check INVENT.mib network interface entries and get performance data.

-w <kBytesec> -c <kBytesec >

Add WARNING or CRITICAL level for interfaces where the In/Out bytes can be monitored. These should be simple Kbytes/sec limits.

ATTENTION: This script sets no default values for this.

ATTENTION: This option cannot be combined with other check options.

4.13.2 Status and Performance Printouts

The status will be OK if no WARNING or CRITICAL limits are set or if no network interface exceeds the limits.

The non-OK network interface information will be added to the long output.

The network interface performance data will be set for any network interface with monitorable values – in this case only if the nominal speed is known.

HINT: For used names in performance data:

NetIF[index] will be used if no short name can be extracted from the description field or connection field.

Loopback[index] will be used if the connection field contains the word "Loopback" (if monitorable).

vEthernet[index] will be used if the connection field contains the word "vEthernet" (if monitorable).

LAN[index] will be used if the connection field contains the word "LAN".

LocalAreaConnection[index] will be used if the connection field contains the term " Local Area Connection".

<CustomerDefinedName>[index] will be used if the connection field contains no spaces and is UTF8.

4.13.3 Network Interface Verbose or Non-OK Information

The following information is available in the case of a non-OK status or for Verbose Level 2.

- NetIF[<idx>] with <idx> as index of the network interface
- Description
- Adapter (if available)
- Connection (if available)
- Usage (if available – in tests this was not yet defined)
- Nominal speed in Kbytes/sec (for monitorable interfaces)
- BytesInOut Kbytes/sec – send also as performance data

4.13.4 Sample Nagios Configuration

```
define service {
    hostgroup_name      primergy-detail-servers,primergy-servers
    service_description  SV Network Interface
    servicegroups        serverview
    use                  generic-service,perfdata-service
    check_command         check_fujitsu_server!-chknetperf
    $_HOSTSV_NETPERF_OPTIONS$
    flap_detection_enabled 1
    register              1
}
```

In the host definition, **_SV_NETPERF_OPTIONS** could be set. This requires knowledge about the KB/sec values for EACH host.

4.14 Other Features

4.14.1 Plug-in for PRIMERGY Server and CPU Load

Hidden ability of the script 'check_fujitsu_server.pl':

With the option '**--chkcpuload**', the CPU load performance data can be read via the "PerformanceTable" of SC2.

BUT it makes no sense to monitor every 10 minutes and store these values, because they vary extremely every time.

ATTENTION: This option cannot be combined with other check options

This can be used as a tool in case of emergencies to check the current performance of CPUs.

5 FUJITSU CIM Server Plug-in: `check_fujitsu_server_CIM.pl`

FUJITSU all Nagios plugins should return four return values for **OK**, **WARNING**, **ERROR** or **UNKNOWN** states and these plugins can print data on standard output.

These plugins are called as scheduled by the Nagios daemon. Schedule values can be set via the configuration.

The **`check_fujitsu_server_CIM.pl`** script uses the executable **`wbemcli`** or the **Perl bindings of OpenWSMAN** selectable by an option. The CIM data can only be fetched if authentication information like user/password is set. This can be done via a file – only readable for the owner of the FUJITSU CIM Server Plug-in – in which corresponding host-specific options are written.

5.1 Basics

5.1.1 Script Name

Name of the CIM script:

`check_fujitsu_server_CIM.pl`

This is a Perl script which uses `wbemcli` calls.

This script monitors FUJITSU servers by reading and analyzing CIM classes. This can be used for Nagios monitoring. Currently this script is able to check servers where ServerView Agent CIM providers are installed.

5.1.2 Options around the Plug-in itself

`-V|--version`

Print version information and help text.

`-h|--help`

Print help text.

5.1.3 Select Usage Mode – CIM-XML or WS-MAN

`-U|--usage={C|W}`

"C" use CIM-XML protocol. This requires `wbemcli` installation. Connect to SFCB or OpenPegasus.

"W" use WS-MAN protocol. This requires OpenWSMAN Perl binding. Connect to OpenWSMAN services or WinRM.

5.1.4 CIM Options around Addressing

`-H|--host=<name-or-ip>`

Host address as DNS name or IP address of the server

`[-A|--admin=<adminaddress>]`

Specify administration address with which all data for the host should be retrieved. This might be the iRMC address if a corresponding CIM provider is enabled on iRMC.

HINT:

The two addresses are meant for Nagios administrators who want to define one host and Nagios services which should use the origin `$HOSTADDRESS$` and parallel to these other Nagios services which should use the iRMC address e.g specified in `$_HOSTSV_ADMIN_OPTION$` or `$_HOSTSV_CIM_OPTIONS$`

-P|--port=<portnumber>

CIM service port number. The **wbemcli** internal default for WBEM https services is 5989. For WS-MAN calls the port number must be entered (e.g. 5985 or 5986)!

For WS-MAN:

- For 5985 the transport type http is default.
- For 5986 the transport type https is default.

-T|--transport=<type>

For CIM this is the transport HTTP type. Default for **wbemcli** is "https".

Other value:

- **http** – for simple http calls

For WS-MAN:

- For 5985 the transport type http is default.
- For 5986 the transport type https is default.

-u|--user=<user> -p|--password=<pwd>

Authentication credentials (Please write these options into a file – see option -I)

These options **must** be set.

ATTENTION:

1. Credentials are verified in *sfc* on managed node site and this verification is done according to *sfc* configuration. Usually credentials are checked by a library (*sfcBasicPAMAuthentication*) which defines the rules how credentials are checked. And these rules might not be as evaluated as one would expect (restrictions in use of characters or length of values etc.). Refer to *sfc* how to replace the standard library by one that meets your requirements if you suffer from some lack of functionality.
The same behavior is true for the other CIM services OpenPegasus, OpenWSMAN and WinRM.
2. Known problems:
 - For *wbemcli*: Passwords must not contain any dots '.'
 - Do not use characters like ' ' or " in the user or password string
 - User and passwords which contain shell relevant signs like \$ should always be set in an option input file see option description -I below
 - Password check might vary from system to system

--cacert=<cafile>

CA certificate file.

If not set *wbemcli* parameter -noverify will be used resp. the "do not verify" flags are set for OpenWSMAN.

See **wbemcli** parameter *-cacert* resp. *wsman* command parameter *--cacert*

ATTENTION:

1. Certificates are handled both in *wbemcli* and in *sfc* on managed node site. On managed node site this is done according to *sfc* configuration and usually certificates are checked under certain conditions by a library (*sfcCertificateAuthentication*) which defines the rules how certificates are checked. As for credential check you might want to replace this library by your own implementation. Please refer to *sfc* how to do so.
The same behavior is true for the other CIM services OpenPegasus, OpenWSMAN and WinRM.
2. Known problems:
 - For *wbemcli*: If <cafile> exists and is not empty but does not contain any certificate, on some systems *wbemcli* dumps a core.

--cert=<certfile> --privkey=<keyfile>

Client certificate file and client private key file.

wbemcli requires both file names if this is to be used.

It depends on configuration on the host side whether these certificates are verified or not!

See **wbemcli** the parameters *-clientcert* and *-clientkey* resp. *wsman* command and parameter *--cert* and *--sslkey*.

-I|--inputfile=<file> [--inputdir=<directory>]

File in which host-specific options like the above-named ones – with the exception of the host option – can be stored. Use of this is **recommended** for security-relevant options like *-u* and *-p*.

This file must be readable for the owner of the plugin script used by the Nagios scheduler from the Nagios plugin directory!

With *inputdir* a directory path or the input option file can be specified. The directory path is ignored if the input file starts with '/'.

5.1.5 Plugin Processing Control Option

-t|--timeout=<timeout in seconds>

Timeout for the script processing.

The script sets a 60-second timeout for ESXi servers and use of **wbemcli** for the first connection calls. If no timeout is given via this option, a maximum timeout of 120 seconds is set.

-v|--verbose=<verbose mode level>

Enable verbose mode (levels: 1,2). Generates multi-line output with inventory information or other additional information.

5.2 Common Rules for Printouts

The rules are identical with the ones described in the SNMP section to-be-monitored on page 31.

5.3 Tool Features

5.3.1 Simple CIM Identify Check

Option

--chkidentify

ATTENTION: This option cannot be combined with other check options.

With this option the script checks if the host is running and if the set authentication data is valid for receiving CIM responses from the host to be monitored.

For this check the **standard** CIM providers are used. This call is independent of installed ServerView CIM providers.

HINT: If this call fails, check the following potential reasons on the server side (to-be-monitored host):

- Is the server powered-on and the CIM service running
- Check the authentication data
- Check other permission or restriction rules in the configurations

5.3.2 Get Only System Information Used for Notify Information

Option

--systeminfo

ATTENTION: This option cannot be combined with other check options.

If one of the following checks returns WARNING or CRITICAL or if Verbose Level 1 is used, system information usable by administrators is collected (if available) and printed.

The printed data is server type dependent. The data is described in separate sections for each type:

PRIMERGY servers: see 5.4.2 "System Information in the Case of WARNING or CRITICAL - for Notifications" on page 74.

5.4 Check of PRIMERGY Servers where SV CIM Provider is installed

This is dependent on the existence and use of SVS_PGY classes. Each Server that supports these can be monitored with this script. The SVS_PGY-Classes are provided by the ServerView CIM Providers.

HINT: Depending on the server type and version of CIM support, it may occur that not all data can be retrieved.

ServerView Agent CIM Provider supports these classes.

ATTENTION: The current ESXi ServerView CIM Provider does not support the total power consumption value! Because of this the "Power" check only monitors the power supply status.

