NEC

SX-Aurora TSUBASA
SX-Aurora TSUBASA
Operation and Management Guide
(with OSS)

ChangeLog

Edition	Date of Issue	Chapter/Section	New/ Change	Changes
1	March 2018	All	New	
2	March 2018	2.1.1. Preparation of the software packages to install for the SX-Aurora TSUBASA system	Change	Download URL was changed.
		2.2.1. Installing Ansible	Change	The reference URL was deleted.
		2.4.1. Files for Ansible	Change	vh-set-bmc.yml was deleted.
		2.4.2. Tuning Ansible performance	Change	The reference URL was deleted.
		2.5. Execution of Commands on VHs	Change	The reference URL was deleted.
		2.6. File Manipulation on VHs	Change	The reference URL was deleted.
		3.1. Zabbix	Change	The reference URL and the screen images of Zabbix were deleted.
		3.2. Ganglia+Nagios	Change	The reference URL and the screen images of Ganglia were deleted.
		3.1.1. Preparation 3.2.1. Preparation	Change	Download URL was changed.
		4.1. Controlling power with BMC	Change	The explanation of the BMC operation user was added. 4.1.1.1. Checking BMC installation and 4.1.1.2. BIOS Configuration on VHs were deleted.
		C.4.1. VH power control using the powerctrl commands fails.	Change	Cause and Solution were changed.
3	May 2018	2.1.1. Preparation of the software packages to install for the SX-Aurora TSUBASA system	Change	NEC MPI was excluded from the update target which Ansible is used. NEC Parallel Debugger was added.
		2.2.5. Creating a Yum repository for SX-Aurora TSUBASA	Change	"4. Update the group definition file." was deleted.
		2.4.1. Files for Ansible	Change	roles/nec-mpi was deleted. roles/parallel-debugger was added.

Edition	Date of Issue	Chapter/Section	New/ Change	Changes
		2.6. File Manipulation on VHs 2.7. Software Installation on VHs	Change	The master playbook was changed vh.yml to vh-set.yml.
		2.10.1. Editing Playbook	Change	nec-mpi was deleted. parallel-debugger was added.
		3.1.2.1. Preparation	Change	Add the operation example of firewalld service.
		3.1.3.1. Manual Setup		
		3.2.2.1. Preparation on the Management Server	Change	Fix the operation example of firewalld service.
		3.2.3.1. Manual Setup		
		A.2. The Item Keys Provided by the Loadable Modules	Change	Fix a misspelling.
		C.2. Operational Status Monitoring(Zabbix)	Change	Fix a misspelling.
4	May 2018	2.1.1. Preparation of the software packages to install for the SX-Aurora TSUBASA system	Change	NEC MPI was re-supported as the update target with Ansible.
		2.4.1. Files for Ansible	Change	roles/nec-mpi was re-supported.
		2.10.1. Editing Playbook	Change	nec-mpi was re-supported.
5	Jan 2019	1.1. Scope	Change	"System Configuration Management" functions by Ansible suspend support.

Trademarks

- Linux is a registered trademark of Linus Torvalds in the United States and other countries.
- Red Hat and Red Hat Enterprise Linux are registered trademarks of Red Hat, Inc. in the United States and other countries.
- Apache is a registered trademark of Apache Software Foundation.
- InfiniBand is a trademark or service mark of InfiniBand Trade Association.
- Ansible is a registered trademark of Red Hat, Inc. in the United States and other countries.
- Python is a registered trademark of the Python Software Foundation.
- Nagios is a registered trademark of Nagios Enterprises, LLC.
- Zabbix is a trademark of Zabbix LLC that is based in Republic of Latvia.
- All other product, brand, or trade names used in this publication are the trademarks or registered trademarks of their respective trademark owners.

Copyright

No part of this document may be reproduced, in any form or by any means, without permission from NEC Corporation.

The information in this document is subject to change at any time, without notice.

Contents

ChangeLog **2**
Trademarks **3**
Copyright **4**

List of Figures **8**

List of Tables **9**

Chapter 1 Overview **10**
1.1. Scope 10
1.2. Glossary 10
1.3. Configuration 11
1.4. Operating Environment 12

Chapter 2 System Configuration Management **13**
2.1. Preparation 13
2.1.1. Preparation of the software packages to install for the SX-Aurora TSUBASA system 13
2.1.2. Preparing of the template package 16
2.2. Initial Setup of the Management Server 16
2.2.1. Installing Ansible 17
2.2.2. Installing Python 17
2.2.3. Installing Apache HTTP Server 17
2.2.4. Installing createrepo 17
2.2.5. Creating a Yum repository for SX-Aurora TSUBASA 17
2.2.6. Creating an administrative user 18
2.2.7. Installing the template package 18
2.2.8. Setting administrative user's SSH public and private keys 19
2.2.9. Creating the VH information file 19
2.2.10. Creating host list file 20
2.3. Initial Setup of VHs 20
2.4. Preparing for using Ansible 22
2.4.1. Files for Ansible 22
2.4.2. Tuning Ansible performance 24
2.4.3. Starting ssh-agent 25
2.4.4. Registering VHs 25
2.4.5. Checking VH connectivity 25
2.4.6. Starting Apache HTTP Server 25
2.5. Execution of Commands on VHs 26
2.6. File Manipulation on VHs 26
2.7. Software Installation on VHs 27
2.8. Software Uninstallation from VHs 28
2.9. Update of Other Software 29
2.10. Software Update on VHs 29
2.10.1. Editing Playbook 29
2.10.2. Execution of the Playbook 31

Chapter 3 Operational Status Monitoring.....33
3.1. Zabbix.....33
3.1.1. Preparation.....34
3.1.2. Initial Setup of Zabbix Server.....35
3.1.2.1. Preparation.....35
3.1.2.2. Installation of Zabbix Server.....37
3.1.2.3. Start of Zabbix Server.....38
3.1.2.4. Initial Setup of Zabbix Server.....38
3.1.3. Initial Setup of Zabbix Agent.....38
3.1.3.1. Manual Setup.....39
3.1.3.2. Procedure with Ansible.....41
3.1.4. Configuration of Host Information.....43
3.1.4.1. Registration of Hosts.....45
3.1.5. Configuration of Monitoring Items.....45
3.1.5.1. Configuration of Monitoring Items Using Templates.....46
3.1.5.2. Addition of Monitoring Items.....46
3.1.5.3. Information Gathering with Loadable Modules.....46
3.1.5.4. Information Gathering with User Parameters.....47
3.1.5.5. Monitoring of VEOS Services.....47
3.1.5.6. Performance Tuning.....48
3.1.6. Configuration of Triggers.....48
3.1.7. Configuration of Actions.....49
3.1.8. Customization of the Web Interface.....49
3.1.9. Creation of Loadable Modules from Source Files.....49
3.1.10. Exclusion of monitoring hosts for updating SX-Aurora TSUBASA software.....50
3.1.11. Inclusion of monitoring hosts updating SX-Aurora TSUBASA software.....50
3.2. Ganglia and Nagios.....50
3.2.1. Preparation.....51
3.2.2. Configuration of the Management Sever.....52
3.2.2.1. Preparation on the Management Server.....52
3.2.2.2. Installation of Ganglia Server.....53
3.2.2.3. Configuration of Ganglia Server.....53
3.2.2.4. Installation of Nagios.....54
3.2.2.5. Configuration of Nagios.....54
3.2.2.6. Configuration of Actions.....55
3.2.2.7. Start of Ganglia Server and Nagios.....55
3.2.3. Configuration of Ganglia Agent.....56
3.2.3.1. Manual Setup.....56
3.2.3.2. Setup Using Ansible.....59
3.2.3.3. Exclude the VH from monitoring.....62
3.2.3.4. Restart monitoring of the VH.....62