The current ESXi ServerView CIM Provider does not support DriverMonitor.

The current ESXi ServerView CIM Provider does not support RAID.

Current Linux and Windows ServerView CIM Provider do not support ServerView RAID but the state of the mass storage adapters are available.

Current Linux and Windows ServerView CIM Provider do not support "DriverMonitor" details but the state of the DriverMonitor adapters are available.

5.4.1 Options

There is no special option for selecting type "Primergy Server". This is the default for the script.

Optional additional options:

--chksystem

Select the range of information. System meaning anything except "Environment" or "Power"

The system information can be split using the following options.

--chkstorage

Select only "MassStorage" data. This is now part of "System".

This is an option for those who want to monitor this component only.

--chkhardware

"Hardware" (voltage, CPU and memory modules). This is now part of "System".

This is an option for those who want to monitor these components only.

The system hardware information can be split with the following options.

--chkcpx CPU information

--chkvoltage Voltage information

--chkmemmodule Memory modules information

--chkenv

"Environment" range if information meaning Fan and Temperature Sensors

The environment information can be split with the following options instead of *chkenv*.

--chkenv-fan Fan

--chkenv-temp Temperature sensor, includes performance data

--chkpower

"Power" range: Monitoring power supply units and get power consumption summary if available.

If none of these are set, than these three ranges will be checked.

--chkdrvmonitor

Print "DirverMonitor" information if available. This might be added to above named options. Only with this option the Driver-Monitor status information might be visible in case of non-OK status.

5.4.2 System Information in the Case of WARNING or CRITICAL - for Notifications

- Serial identification - Additional information in the case of errors or for Verbose Level 1:
- Other information - Additional information in the case of errors or for Verbose Level 1:
 - Serial number (ID) (if available)
 - Server system name
 - Description string (if available)
 - Location (if defined)
 - Contact (if defined)
 - Model
 - Housing type
 - Administrative URL (if available)
 - Parent MMB address (if host is a server blade)
- Additional inventory information
 - Operating system (OS) and OS revision
- Notify sample or Verbose Level 1 sample:


```
ID=Y LAPXXXXX Name=CX122S1HE11.svsnet.qanet Description="System Asset Tag"
Model="PRIMERGY CX122 S1" Housing="CX122S1R" Contact="root@localhost"
AdminURL=http://nnn.nnn.nnn.nnn OS="VMware ESXi 5.0.0 build-768111"
```

5.4.3 System Status Information

This is part of the "system" information (chkssystem).

- Summary line
 - Nagios status string
 - Serial number (ID)
 - Status of each subcomponent.

If chkssystem is selected, only subcomponents other than "Environment" and "Power" are printed.

If the chkssystem option is selected than SystemBoard assigned "Hardware" components assigned to the system board such as CPU, Voltage and Memory-Modules are monitored.

For more on these, see the following sections:

- See 5.4.10 "Hardware – CPU Information" on page 77
- See 5.4.9 "Hardware – Voltage Information" on page 77
- See 5.4.11 "Hardware – Memory Module Information" on page 78
- With Verbose Level 1 or in if the above-named component has a non-OK status, the **additional** status components are printed – e.g. "Deployment". Currently there is no detailed information available on these components. If only some of them are not OK check the server information via the administrative URL.

- Sample


```
OK - ID=YKHJXXXXXX - MassStorage(ok) SystemBoard(ok)
```

5.4.4 Additional System Information

With option *verbose=3* in addition to *-chkssystem*, additional information is available for diagnostic or inventory reasons. This can also be obtained with *verbose=200*, where other "Check System" verbose prints are omitted.

The following information can be retrieved:

- ServerView agent information
- System board inventory information
- IP addresses
- MAC addresses
- UUIDs

Sample:

```

...
AgentInfo - Name="ServerView CIM Provider" Version=6.21.00 Company="Fujitsu
Technology Solutions"
* System Board Table:
SystemBoard[] - ID=33735374 Name=SB#0 Product="S26361-D2899-D11" Model="D2899"
Manufacturer="FUJITSU"
* IP Addresses:
IPProtocol[] - IP=127.0.0.1 Name=lo0-1-ipv4 AssignedTo=ComputerSystem
IPProtocol[] - IP=nnn.nnn.nnn.221 Name=vmk0-2-ipv4 AssignedTo=ComputerSystem
IPProtocol[] - IP=nnn.nnn.nnn.222 Name=IPMI-ipv4-0 AssignedTo=ManagementController
IPProtocol[] - IP=fd7d:hhhh:hhhh:hhhh:hhhh:hhhh:hhhh:hhhh Name=IPMI-ipv6-0
AssignedTo=ManagementController
* MAC Addresses:
LAN[] - MAC=00:00:7F:00:00:01 Name=lo0 AssignedTo=ComputerSystem
LAN[] - MAC=00:19:99:83:A7:4D Name=vmk0 AssignedTo=ComputerSystem
LAN[] - MAC=00:19:99:69:FE:FC Name=IPMI-0 AssignedTo=ManagementController
* Chassis UUID:
Chassis[] - UUID=6111BB8A-A06A-11DF-A847-00199983A74C AssignedTo=ComputerSystem

```

5.4.5 Fan Information

This is part of the "Environment" information (chkenv).

- Summary line for chkenv:
Environment(<combinedstatus>)
This is the combined information of all Fans and Temperature Sensors.
- Summary line for chkenv-fan:
Fans (<state>)
- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each fan:

- Fan status
- Fan[<idx>] with <idx> as device ID of the fan
- Name
- Speed in rpm (rotations per minute)
- The relation to the normal value (if available)

- Verbose sample
* Fans:
ok: Fan[0000] Name=FAN1_SYS Speed=2340rpm
ok: Fan[0001] Name=FAN2_SYS Speed=2280rpm
ok: Fan[0002] Name=FAN3_SYS Speed=2280rpm
ok: Fan[0003] Name=FAN4_SYS Speed=2340rpm
ok: Fan[0004] Name=FAN5_SYS Speed=2340rpm
ok: Fan[0005] Name=FAN6_SYS Speed=2220rpm
ok: Fan[0006] Name=FAN_PSU1 Speed=6640rpm
ok: Fan[0007] Name=FAN_PSU2 Speed=6640rpm

5.4.6 Temperature Sensor Information

This is part of the "Environment" information (chkenv).

- Summary line for chkenv:
Environment(<combinedstatus>)
This is the combined information of all Fans and Temperature Sensors.
- Summary line for chkenf-temp:
Temperatures(<state>)

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each sensor:

- Temperature sensor status
- Sensor[<idx>] with <idx> as device ID of the sensor
- Name
- Current value and WARNING level and CRITICAL level in degrees Celsius

- Performance data:

<identification of sensor>=<current>C;<warn>;<critical>

- Verbose sample:

* Temperature Sensors:

```
ok: Sensor[0000] Name=Ambient Temperature=20C Warning=37C Critical=42C
ok: Sensor[0001] Name=Systemboard_1 Temperature=22C Warning=75C Critical=80C
ok: Sensor[0002] Name=Systemboard_2 Temperature=27C Warning=75C Critical=80C
ok: Sensor[0003] Name=CPU1 Temperature=28C Warning=77C Critical=81C
ok: Sensor[0004] Name=CPU2 Temperature=29C Warning=77C Critical=81C
ok: Sensor[0005] Name=MEM_A Temperature=23C Warning=78C Critical=82C
ok: Sensor[0009] Name=MEM_E Temperature=24C Warning=78C Critical=82C
ok: Sensor[0013] Name=PSU1_Inlet Temperature=24C Warning=57C Critical=61C
ok: Sensor[0014] Name=PSU2_Inlet Temperature=24C Warning=57C Critical=61C
ok: Sensor[0017] Name=PSU1 Temperature=49C Warning=102C Critical=107C
ok: Sensor[0018] Name=PSU2 Temperature=50C Warning=102C Critical=107C
| Ambient=20C;37;42 Systemboard_1=22C;75;80 Systemboard_2=27C;75;80 CPU1=28C;77;81
CPU2=29C;77;81 MEM_A=23C;78;82 MEM_E=24C;78;82 PSU1_Inlet=24C;57;61
PSU2_Inlet=24C;57;61 PSU1=49C;102;107 PSU2=50C;102;107
```

5.4.7 Power Supply Information

This is part of the "Power" information (chkpower).

- Summary line

PowerSupplies(<status>)

with summary status of all power supplies.

- Long output in the case of WARNING or CRITICAL or verbose=2 output:

For each power supply unit:

- PSU status
- PSU[<idx>] with <idx> as device ID of the power supply sensor
- Name
- Current load in watts if available
- Maximum load in watts

- Verbose sample

* Power Supplies:

```
ok: PSU[0000] Name=PSU Max=500Watt
```

5.4.8 Power Consumption Information

This is part of the "Power" information (chkpower).

- Long output for verbose=2:

List of all power consumption parts and their details.

- Performance data

PowerConsumption=<average>Watt

- Verbose sample

* Power Consumption:

```
ok: PowerCons[0-224-0] Name=Total_Power CurrentReading=24Watt
| PowerConsumption=24Watt
```

5.4.9 Hardware – Voltage Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkhardware:
... **Voltages(<all_voltage_unit_Status>)** ...
- Summary line for chkssystem:
... **SystemBoard(<combinedstatus>)** ...
The status includes at least all CPUs, voltages and memory modules.
- Performance data

ATTENTION: Currently there will be no performance data generated for the Nagios performance tool.
It is only visible with verbose data.

- Long output in the case of WARNING or CRITICAL or verbose=2:

For each voltage unit:

- Voltage status
- Voltage[idx] with <idx> as device ID of the voltage sensor
- Name
- Current value in V
- If available: critical and max value in V

- Verbose sample

```
...
* Voltages:
ok: Voltage[0-0] Name=BATT_3.0V Current=3.24V Critical=2.01V Max=3.49V
ok: Voltage[0-1] Name=STBY_5V Current=5.12V Critical=4.59V Max=5.4V
ok: Voltage[0-2] Name=STBY_3.3V Current=3.25V Critical=3.04V Max=3.59V
ok: Voltage[0-3] Name=LAN_1.9V_STBY Current=1.89V Critical=1.77V Max=2.04V
ok: Voltage[0-4] Name=iRMC_1.5V_STBY Current=1.48V Critical=1.39V Max=1.61V
ok: Voltage[0-5] Name=MAIN1_12V Current=12.11V Critical=11.25V Max=12.96V
ok: Voltage[0-6] Name=MAIN2_12V Current=12.11V Critical=11.25V Max=12.96V
ok: Voltage[0-7] Name=MAIN_5V Current=5.12V Critical=4.59V Max=5.43V
ok: Voltage[0-8] Name=MAIN_3.3V Current=3.3V Critical=3.03V Max=3.58V
ok: Voltage[0-9] Name=SFR_1.8V Current=1.78V Critical=1.67V Max=1.93V
ok: Voltage[0-10] Name=MEM_1.5V Current=1.48V Critical=1.25V Max=1.61V
ok: Voltage[0-11] Name=PCH_1.05V Current=1.04V Critical=0.97V Max=1.13V
```

5.4.10 Hardware – CPU Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkhardware:
... **CPUs(<cpus_status>)**...
- Summary line for chkssystem:
... **SystemBoard(<combinedstatus>)** ...
The status includes at least all CPUs, voltages and memory modules.
- Long output in the case of WARNING or CRITICAL or verbose=2:

For each CPU:

- CPU status
- CPU[<idx>] with <idx> as device ID of the CPU
- Name
- Model
- Speed in MHz

- Verbose sample

```
* CPU Table:
ok: CPU[0000] Name=SOCKET_0 Model="Intel(R) Xeon(TM)" Speed=2600MHz
ok: CPU[0001] Name=SOCKET_1 Model="Intel(R) Xeon(TM)" Speed=2600MHz
```

5.4.11 Hardware – Memory Module Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkhardware:
...**MemoryModules(<status>)**...
- Summary line for chkssystem:
... **SystemBoard(<combinedstatus>)** ...
The status includes at least all CPUs, voltages and memory modules.
- Long output in the case of WARNING or CRITICAL or verbose=2:

For each memory module:

- Memory status
- Memory[<idx>] with <idx> as "tag" index of memory module
- Name
- Type
- Serial number
- Capacity in Mbytes
- If available: Current and max frequencies

- Verbose sample

* Memory Modules Table:

ok: Memory[0] Name=Node0_Dimm0 Type="DDR3" ID=DC924DEE Capacity=4194304MB

ok: Memory[2] Name=Node1_Dimm0 Type="DDR3" ID=DC924E1D Capacity=4194304MB

5.5 Check of PRIMERGY Servers via iRMC CIM

There are three known sources of information to be monitored which can be fetched using the iRMC address for connection

- iRMC / IPMI data independent on each installation on the server to be monitored
- Additional information from the ServerView Agent installed on the server to be monitored
- Additional information from the ServerView Agentless Service installed on the server to be monitored

ATTENTION:

Tests with iRMC S4 and firmware version 7.32 have shown that there are no summary status values available. Because of this fact the script needs about 1 minute to retrieve all data and calculate corresponding summary status itself.

5.5.1 Options

About addressing via iRMC see --A|--admin= option.

Optional additional options to select the range of information:

--chkssystem

"System" means anything other than "Environment" or "Power"

This system information can be split with following options.

ATTENTION:

--chkstorage is ignored currently because there is no information available

--chkhardware

"Hardware" (voltage, CPU and memory modules). This is now part of "System".

This is only an option for those who want to monitor only these components only.

The system hardware information can be split with following options.

- chkcpu CPU information
- chkvoltage Voltage information
- chkmemmodule Memory modules information

--chkenv

"Environment" range - meaning fan and temperature sensors

The environment information can be split with following options instead of chkenv.

--chkenv-fan Fan

--chkenv-temp Temperature sensor, includes performance data

--chkpower

"Power" range: Monitoring power supply units and get power consumption summary if available.

If none of these are set than these three ranges will be checked.

ATTENTION:

--chkdrvmonitor is ignored currently because there is no information available

5.5.2 System Information in Case of WARNING or CRITICAL - for Notifications

Following data will be printed:

- Serial identification - Additional information In case of errors or for Verbose-Level 1:
- Other Information - Additional information In case of errors or for Verbose-Level 1:
 - Server system name (if available)
 - Specified host address if options -H and -A are used
 - Model (if available)
 - Housing type (if available)
 - Specified administration address if options -H and -A are used
- Notify sample or Verbose-Level 1 sample:
 ID=YLNT000533 Name=RX300S82 SpecifiedAddress=nnn.nnn.nnn.41 Model="PRIMERGY RX300 S8" Housing="RX300S8R4" AdminAddress=nnn.nnn.nnn.141

5.5.3 System Status Information

This is part of the "System" information (chkssystem).

- Summary Line
 - Nagios status string
 - Serial number (ID)
 - Status of each subcomponent.

If chkssystem is selected only subcomponents other than "Environment" and "Power" are printed

HINT:

Currently this will be only Systemboard unless iRMC supports MassStorage status monitoring

If option chkssystem is selected than "Hardware components assigned to the system board such a CPU, voltage and memory modules are monitored.

For more on these see the following sections:

- See 5.5.9 "Hardware – Voltage Information" on page 82
- See 5.5.10 "Hardware – CPU Information" on page 83
- See 5.5.11 "Hardware – Memory-Module Information" on page 83

- Sample

OK - ID=YLNT000533 - Systemboard(ok)

5.5.4 Additional System Information

With option verbose=3 additional to chkssystem additional information is available for diagnostic or inventory reasons. This can also be obtained with verbose=200 where other "Check System" verbose prints are omitted.

Following information can be retrieved

- Provider firmware version information
- System board inventory information
- IP addresses
- MAC addresses
- UUIDs

Sample:

```
...
ProviderFirmwareInfo - Version="7.34F, 7.32F"
* System Board Table:
SystemBoard[] - ID=41420274 Name=Systemboard Product="S26361-D2939-E17" Model="WGS08
GS52" Manufacturer="FUJITSU"
* IP Addresses:
IPProtocol[] - IP=fd7d:d06b:8656:1280:219:99ff:feff:6970
Name=IPProtocolEndpoint6*Channel-2*eth0-6
IPProtocol[] - Name=IPProtocolEndpoint5*Channel-2*eth0-6
IPProtocol[] - IP=10.172.130.141 Name=IPProtocolEndpoint4*Channel-2*eth0-4
IPProtocol[] - Name=SystemLAN_1_IPv6_0
IPProtocol[] - IP=fd7d:d06b:8656:1280:819e:9cc7:646d:1eff Name=SystemLAN_0_IPv6_0
IPProtocol[] - IP=10.172.130.140 Name=SystemLAN_0_IPv4_0
* MAC Addresses:
LAN[] - MAC=00:19:99:EF:63:D5 Name=SystemLAN_2
LAN[] - MAC=00:19:99:EF:63:D4 Name=SystemLAN_1
LAN[] - MAC=00:19:99:FD:69:70 Name=ManagementLAN_1
Chassis[] - UUID=cfe9a7b4-c6f9-e211-8add-4ddef388d411
```

5.5.5 Fan Information

This is part of the "Environment" information (chkenv).

- Summary Line for chkenv

Environment(<combinedstatus>)

This is the combined information of all Fans and Temperature Sensors:

- Summary Line for chkenv-fan

Fans(<state>)

- Long output in case of WARNING or CRITICAL or verbose=2 output:

For each fan:

- Fan status
- Name
- Speed in rpm (rotation per minute)
- The relation in percent to the nominal value (if available)

- Verbose sample

```
* Fans:
ok: Fan[] Name=FAN_PSU2 Speed=2400rpm NominalRelation=100%
ok: Fan[] Name=FAN_PSU1 Speed=2400rpm NominalRelation=100%
ok: Fan[] Name=FAN5_SYS Speed=5280rpm NominalRelation=97%
ok: Fan[] Name=FAN4_SYS Speed=4980rpm NominalRelation=97%
ok: Fan[] Name=FAN3_SYS Speed=5100rpm NominalRelation=98%
ok: Fan[] Name=FAN2_SYS Speed=5100rpm NominalRelation=97%
ok: Fan[] Name=FAN1_SYS Speed=5280rpm NominalRelation=96%
```


5.5.6 Temperature Sensors Information

This is part of the "Environment" information (chkenv).