Chapter 4 Power Control.....64
4.1. Controlling power with BMC.....64
4.1.1. Configuration for controlling power with BMC.....64
4.1.1.1. Installing IPMITool on the management server.....64
4.1.1.2. Configuration for Power control tool powerctrl.....64
4.1.2. Powering on/off VHs with BMC.....64
4.2. Powering on/off VHs with WOL.....65
4.2.1. Configuration of controlling power with WOL.....65
4.2.1.1. Check the network card.....65

4.2.1.2. Configuration for the power control tool powerctrl_w..... 65
4.2.2. Powering on/off VHs with WOL..... 65

Chapter 5 Reference..... 67
5.1. create-hostlist.py..... 67
5.2. powerctrl..... 68
5.3. powerctrl_w..... 70
5.4. setup-hostlist..... 72
5.5. vh_host.conf..... 72
5.6. vh_host_w.conf..... 73

Appendix A Monitoring Items of Zabbix..... 75
A.1. The Item Keys Provided by the Loadable Modules..... 75
A.2. The Items and Triggers Provided by the Templates..... 82

Appendix B Monitoring Items of Ganglia..... 88
B.1. The List of the Monitoring Items (Ganglia)..... 88

Appendix C Trouble Shooting..... 92
C.1. Operation Management..... 92
C.1.1. Entry of the SSH private key's password is required..... 92
C.1.2. Playbook execution fails with "sudo: sorry, you must have a tty to run sudo".... 92
C.1.3. Package download fails with "urlopen error [Errno 113] No route to host"..... 93
C.1.4. Yum repository update fails with "[Errno 14] HTTP Error 502 - Bad Gateway"..... 94
C.2. Operational Status Monitoring(Zabbix)..... 95
C.2.1. Cannot get VE information, and 'Not supported by Zabbix Agent' is shown to item information of monitoring host..... 95
C.2.2. Cannot get VE sensor information, and 'File can not access' is shown to item information of monitoring host..... 95
C.2.3. Cannot get a core temperature of VE, and 'Specified core is not available' is shown to item information of monitoring host..... 95
C.3. Operational State Monitoring (Ganglia+Nagios)..... 96
C.3.1. Metrics for the SX-Aurora TSUBASA system are not displayed on the Ganglia web interface..... 96
C.3.2. The values in a graph for the SX-Aurora TSUBASA system are displayed as “0” on the Ganglia web interface..... 96
C.3.3. The status of VH services is displayed as "UNKNOWN" on the Nagios web interface..... 96
C.4. Power Control..... 97
C.4.1. VH power control using the powerctrl commands fails..... 97
C.4.2. VH power control using the powerctrl_w commands fails..... 97

Appendix D Ansible Playbook..... 99
D.1. How to check Ansible playbooks..... 99
Bibliography..... 100

List of Figures

Figure 1: The configuration of the system configuration management..... 11

Figure 2: The management network and BMC network are identical..... 11

Figure 3: The management network and BMC network are different..... 12

Figure 4: The Configuration of Zabbix..... 33

Figure 5: VEs are registered as hosts..... 43

Figure 6: VEs are not Registered as Hosts..... 44

Figure 7: Example 1: Configuration of Host Groups..... 44

Figure 8: Example 2: Configuration of Host Groups..... 45

Figure 9: Configuration of Ganglia and Nagios..... 51

List of Tables

Table 1: Items in the VH information file..... 19
Table 2: Structure of Sample Playbooks..... 22
Table 3: Structure of <roles-dir>..... 23
Table 4: Settings of ansible.cfg..... 24
Table 5: List of sample files Installation Path..... 34
Table 6: List of sample files Installation Path..... 51
Table 7: The Item Keys Provided by ve_hw_item.so..... 75
Table 8: The List of Keys Provided by ve_os_item.so..... 81
Table 9: The List of the Items of the Application Name(VEHW)..... 82
Table 10: The List of the Application Name (VEOS)..... 86
Table 11: The List of the Items of the Application Name (VEOS-SERVICE)..... 87
Table 12: The List of the Metrics Provided by ve.py..... 88
Table 13: The List of the Keys Provided by veos.py..... 90

Chapter 1 Overview

1.1. Scope

This document explains how to manage system configuration, monitor operational status, and control power supply using open source software (OSS) in a large-scale SX-Aurora TSUBASA system.