- Summary Line for chkenv:
Environment(<combinedstatus>)
This is the combined information of all Fans and Temperature Sensors:
- Summary Line for chkenv-temp
Temperatures(<state>)
- Long output in case of WARNING or CRITICAL or verbose=2 output:
For each temperature sensor:
 - Temperature sensor status
 - Name
 - Current value and warning level and critical level in degree Celsius
- Performance data:
<identification of sensor>=<current>C;<warn>;<critical>
- Verbose sample:
* Temperature Sensors:
ok: Sensor[] Name=RAID_Controller Temperature=66C Warning=110C Critical=115C
ok: Sensor[] Name=BBU Temperature=26C Warning=50C Critical=55C
ok: Sensor[] Name=PSU2 Temperature=50C Warning=102C Critical=107C
ok: Sensor[] Name=PSU1 Temperature=50C Warning=102C Critical=107C
ok: Sensor[] Name=PSU2_Inlet Temperature=26C Warning=57C Critical=61C
ok: Sensor[] Name=PSU1_Inlet Temperature=26C Warning=57C Critical=61C
unknown: Sensor[] Name=MEM_H
unknown: Sensor[] Name=MEM_G
unknown: Sensor[] Name=MEM_F
ok: Sensor[] Name=MEM_E Temperature=28C Warning=78C Critical=82C
unknown: Sensor[] Name=MEM_D
unknown: Sensor[] Name=MEM_C
unknown: Sensor[] Name=MEM_B
ok: Sensor[] Name=MEM_A Temperature=27C Warning=78C Critical=82C
ok: Sensor[] Name=CPU2 Temperature=29C Warning=88C Critical=92C
ok: Sensor[] Name=CPU1 Temperature=27C Warning=88C Critical=92C
ok: Sensor[] Name=Systemboard_2 Temperature=32C Warning=75C Critical=80C
ok: Sensor[] Name=Systemboard_1 Temperature=26C Warning=75C Critical=80C
ok: Sensor[] Name=Ambient Temperature=25C Warning=37C Critical=42C
| RAID_Controller=66C;110;115 BBU=26C;50;55 PSU2=50C;102;107 PSU1=50C;102;107
PSU2_Inlet=26C;57;61 PSU1_Inlet=26C;57;61 MEM_E=28C;78;82 MEM_A=27C;78;82
CPU2=29C;88;92 CPU1=27C;88;92 Systemboard_2=32C;75;80 Systemboard_1=26C;75;80
Ambient=25C;37;42

5.5.7 Power Supply Information

This is part of the "Power" information (chkpower).

- Summary Line
PowerSupplies(<status>)
with summary status of all power supplies.
- Long output in case of WARNING or CRITICAL or verbose=2 output:
For each Power Supply Unit:
 - PSU status
 - Name
 - Current load in watts if available

- Verbose sample
 - * Power Supplies:
 - ok: PSU[] Name=PSU2_Power CurrentLoad=56Watt
 - ok: PSU[] Name=PSU1_Power CurrentLoad=52Watt

5.5.8 Power Consumption Information

This is part of the "Power" information (chkpower).

- Long output for verbose=2:
List of all power consumption parts and their details
- Performance data

PowerConsumption=<average>Watt

- Verbose sample
 - * Power Consumption:
 - ok: PowerCons[] Name=Total_Power_Out CurrentReading=80Watt
 - ok: PowerCons[] Name=Total_Power CurrentReading=108Watt
 - ok: PowerCons[] Name=PSU2_Power CurrentReading=56Watt
 - ok: PowerCons[] Name=PSU1_Power CurrentReading=52Watt
 - ok: PowerCons[] Name=HDD_Power CurrentReading=14Watt
 - ok: PowerCons[] Name=System_Power CurrentReading=36Watt
 - ok: PowerCons[] Name=CPU2_Power CurrentReading=18Watt
 - ok: PowerCons[] Name=CPU1_Power CurrentReading=10Watt
 - unknown: PowerCons[] Name=Power_Level
 - | PowerConsumption=108Watt

5.5.9 Hardware – Voltage Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkhardware:
... Voltages(<all_voltage_unit_Status>) ...
- Summary line for chkssystem:
... SystemBoard(<combinedstatus>) ...
The status includes at least all CPUs, voltage and memory modules
- Performance data
ATTENTION: Currently there will be no performance data generated for the Nagios performance tool.
Only with verbose data they are visible.
- Long output in case of WARNING or CRITICAL or verbose=2:
For each Voltage Unit:
 - Voltage status
 - Name
 - Current value in V
 - If available: Critical and max value in V
- Verbose sample
 - ...
 - * Voltages:
 - ok: Voltage[] Name=CPU2_1V Current=0.97V Critical=0.93V Max=1.06V
 - ok: Voltage[] Name=CPU1_1V Current=0.97V Critical=0.93V Max=1.06V
 - ok: Voltage[] Name=PCH_1.1V Current=1.09V Critical=1.02V Max=1.17V
 - ok: Voltage[] Name=PCH_1.5V Current=1.46V Critical=1.42V Max=1.58V
 - ok: Voltage[] Name=MAIN_3.3V Current=3.33V Critical=3.01V Max=3.56V
 - ok: Voltage[] Name=MAIN_5V Current=5V Critical=4.62V Max=5.42V
 - ok: Voltage[] Name=MAIN_12V Current=12.1V Critical=11.31V Max=12.9V
 - ok: Voltage[] Name=LAN_1.0V_STBY Current=0.97V Critical=0.93V Max=1.07V

```

ok: Voltage[] Name=iRMC_1.5V_STBY Current=1.45V Critical=1.39V Max=1.61V
ok: Voltage[] Name=LAN_1.8V_STBY Current=1.8V Critical=1.67V Max=1.93V
ok: Voltage[] Name=STBY_3.3V Current=3.31V Critical=3.01V Max=3.56V
ok: Voltage[] Name=STBY_5V Current=5.09V Critical=4.62V Max=5.42V
ok: Voltage[] Name=STBY_12V Current=11.75V Critical=11.28V Max=12.96V

```

5.5.10 Hardware – CPU Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkhardware:
...CPUs(<cpus_status>)...
- Summary line for chkssystem:
... SystemBoard(<combinedstatus>) ...
 The status includes at least all CPUs, voltage and memory modules
- Long output in case of WARNING or CRITICAL or verbose=2:
 For each CPU:
 - CPU status
 - Name
 - Model
 - Max clock and external bus clock speed in MHz
- Verbose sample
 * CPU Table:
 unknown: CPU[] Name=CPU2 Model="Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz"
 MaxClockSpeed=2800MHz ExternalBusClockSpeed=8000MHz
 unknown: CPU[] Name=CPU1 Model="Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz"
 MaxClockSpeed=2800MHz ExternalBusClockSpeed=8000MHz
 CKAE: There seems to be an error in 7.32 FW

5.5.11 Hardware – Memory-Module Information

This is part of the "Hardware" information (chkhardware and chkssystem).

- Summary line for chkhardware:
...MemoryModules(<status>)...
- Summary line for chkssystem:
... SystemBoard(<combinedstatus>) ...
 The status includes at least all CPUs, voltage and memory modules
- Long output in case of WARNING or CRITICAL or verbose=2
 For each memory module
 - Memory status
 - Name
 - Type
 - Capacity in Mbytes
- Verbose sample
 * Memory Modules Table:
 ok: Memory[2] Name=DIMM-1E Type="FBD2" Capacity=4096MB
 ok: Memory[0] Name=DIMM-1A Type="FBD2" Capacity=4096MB

6 FUJITSU SNMP Server Tool: tool_fujitsu_server.pl

This is an additional script which might be usable for interactive – not-scheduled – calls. See the following sections for its functionality.

6.1 Basics

6.1.1 Script Name

Name of the script
tool_fujitsu_server.pl

This is a Perl script which uses Net::SNMP calls.

6.1.2 Options around the Script Itself

-V|--version

Print version information and usage.

-h|--help

Print help text.

6.1.3 SNMP Options around Addressing

-H|--host=<name-or-ip>

Host address as DNS name or IP address of the server.

[-A|--admin=<adminaddress>]

Specify administration address with which all data for the host should be retrieved. This might be the iRMC address if a corresponding SNMP agent is enabled on iRMC.

HINT:

The two addresses are meant for Nagios administrators who want to define one host and Nagios services which should use the origin \$HOSTADDRESS\$ and parallel to these other Nagios services which should use the iRMC address e.g specified in
\$_HOSTSV_ADMIN_OPTION\$ or \$_HOSTSV_OPTIONS\$

-P|-p|--port=<portnumber>

SNMP port number. The default is 161.

-T|--transport=<type>

SNMP transport domain type. For a full description of available values, see *Net::SNMP->session* parameters and parameter '-domain'.

Default is "udp", meaning the UDP service for IPv4 addresses.

Samples for other values:

- tcp – for TCP connection instead of UDP
- udp6 or tcp6 – for IPv6 addresses

ATTENTION: For IPv6 requirements, see sections 3.2 "Requirements" on page 14 and 4.1.4 "About IPv6 Usage" on page 31.

-C|--community=<SNMP community string>

SNMP community of the server. Usable for SNMPv1 and SNMPv2. Default is public.

**-u|--user=<username> [--authpassword=<pwd>] [--authkey=<key>] [--authprot=<prot>]
[--privpassword=<pwd>] [--privkey=<key>] [--privprot=<prot>]**

SNMPv3 authentication credentials.

-I|--inputfile=<file> [--inputdir=<directory>]

File in which host specific options like the above named ones beside the host option can be stored. The usage of this is **recommended** for security relevant options like `-u` and the other SNMPv3 credentials.

This file must be readable for the owner of the plug-in script used by the Nagios scheduler out of the Nagios Plug-in directory!

With `inputdir` a directory path of the input option file can be specified. The directory path is ignored if the input file starts with `'/'`.

6.1.4 About IPv6 Usage

For requirements for IPv6, see sections 3.2 "Requirements" on page 14 and 2.4 "SNMP Support: Supported Servers and SNMP MIBs" on page 12.