- Operational Status Monitoring

  The SX-Aurora TSUBASA system recommends the following OSS for this purpose, and explains the usage of them.

  - Zabbix
  - Ganglia+Nagios

  In addition to the monitoring, system failure and shortage of resources can be detected and informed to the administrator with the OSS above. Please select Zabbix or the combination of Ganglia and Nagios according to your system requirement.
- System Configuration Management

  How to execute commands, edit files, start or stop services, and install, uninstall, or update packages on VHs from the management server.
- Power Supply Control

  How to turn on or off the power of VHs.

Notice

The "System Configuration Management" function by Ansible suspend support at Jan 2019.

This document includes the following contents and the description related to Ansible, but you cannot execute as the document. Please use only a reference.

- " *Chapter 2 System Configuration Management* on page 13 "
- " *3.1.3.2. Procedure with Ansible* on page 41 "
- " *3.2.3.2. Setup Using Ansible* on page 59 "

1.2. Glossary

The table below lists terms used in this document.

Terms	Description
VE	Engine based on NEC's vector architecture and developed to run vector programs
VH	Linux server host mounting VE

1.3. Configuration

The configuration for operational status monitoring of the SX-Aurora TSUBASA system depends on the OSS you choose. Please refer to Chapter 3. " *3.1. Zabbix* on page 33 " " *3.2. Ganglia and Nagios* on page 50 "

The system configuration management of VHs requires Ansible, which is OSS. Therefore, the management server where Ansible is running and all VHs in a system must be connected by a network.

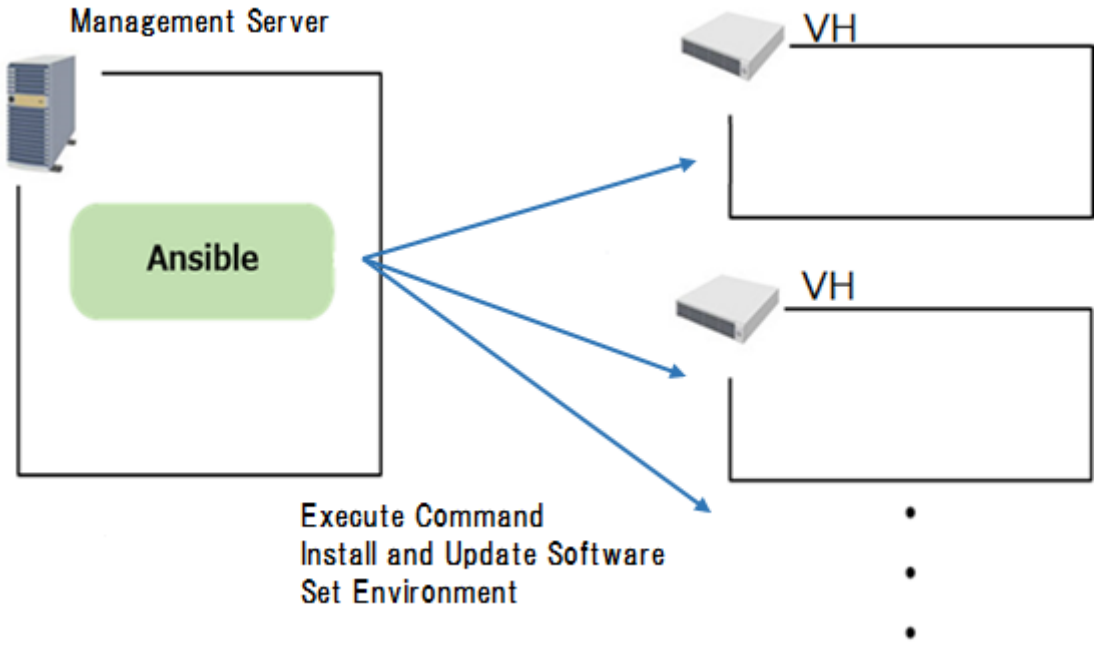

Figure 1: The configuration of the system configuration management

To turn on and off the power of VHs from the management server, the server needs to have access to Baseboard Management Controllers (BMC) on the VHs. Examples of the configurations are shown in Figures 2 and 3.

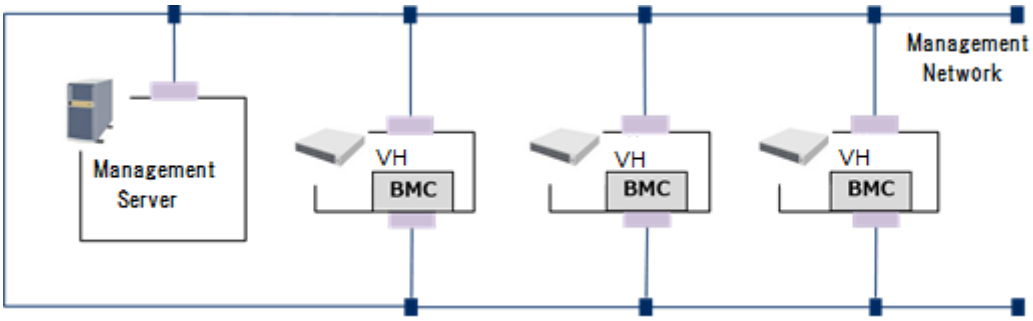

Figure 2: The management network and BMC network are identical.

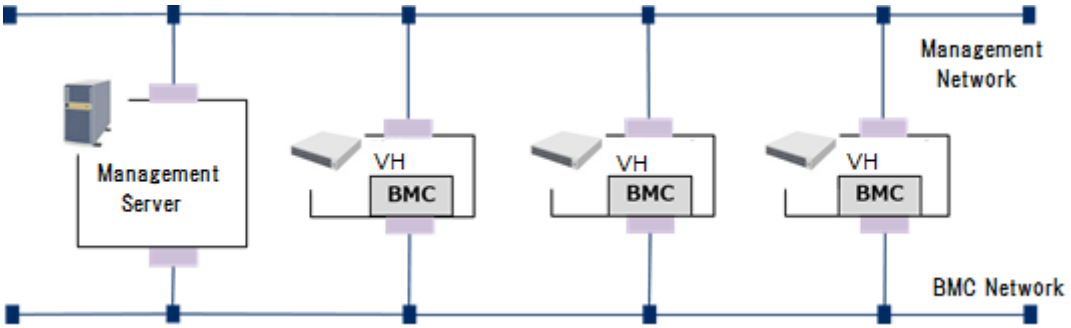

Figure 3: The management network and BMC network are different.

1.4. Operating Environment

The following environment supports the system management of VHs described in this document.

[Management Server]

H/W	x86_64 Architecture machine
OS	Red Hat Enterprise Linux 7
OSS	Zabbix 3.0.11
	Ganglia 3.7.2
	Nagios 4.3.2
	Ansible 2.3.0.0
	IPMItool 1.8.15

[VH]

H/W	The models listed in the SX-Aurora TSUBASA product catalog.
OS	Please refer to *SX-Aurora TSUBASA Installation Guide*.
OSS	Zabbix 3.0.11
	Ganglia 3.7.2
	Nagios 4.3.2

Chapter 2
System Configuration Management

Ansible enables the configuration management of the SX-Aurora TSUBASA environment on VHs. The steps for setting up Ansible are as follows:

1. Preparation
2. Initial Setup of the Management Server

   Set up the environment for using Ansible on the management server.
3. Initial Setup of VHs

   Set up the environment in which you can manage the system configuration of VHs from the management server.
4. Initial Setup of Ansible

These steps enable execution of commands and updates of software on VHs from the management server as the management user.