Before using an IPv6 address, make sure that on the host to be monitored there is an SNMP which is able to handle IPv6 addresses:

- Check SNMP configuration settings
- Check firewalls
- Check IPv6 kernel abilities and configurations of the operating system itself

ATTENTION: On SLES10 there is an SNMP which does not fully support IPv6 and is unstable if called with IPv6!

If an IPv6 address is entered, then the Plug-in automatically sets the transport type to UDP-IPv6 (if not specified).

If an IPv4 address is entered and no `-T` option is set, the default transport type of local Perl Net::SNMP is used.

For DNS names as host address, the default transport type of local Perl Net::SNMP is used.

6.1.5 Plug-in Processing Control Option

-t|--timeout=<timeout in seconds>

Timeout for the script processing.

-v|--verbose=<verbose mode level>

Enable verbose mode (levels: 1,2). Generates multi-line output with inventory information or other additional information.

6.2 SNMP MIB Test – Test of Server Types

This should help find information about servers with SNMP to enable the assignment to Nagios 'host groups' and their Nagios 'services'.

6.2.1 Option

--mibtest

This is the default for this script.

This "mibtest" information test can be reduced with the following options.

--typetest

Check only information relevant to analyse type of server.
One part of this is

--connectiontest

Test only SNMP test connectivity using RFC1213 tests.

[-e or --extended]

With this special option the scan for servers is extended to search for following data:

- Parent management blade for a PRIMERGY server blade
- Parent multi-node name and model information (CX series)
- Fully Qualified Domain Name (FQDN) of a PRIMERGY server if available

[--nopp]

This is an option for suppressing in-between process result printouts. Only a kind of summary of available system information about the host is printed.

6.2.2 Common Print Rules

Any test starts with ">>> topic".

Any answer starts with " <<< status [additional information]".

The last lines are a kind of summary, starting with a summary status and information found about the server.

6.2.3 Initial Tests – Connection and RC1213.mib Tests

The first test is the SNMP connection. If this fails, a corresponding error message is printed and the script stops.

The second test is the test to get the SNMP uptime information (from RFC1213.mib), which should be available on all servers. If this fails, the access or permission rights must be checked. The script stops if this check fails.

The third call reads the system information available in RFC1213.mib.

Sample part on success:

```
>>> SNMP connect to host xxx.xxx.xxx.xxx
<<< OK
>>> RFC1213 - Uptime
<<< OK - UpTime = 257 days, 01:48:41.13
>>> RFC1213 - System Information
<<< OK:
Systemname=BX600-3 Description="BX600" Location="Xxx" Contact="Xxx"
```

This information is available for any server with enabled SNMP permission to get the data.

6.2.4 ServerView SNMP Agent – Test of Agent-Specific MIB

Test of

- Status.mib,
- BIOS.mib,
- INVENT.mib,
- SC.mib,
- SVUpdate.mib, and
- SC2.mib

After initial tests the Status.mib is checked to prove whether additional MIBs are available or not.

For servers with ServerView SNMP Agent installed, the following information is available.

Sample on success:

```
>>> SNMP connect to host xxx.xxx.xxx.xxx
<<< OK
>>> RFC1213 - Uptime
<<< OK - UpTime = 2 days, 05:06:39.25
>>> RFC1213 - System Information
<<< OK:
Systemname=RX200S52 Description="Hardware: x86 Family 6 Model 26 Stepping 4
AT/AT COMPATIBLE - Software: Windows Version 6.0 (Build 6002 Multiprocessor
Free)" Location="Xxxx" Contact="..."
>>> Status - Subsystem Status Info
<<< OK - SUMMARY(ok) Environment(ok) PowerSupply(ok) MassStorage(ok)
Systemboard(ok) Network(ok) DrvMonitor(ok)
>>> INVENT - Central Info
<<< OK - INVENT - Version=6.10 Name=RX200S52 OS="Windows Server 2008
Standard Service Pack 2"
>>> SC2 - Agent Info
<<< OK - Server with SV SNMP Agent - Agent="ServerView ServerControl 2
hardware monitoring agent" Version=6.10.05.13 Company="Fujitsu"
>>> SVUpdate
<<< OK
>>> SC SC - BIOS Version
<<< OK - BIOS - V4.6.5.4 R1.2.0 for D3239-Alx
>>> BIOS - Version and Diagnostic
<<< OK - BIOS - Version=4.6 Diagnostic-Bit-Status=0x0
>>> S31 - Agent Info
<<< FAILED
>>> MMB-COM-MIB - Chassis Unit Info
```

```

    <<< FAILED
>>> RAID - Overall Status
    <<< OK
OK
    Name           = RX200S52
    Components     = Environment PowerSupply MassStorage Systemboard Network
DrvMonitor
    OS             = Windows Server 2008 Standard Service Pack 2
    Type           = PRIMERGY with SV SNMP Agent
    Model          = PRIMERGY RX200 S5
    AdminURL       = http://xxx.xxx.xxx.xxx:80
    UpdateAgent    = Additional SNMP monitoring available
    RAID           = Status(ok)

```

For Non-PRIMERGY servers two lines might be different. Sample:

```

    Type           = Server with SV SNMP Agent
    Model          = ESPRIMO P900 0-Watt

```

6.2.5 PRIMERGY Blade – Test of S31.mib

After initial tests and after the Status.mib check, it is checked whether S31 information is available.

The following information is available only for PRIMERGY Blade MMBs.

Sample on success:

```

>>> SNMP connect to host xxx.xxx.xxx.xxx
    <<< OK
>>> RFC1213 - Uptime
    <<< OK - UpTime = 22 days, 00:05:16.25
>>> RFC1213 - System Information
    <<< OK:
Systemname=BX400-VIOM2 Description="PRIMERGY BX400 S1 ABN:K1335-V200"
Location="..." Contact="Xxx"
>>> Status - Subsystem Status Info
    <<< FAILED
>>> S31 - Agent Info
    <<< OK - Blade - Name=BX400-VIOM2 IP=xxx.xxx.xxx.xxx
AdminURL=http://xxx.xxx.xxx.xxx:80 DateTime="10/11/2012 14:44:39"
>>> S31 - SubBlades
    <<< OK - ServerBlades(2) FSIOM(1) Switch(1) FCSwitch(1) SASwitch(2)
Storage(3)
>>> MMB-COM-MIB - Chassis Unit Info
    <<< FAILED
>>> RAID - Overall Status
    <<< FAILED
OK
    Name           = BX400-VIOM2
    Type           = Primergy Blade
    AdminURL       = http://xxx.xxx.xxx.xxx:80
    Sub-Blades     = ServerBlades(2) FSIOM(1) Switch(1) FSCSwitch(1)
SASwitch(2) Storage(3)

```

6.2.6 PRIMEQUEST – Test of MMB-COM-MIB.mib

After initial tests and after the Status.mib and S31.mib check, it is checked whether MMB-COM-MIB information is available.

The following information is only available for PRIMEQUEST MMB..

Sample on success:

```

>>> SNMP connect to host xxx.xxx.xxx.xxx
    <<< OK
>>> RFC1213 - Uptime
    <<< OK - UpTime = 1 day, 01:32:06.53
>>> RFC1213 - System Information
    <<< OK:

```

```

Systemname=FTS5 Description="PRIMEQUEST Management Board" Location="..."
Contact="..."
>>> Status - Subsystem Status Info
<<< FAILED
>>> S31 - Agent Info
<<< FAILED
>>> MMB-COM-MIB - Chassis Unit Info
<<< OK - PRIMEQUEST - Name="MCD3AC111" Location="..." Contact="..."
AdminUrl="http://xxx.xxx.xxx.xxx:8081" Model="PRIMEQUEST 1800E"
>>> RAID - Overall Status
<<< FAILED
OK
Name          = FTS5
Type           = Primequest
AdminURL       = http://xxx.xxx.xxx.xxx:8081
Model          = PRIMEQUEST 1800E

```

6.2.7 Any other Server – RFC1213.mib only

Sample on success – if RFC1213 can be read:

```

>>> SNMP connect to host xxx.xxx.xxx.xxx
<<< OK
>>> RFC1213 - Uptime
<<< OK - UpTime = 1 day, 11:30:57.67
>>> RFC1213 - System Information
<<< OK:
Systemname=SCOURIE Description="Hardware: x86 Family 6 Model 15 Stepping 11
AT/AT COMPATIBLE - Software: Windows Version 6.0 (Build 6002 Multiprocessor
Free)" Location="..." Contact="..."
>>> Status - Subsystem Status Info
<<< FAILED
>>> S31 - Agent Info
<<< FAILED
>>> MMB-COM-MIB - Chassis Unit Info
<<< FAILED
>>> RAID - Overall Status
<<< FAILED
OK
Name          = SCOURIE

```

6.3 IPv4 Discovery

Usable if 256 IPv4 addresses are to be checked in one run. The above described MIB test will be run in a loop. All process prints starting with "<<<" or ">>>" will be suppressed.

ATTENTION: This might need some time if there are multiple nodes where SNMP is not running or if the nodes do not permit access (for the selected SNMP community or to the local system).

6.3.1 Option

--ipv4-discovery -H<three-parts-of-an-ipv4address>
 This requires -H <address> to be "<nnn>.<nnn>.<nnn>."

6.3.2 Common Print Rules

Print the summary lines of the MIB test described above.

7 FUJITSU SNMP Trap Configuration Files

The status change of a host is only seen at the time when the Nagios scheduler calls the corresponding Nagios plug-in. Only with SNMP traps could a notification chain be started at the moment when the trap is triggered.

7.1 Standards

snmptrapd is a listener for receiving SNMP traps in LINUX. This program supports configurable follow-up actions. One of these actions could be a tool named **snmptt**.

The trap data received via **snmptrapd** includes:

- Sender IP address
ATTENTION: This is not the origin of the trap if forwarded traps are received!
- Timestamp information
- Trap SNMP OID
- One origin host name
- Trap arguments
- One origin IP address

There is no hint in this on severities or the trap OID related texts.

snmptrapd enables **the first opportunity** for notifications and for storage and filtering.

snmptt enables **the second opportunity** for notifications and for storage and filtering if triggered by **snmptrapd**.

ATTENTION: There is a particular problem around sent IP addresses and host names.

- The sender IP address is one of multiple potential IP addresses of the sender and this is NOT the IP address of the origin of traps in the case of forwarded traps! Even if origin and sender are identical, the IP addresses might differ!
- The host name is one of multiple potential host names of the originator.
- The origin IP address is one of multiple potential IP addresses of the sender – this might even be a link-local address!

Because of this the association of sent address information with a Nagios "*host*" is a problem which can only be solved by the administrators of each Nagios system.

7.2 Plugin Support

With the FUJITSU Server Plug-in, configuration files in two formats will be supported.

- One is in a separate format – easy to scan for tools written by any user and usable for user-specific **snmptrapd** handler.
- The second is the *snmptt* format, which is usable as-is for **snmptt** configurations and extendable for EXEC directives.

These configuration files contain information for associating SNMP OID regarding severities and texts, and how to insert the sent arguments!

Path in supported *tgz file*:

fujitsu/ServerViewSuite/nagios/trap

Subdirectories:

- **trapconf** (own format)
- **snmpttconf** (snmptt format)

7.3 Brief Hints around SNMPTT

URL: <http://www.snmpptt.org/>

Suggested snmptrapd.conf configuration:

```
traphandle default /usr/local/sbin/snmpptt
```

The traps can be stored in an snmpptt database which might be interpreted by already implemented Nagios plugins and other user interfaces.

At <http://www.nagios-wiki.de/nagios/howtos/snmpptt> you will find a "How to" description for snmpptt.

HINT ABOUT IP AND HOST NAME

Snmpptt expects the host name specified for Nagios to be identical with the one that *snmptrapd* can compute for the sender IP address!

ATTENTION: This does not work for forwarded traps!

8 FUJITSU CIM Server Tool: `tool_fujitsu_server_CIM.pl`

There is an additional script which might be usable for interactive – not-scheduled – calls. See following sections about functionality.

8.1 Basics

8.1.1 Script Name

Name of the script

`tool_fujitsu_server_CIM.pl`

This is a Perl script which uses `check_fujitsu_server_CIM.pl` and with this either `wbemcli` or `OpenWSMAN` perl binding.

8.1.2 Options around the Plug-in itself

`-V|--version`

Print version information and help text.

`-h|--help`

Print help text.

8.1.3 Select Usage Mode – CIM-XML or WS-MAN

`-U|--usage={C|W}`

"C" use CIM-XML protocol. This requires `wbemcli` installation. Connect to SFCB or OpenPegasus.

"W" use WS-MAN protocol. This requires `OpenWSMAN` Perl binding. Connect to `OpenWSMAN` services or `WinRM`

8.1.4 CIM Options around Addressing

`-H|--host=<name-or-ip>`

Host address as DNS name or IP address of the server.

`[-A|--admin=<adminaddress>]`

Specify administration address with which all data for the host should be retrieved. This might be the iRMC address if a corresponding CIM provider is enabled on iRMC.

HINT:

The two addresses are meant for Nagios administrators who want to define one host and Nagios services which should use the origin `$HOSTADDRESS$` and parallel to these other Nagios services which should use the iRMC address e.g. specified in `$_HOSTSV_ADMIN_OPTION$` or `$_HOSTSV_CIM_OPTIONS$`

`-P |--port=<portnumber>`

CIM service port number. The `wbemcli` internal default for WBEM https services is 5989. For WS-MAN calls the port number must be entered (e.g. 5985 or 5986) !

For WS-MAN:

For 5985 the transport type http is default.

For 5986 the transport type https is default.

`-T|--transport=<type>`

For CIM this is the transport HTTP type. Default for `wbemcli` is "https".

Other value:

- http – for simple http calls

For WS-MAN:

For 5985 the transport type http is default.

For 5986 the transport type https is default.

-u|--user=<user> -p|--password=<pwd>

Authentication credentials (Please write these options into a file – see option -I)

These options *must* be set.

ATTENTION:

1. Credentials are verified in *sfc* on managed node site and this verification is done according to *sfc* configuration. Usually credentials are checked by a library (*sfcBasicPAMAuthentication*) which defines the rules how credentials are checked. And these rules might not be as evaluated as one would expect (restrictions in use of characters or length of values etc.). Refer to *sfc* how to replace the standard library by one that meets your requirements if you suffer from some lack of functionality.

The same behaviour is true for the other CIM services OpenPegasus, OpenWSMAN and WinRM

2. Known problems:

- for *wbemcli*: Passwords must not contain any dots '.'
- Do not use characters like ' or " in the user or password string
- user and passwords which contain shell relevant signs like \$ should always be set in an option input file see option description -I below.
- Password check might vary from system to system

--cacert=<cafile>

CA certificate file.

If not set *wbemcli* parameter *-noverify* will be used resp. the "do not verify" flags are set for OpenWSMAN.

See *wbemcli* parameter *--cacert* resp *wsman* command parameter *--cacert*

1. Certificates are handled both in *wbemcli* and in *sfc* on managed node site. On managed node site this is done according to *sfc* configuration and usually certificates are checked under certain conditions by a library (*sfcCertificateAuthentication*) which defines the rules how certificates are checked. As for credential check you might want to replace this library by your own implementation. Please refer to *sfc* how to do so.

The same behaviour is true for the other CIM services OpenPegasus, OpenWSMAN and WinRM

2. Known problems:

- *wbemcli*: If *<cafile>* exists and is not empty but does not contain any certificate, on some systems *wbemcli* dumps a core.

--cert=<certfile> --privkey=<keyfile>

Client certificate file and Client private key file.

wbemcli requires both file names if this should be used.

It depends on configuration on the host side if these certificates are verified or not !

See *wbemcli* parameter *-clientcert* and *-clientkey* resp. *wsman* command and parameter

--cert and *--sslkey*

-I|--inputfile=<file> [--inputdir=<directory>]

File in which host specific options like the above named ones - with the exception of the host option - can be stored. The usage of this is *recommended* for security relevant options like *-u* and *-p*.

This file must be readable for the owner of the Plugin script used by the Nagios scheduler out of the Nagios Plugin directory !

With *inputdir* a directory path of the input option file can be specified. The directory path is ignored if the input file starts with '/'.

8.1.5 Plugin Processing Control Option

-t|--timeout=<timeout in seconds>

Timeout for the script processing.

The script sets a 60-seconds timeout for ESXi server and usage of *wbemcli* for the first connection calls. If no timeout is given via this option, a maximum timeout of 120 seconds is set.

-v|--verbose=<verbose mode level>

Enable verbose mode (levels: 1,2). Generates Multi-line output with inventory information or other additional information.

8.2 CIM Server Type Test

This should help to find information about servers with CIM to enable the assignment to NAGIOS host groups and their NAGIOS 'services'.

8.2.1 Option

--typetest

This is default for this script.

This "typetest" information test can be reduced with the following options.

--connectiontest

Test only CIM test connectivity testing CIM-XML and WS-MAN connections.

[-e or --extended]

Extended information print: Print ServerView-Agent version.

[--nopp]

This is an option for suppressing in-between process result printouts. Only a kind of summary will be printed with available system information of the host.

8.2.2 Common Print Rules

Any test starts with ">>> topic".

Any answer starts with "<<< status [additional information]".

The last lines are a kind of summary starting with a summary status and information about the server.

8.2.3 Connection Test

Here again an overview over possible connection variants:

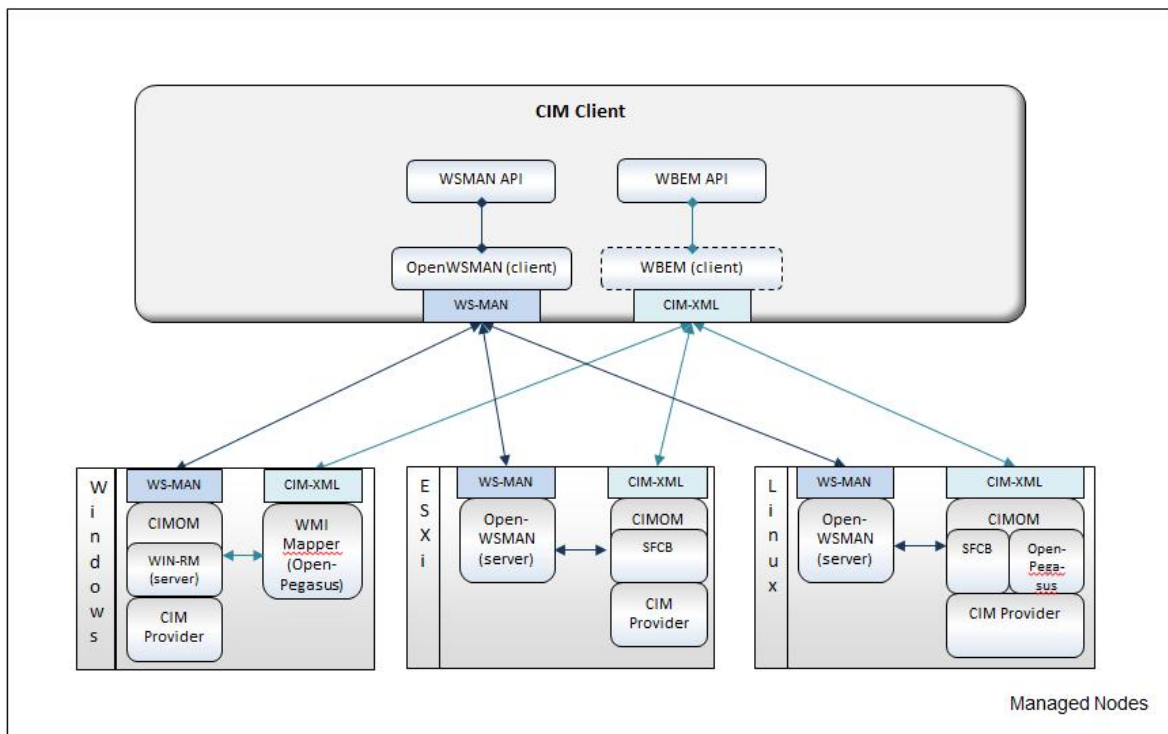


Figure 2: Protocol stacks

Dependent on specified or not specified use-mode (option -U) the connectivity via CIM-XML or WS-MAN will be checked with various port numbers and transport types (http or https).