Please perform the operation in " *2.4. Preparing for using Ansible* on page 22 " and after in this chapter as the management user on the management server.

2.1. Preparation

This chapter describes files to be prepared before starting tasks described in this document.

2.1.1. Preparation of the software packages to install for the SX-Aurora TSUBASA system

When updating the SX-Aurora TSUBASA system software installed on VHs, prepare the software packages of the SX-Aurora TSUBASA system and the yum repository's group definition file (TSUBASA-groups.xml). These files will be used in " *2.2.5. Creating a Yum repository for SX-Aurora TSUBASA* on page 17 "" *2.10. Software Update on VHs* on page 29 ".

Some of the packages are released separately for RHEL versions. So download and prepare the packages for using system.

This document does not cover updating of C/C++ compiler and Fortran compiler. For updating the software, please refer to *SX-Aurora TSUBASA Installation Guide*.

- List of the Package files

  The following packages are included in the SX-Aurora TSUBASA system software.

Products	Package files	Free/ Non-free
License access library	aurlic-lib.x86_64	Free
VEOS Application Runtime	coreutils-ve.x86_64	Free
	gdb-ve.x86_64	
	libsysve-musl.x86_64	
	libved.x86_64	

Products	Package files	Free/ Non-free
	musl-libc-ve.x86_64	
	procps-ng-ve.x86_64	
	psacct-ve.x86_64	
	psmisc-ve.x86_64	
	strace-ve.x86_64	
	sysstat-ve.x86_64	
	time-ve.x86_64	
	util-linux-ve.x86_64	
	ve-memory-mapping.x86_64	
	ve_drv-kmod.x86_64	
	velayout.x86_64	
	veos.x86_64	
	veos-libveptrace.x86_64	
	veosinfo.x86_64	
	vesysinit.noarch	
	vesysinit-udev.noarch	
	vp-kmod.x86_64	
VEOS Application Development	autoconf-ve.noarch	Free
	automake-ve.noarch	
	gdb-ve.x86_64	
	libsysve-musl.x86_64	
	libsysve-musl-devel.x86_64	
	libtool-ve.x86_64	
	libved.x86_64	
	musl-libc-ve.x86_64	
	musl-libc-ve-devel.x86_64	
	vedebuginfo.noarch	
	velayout.x86_64	
	veos-libveptrace.x86_64	
	veos-musl-headers.x86_64	
InfiniBand for SX-Aurora TSUBASA	libibverbs-ve-musl.x86_64	Free
	libvedma-ve-musl.x86_64	
	libmlx5-ve-musl.x86_64	
	libveib.x86_64	

Products	Package files	Free/ Non-free
	ve_peermem.x86_64	
	ve_peermem.src	
MMM	ftmon.x86_64	Free
	libsignature.x86_64	
	mmm.x86_64	
	mmm-analysis.x86_64	
	mmm-msl.x86_64	
	rtmon.x86_64	
	ve-firmware.noarch	
	ve-power.x86_64	
VMC Firmware	vmcfw.noarch	Free
ScaTeFS Client	[RHEL 7.3/CentOS 7.3]	Non-free
	scatefs-client-libscatefsib.x86_64	
	scatefs-client-libscatefsib_ve.x86_64	
	scatefs-client-modules-mlnx_ofed.x86_64	
	scatefs-client-mount-utils.x86_64	
	scatefs-client-rcli-utils.x86_64	
	scatefs-client-utils.x86_64	
	[RHEL 7.4 or later/CentOS 7.4 or later]	
	kmod-scatefs-client-modules-mlnx_ofed.x86_64	
	scatefs-client-libscatefsib.x86_64	
	scatefs-client-libscatefsib_ve.x86_64	
	scatefs-client-mount-utils.x86_64	
	scatefs-client-rcli-utils.x86_64	
	scatefs-client-utils.x86_64	
NEC MPI	nec-mpi-devel-1-0-0.x86_64	Non-free
	nec-mpi-libs-1-0-0.x86_64	
	nec-mpi-utils-1-0-0.x86_64	
	nec-mpi-runtime.x86_64	
Tuning Tool	nec-veperf.x86_64	Non-free
	nec-ftraceviewer.x86_64	
NEC Parallel Debugger	nec-paralleldebugger.x86_64	Non-free
NQSV/JobServer	NQSV-JobServer.x86_64	Non-free