For each connectivity check a time limit of 30 seconds is internal set. This is limited because some systems in tests seem to need a lot of time if a port is not accessible.

Following tests are done:

- CIM-XML
 - Test with specified port and transport type or test with *wbemcli* default port and transport type.
 - Only if this fails and no authentication error is detected and no specified port is set:
Test with 5989 https.
 - Only if this fails and no authentication error is detected and no specified port is set:
Test with 5988 http.
- WS-MAN
 - Test with specified port and transport type or test with 5986 https.
 - Only if this fails and no authentication error is detected and no specified port is set:
Test with 5985 http.
 - Only if this fails and no authentication error is detected and no specified port is set:
Test with 8888 https.
 - Only if this fails and no authentication error is detected and no specified port is set:
Test with 8889 http.

Sample processing information:

```
>>> connect test CIM-XML: use standard parameters
<<< CONNECTION ERROR
>>> connect test CIM-XML: 5989
<<< CONNECTION ERROR
>>> connect test CIM-XML: 5988
<<< CONNECTION ERROR
>>> connect test WS-MAN: entered parameters or port 5986
<<< CONNECTION ERROR
>>> connect test WS-MAN: port 5985
<<< OK
```

Sample result on success concerning:

```
OK
InAddress    = nnn.nnn.nnn.103
Protocol     = WS-MAN
Port         = 5985
TransType    = http
OptionFile   = AuthentQA3Win.txt
```

8.2.4 Server Type Information with ServerView Agent CIM Information

After the connection test (see above) the information of some ServerView CIM classes is fetched to get basic system type information.

The sample result data shows which data are read (if available).

If none of the information including name and the rest below is available, ServerView Agent CIM is not installed or the ServerView namespaces data can't be accessed.

Sample processing information:

```
...
>>> get basic systeminfo
<<< OK
>>> get component list
<<< OK
>>> get server update status
<<< OK
```

Sample result on success:

```
OK
InAddress    = nnn.nnn.nnn.103
Protocol     = WS-MAN
Port         = 5985
TransType    = http
OptionFile   = AuthentQA3Win.txt

Name         = RX100S7-89
Model        = PRIMERGY RX100 S7
AdminURL     = http://nnn.nnn.nnn.nnn:80
```

```

OS           = "Microsoft Windows Server 2008 R2"
Components   = Environment Power MassStorage Systemboard Network
DriverMonitor
UpdateAgent  = Status(UNKNOWN) Monitoring=available

```

8.3 IPv4 Discovery

Usable if 256 IPv4 Addresses should be checked in one run. The above described server type test will be run in a loop. All process prints starting with "<<<" or ">>>" will be suppressed.

ATTENTION: This might need some time if there are multiple nodes where it needs timeouts for the connection tests).

8.3.1 Option

--ipv4-discovery -H<three-parts-of-an-ipv4address>

This requires -H <address> to be "<nnn>.<nnn>.<nnn>."

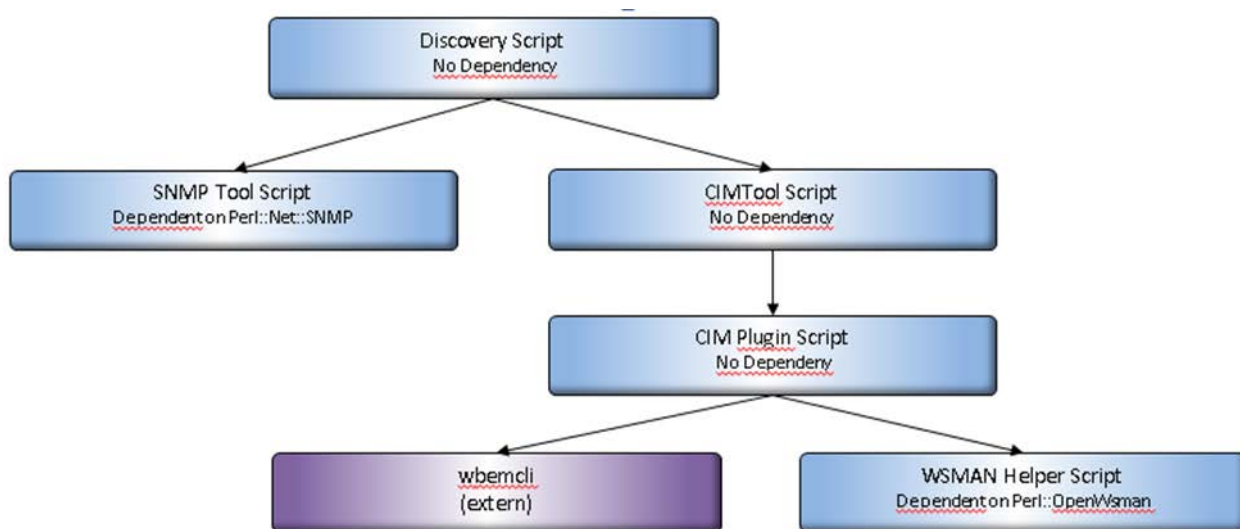
8.3.2 Common Print Rules

Print the summary lines of the tests described above.

9 Discovery – Check of Server Types and Generation of Configurations

9.1 Script-Hierarchy

Script hierarchy:



The script **discovery_fujitsu_server.pl** uses the other scripts to get the required data. To do so the other scripts must be in the same directory like the discovery script.

Scripts:

- SNMP tool script: tool_fujitsu_server.pl
- CIM tool script: tool_fujitsu_server_CIM.pl
- CIM plugin script: check_fujitsu_server_CIM.pl
- WSMAN helper script: fujitsu_server_wsman.pl

9.2 Basics

9.2.1 Script Name

Name of the script

discover_fujitsu_server_CIM.pl

9.2.2 Options around the Plugin itself

-V|--version

Print version information and help text.

-h|--help

Print help text.

9.2.3 Control Options for Called Scripts

-t|--timeout=<timeout in seconds>

Specify timeout in seconds for the call of the other scripts (SNMP or CIM checks).

ATTENTION:

This is no timeout for this script !.

[--ctimeout=<connection timeout in seconds>]

Specify timeout for the connection test to the CIM services. Default is 30 seconds. All values higher than 30 will be ignored.

ATTENTION:

This is used for the call of the CIM script !

9.2.4 Select Host Options

-H|--host=<name-or-ip>

Enter host address as DNS name or IP address of the server.

[--onehost] | --ipv4-discovery | --hc|--hostcollection=<file>

"onehost": Test only one host address specified with option -H. This is default.

"ipv4-discovery":

Test for 256 servers for a given n.n.n. IPv4 address.

Test for a limited range servers for a specified n.n.n.<firstn>-<lastn> IPv4 address.

"hostcollection": Test a list of host addresses written into a file. – In this case option -H is ignored.

Format:

Specify one address resp. one "ipv4-discovery" range in each line

9.2.5 Select Usage Mode – SNMP or CIM-XML or WS-MAN

[-U|--use={S|C|W|CW}]

Default: Try any connection abilities.

"S" use only SNMP: This requires Perl Net::SNMP and uses tool_fujitsu_server.pl

"C" use only CIM-XML protocol. This requires wbemcli installation. The script tries to connect to SFCB or OpenPegasus. This uses tool_fujitsu_server_CIM.pl and check_fujitsu_server.pl

"W" use only WS-MAN protocol. This requires OpenWSMAN Perl binding. The script tries to connect to OpenWSMAN services or WinRM. This uses tool_fujitsu_server_CIM.pl and check_fujitsu_server.pl and fujitsu_server_wsman.pl

"CW" Try CIM-CML and WS-MAN protocol.

9.2.6 Additional Authentication and Connection Options

-l|--inputfile=<filename> [-i|--inputdir=<inputfiledir>] | --ic|--inputcollection=<dir>

The input option file will be used for the tool script calls. In the files should be options for these tools - e.g. credentials, port and transport type restrictions. The discovery script tries to analyse if the content is meant for SNMP or CIM calls.

"inputfiledir" - directory path of the input option file specified with -l

"inputcollection" - directory path in which a collection of multiple input option files

9.2.7 Output Options

[-c|--config] | --txt

Store simple tool text output or store text output and generate Nagios host definition configuration. Default is --config.

[-O|--outputdir=<dir>]

Specify output directory for resulting text and configuration files. Default is "svout".

9.3 Processing Loop Overview

Here is an overview in which way the script handles multiple host and multiple option input files. This description should help to understand how this script works.

ATTENTION:

Currently the script processes all actions step-by-step (synchronic). This script can be called parallel but the amount of resulting parallel socket connections must be considered by the caller !

The first loop level is for one or more specified host addresses.

For each host address there will be a loop for one or more specified option input files sorted for protocol usage and sorted (Perl sort()) by file names.

For SNMP: The default community "public" is tested and each SNMP option input file unless a successful connection is registered.

For CIM: The option input files are tested unless a successful connection is registered and "authentication error" occur and no other failure reason is registered.

9.4 Logging Information and Results

The discovery script writes logging information to STDOUT. Additional three files for each host are stored into the output directory. One is a log file, one is a text file if the connection was successful and one is the configuration file if ServerView information can be detected.

9.4.1 Central Logging

The information starts with hint on the date and the hint in which directory the output files are stored. At the beginning the option input files are checked and after that the loops for each host and option input file protocols basic findings. At the end a summary section follows.

See following sub chapters for the description of these parts.

9.4.1.1 Checks for the Option Input Files

At the beginning it is logged for which connection protocol the option input files can be used. There will be ERROR hints if one file can't be found or read and there will be a WARNING hint if a file can't be assigned to SNMP or CIM usage.

If the "inputcollection" directory is specified only flat files inside will be checked !

ATTENTION:

Unusable or not existing files will be ignored !

Example for input collection option usage:

```
>>> read input option file AUTHQA3/A_Admin.txt
<<< OK - usable for CIM calls
>>> read input option file AUTHQA3/A_LX_VAR.txt
<<< OK - usable for CIM calls
>>> read input option file AUTHQA3/A_WIN_VAR.txt
<<< OK - usable for CIM calls
>>> read input option file AUTHQA3/AuthentQA3.txt
<<< OK - usable for CIM calls
>>> read input option file AUTHQA3/AuthentQA3Win.txt
<<< OK - usable for CIM calls
>>> read input option file AUTHQA3/AuthentSNMP.txt
<<< OK - usable for SNMP calls
```

9.4.1.2 Central Process Logging for Each Host

The middle part of the central logging is the process log for each host and option input files.

For SNMP each option input file will be checked unless a successful connection can be registered. For CIM calls multiple option input files are checked only if an authentication error can be discovered.

Example of inaccessible host:

```
>>> nnn.nnn.nnn.0
... try Community public
... try AUTHQA3/AuthentSNMP.txt
... try AUTHQA3/A_Admin.txt
<<< nnn.nnn.nnn.0 3
```

Legend:

The number after "<<< address" shows 0 for connection OK, 3 for connection UNKNOWN

Example of SNMP connectable host:

```
>>> nnn.nnn.nnn.6
... try Community public
... SNMP connection OK
... no ServerView SNMP Agent information
... try AUTHQA3/A_LX_VAR.txt
<<< nnn.nnn.nnn.6 0
```

Example of detected CIM authentication errors:

```
>>> nnn.nnn.nnn.54
... try Community public
... SNMP connection OK
... SNMP ServerView information OK
... try AUTHQA3/A_admin.txt
... CIM AUTHENTICATION ERROR
... try AUTHQA3/A_LX_VAR.txt
... CIM AUTHENTICATION ERROR
...
<<< nnn.nnn.nnn.54 0
```

9.4.1.3 Summary Information

In the summary part are the result collections for "timeout", "Authentication errors", "Connection is OK but no ServerView information is found" and detected host names.

For each detected hostname a collection of found address hints is printed.

"address hints" are <address>|{SNMP,CIM}|[{SV,SViRMC}]

- The first part is the tested address
- The second part is a hint on the connection protocol
- The third part is a hint if ServerView Agent "SV" or ServerView iRMC information "SViRMC" can be detected

Any host which is definitely not reachable is omitted in the summary.

Example extract:

```
SUMMARY:
TIMEOUT
    nnn.nnn.nnn.4|CIM + nnn.nnn.nnn.11|CIM + ...
AUTHENTICATION ERROR
    nnn.nnn.nnn.10|CIM + ...
CONNECTION OK - NO SERVERVIEW
    nnn.nnn.nnn.6|CIM + nnn.nnn.nnn.28|CIM
CLIENT_TX120
    nnn.nnn.nnn.49|SNMP|SV
...
MX130-S22
    nnn.nnn.nnn.99|SNMP|SV
N.A.
    nnn.nnn.nnn.143|CIM|SViRMC + nnn.nnn.nnn.151|CIM|SViRMC
RX100S42
    nnn.nnn.nnn.26|SNMP|SV
...
RX200-S82
    nnn.nnn.nnn.144|SNMP|SV + nnn.nnn.nnn.145|CIM|SViRMC
...
```

9.4.2 Logging for Each Host Discovery

File name: <outputdir>/<address>.log

HINT:

This file is opened in the append mode.

Each test log sequence is embraced with current date and time hints. Inside are hints on tested protocol, used script call and its output data (independent on success or not) and the resulting connection status.

Example svout/nnn.nnn.nnn.10.log

```
START DATE:      Wed Apr 23 13:48:51 2014
ADDRESS:         nnn.nnn.nnn.10
>>> SNMP
... call: ./tool_fujitsu_server.pl -H nnn.nnn.nnn.10 --typetest --nopp -e
OK
    Protocol      = SNMP
    Name          = RX200S52
    Components    = Environment PowerSupply MassStorage Systemboard Network DrvMonitor
    OS            = Windows Server 2008 Standard Service Pack 2
    FQDN          = RX200S52
    Type          = PRIMERGY with SV SNMP Agent
    Model         = PRIMERGY RX200 S5
    AdminURL      = http://nnn.nnn.nnn.11:80
    UpdateAgent   = Status(UNKNOWN) SNMP-Monitoring=available
<<< OK
END DATE:        Wed Apr 23 13:48:53 2014
START DATE:      Wed Apr 23 13:48:53 2014
```

```

ADDRESS:      nnn.nnn.nnn.10
INFILE:       AUTHQA3/A_Admin.txt
>>> CIM
... call: ./tool_fujitsu_server_CIM.pl -H nnn.nnn.nnn.10 -I AUTHQA3/A_Admin.txt --
typetest -nopp -e
UNKNOWN - AUTHENTICATION FAILED
      InAddress   = nnn.nnn.nnn.10

<<< AUTHENTICATION ERROR
END DATE:      Wed Apr 23 13:48:55 2014
...

```

9.4.3 Text Information for Each Host

File name: <outputdir>/<address>_<protocol>.txt

This file will be only created if the connection test is successful. The file contains the output information of the tool_fujitsu_server.pl resp. the tool_fujitsu_server_CIM.pl.

Example svout/nnn.nnn.nnn.10_SNMP.txt:

```

OK
Protocol      = SNMP
Name          = RX200S52
Components    = Environment PowerSupply MassStorage Systemboard Network DrvMonitor
OS            = Windows Server 2008 Standard Service Pack 2
FQDN          = RX200S52
Type          = PRIMERGY with SV SNMP Agent
Model         = PRIMERGY RX200 S5
AdminURL      = http://nnn.nnn.nnn.11:80
UpdateAgent   = Status(UNKNOWN) SNMP-Monitoring=available

```

Example svout/nnn.nnn.nnn.nnn_CIM.txt

```

OK
InAddress     = nnn.nnn.nnn.103
Protocol      = WS-MAN
Port          = 5985
TransType     = http
ServiceType   = Windows
OptionFile    = AUTHQA3/AuthentQA3Win.txt

Name          = RX100S7-89
Model         = PRIMERGY RX100 S7
AdminURL      = http://nnn.nnn.nnn.104:80
OS            = "Windows Server 2008 R2 Datacenter (x64)"
Components    = Environment Power MassStorage Systemboard Network DriverMonitor
UpdateAgent   = Status(UNKNOWN) Monitoring=available

```

9.4.4 Nagios Configuration Files for Each Host

File name: <outputdir>/<address>_<protocol>_<hostname>.cfg

These files are only written if ServerView information are available (and readable). The format is meant for Nagios.

Example svout/nnn.nnn.nnn.10_SNMP_RX200S52.cfg

```

define host {
    host_name      nnn.nnn.nnn.10_RX200S52_SNMP
    display_name   RX200S52
    address        nnn.nnn.nnn.10
    hostgroups     primergy-servers,primergy-update-monitor
    use            windows-server,fj_server_icon
    notes_url      http://nnn.nnn.nnn.11:80
    register       1
}

```

```
Example svout/nnn.nnn.nnn.103_CIM_RX100S789.cfg
define host {
    host_name      nnn.nnn.nnn.103_RX100S789_CIM
    display_name   RX100S789
    address        nnn.nnn.nnn.103
    hostgroups     primergy-servers-CIM
    use            linux-server,fj_server_icon
    _SV_CIM_OPTIONS -UW -P5985 -Thttp -SW -I AUTHQA3/AuthentQA3Win.txt
    notes_url      http://nnn.nnn.nnn.104:80
    register       1
}
```

HINTS:

This configuration file can be used as-is but the generated Nagios host name "host_name" inside is not a "beautiful" one.

The Nagios host name is created with address and host name and protocol because it might happen that one host has multiple addresses or in worst case that multiple hosts have the same internal host name. It may also happen that one host is reachable by SNMP and CIM protocol. In these case there will be multiple configuration files for one server !

Check information in the summary information in the central log file and check the files in the output directory.