Products	Package files	Free/ Non-free
NQSV/Client	NQSV-Client.x86_64	Non-free
Numeric Library Collection	nec-asl-ve-1.0.0.x86_64	Non-free
	nec-aslfftw-ve-1.0.0.x86_64	
	nec-blas-ve-1.0.0.x86_64	
	nec-heterosolver-ve-1.0.0.x86_64	
	nec-lapack-ve-1.0.0.x86_64	
	nec-nlc-base-1.0.0.noarch	
	nec-nlc-doc-1.0.0.noarch	
	nec-sblas-ve-1.0.0.x86_64	
	nec-scalapack-ve-1.0.0.x86_64	
binutils	binutils-ve.x86_64	Non-free
C/C++ compiler	nec-nc++.x86_64	Non-free
	nec-nc++-musl-inst.noarch	
	nec-nc++-doc.noarch	
Fortran compiler	nec-nfort.x86_64	Non-free
	nec-nfort-musl-inst.noarch	
	nec-nfort-doc.noarch	

These files can be got from the following places.

Files	Places
TSUBASA-groups.xml	" *https://jpn.nec.com/hpc/aurora/ve-software/* " or " *https://www.nec.com/en/global/prod/hpc/aurora/ve-software/* "
Update package files	-

There are difference in the contents of the group definition file (TSUBASA-groups.xml) for each OS version. Please download the file of the corresponding version.

2.1.2. Preparing of the template package

The template package, TSUBASA-sysmng-soft-X.X- Y.noarch.rpm, provides sample files and tools described in the chapter of Configuration Management. Download the latest TSUBASA-sysmng-soft-X.X- Y.noarch.rpm and save it in any directory you wish on the management server.

2.2. Initial Setup of the Management Server

This section explains how to set up the environment for using Ansible on the management server.

Caution
If you have already installed using *SX-Aurora TSUBASA Installation Guide (with OSS)*, skip this section.

2.2.1. Installing Ansible

Install Ansible on the management server. For information about verified versions, see " *1.4. Operating Environment* on page 12 ". Please refer to the official website of Ansible for the installation sequence.

2.2.2. Installing Python

The conversion tool of the VH information file, " *5.1. create-hostlist.py* on page 67 " uses python. Check Python 2.7 or later of the Python 2.x series is installed.

The xlrd package of the python-excel library is used. Install and run the `pip` command to install the xlrd package.

Example:

```
# yum install python-pip
# pip install xlrd
```

2.2.3. Installing Apache HTTP Server

As described in " *2.10. Software Update on VHs* on page 29 ", " *3.1.3.2. Procedure with Ansible* on page 41 " and " *3.2.3.2. Setup Using Ansible* on page 59 ", you use the HTTP server on the management server to download files to each VH. Install the Apache HTTP Server on the management server and start the HTTP server.

```
# yum -y install httpd
# systemctl start httpd.service
```

2.2.4. Installing createrepo

As described in " *2.10. Software Update on VHs* on page 29 ", " *3.1.3.2. Procedure with Ansible* on page 41 " and " *3.2.3.2. Setup Using Ansible* on page 59 ", the SX-Aurora TSUBASA system software programs are updated and the package of monitoring software is installed on VHs from the Yum repository for SX-Aurora TSUBASA on the management server. Please install the tool to create Yum repository, 'createrepo' on the management server.

```
# yum -y install createrepo
```

2.2.5. Creating a Yum repository for SX-Aurora TSUBASA

SX-Aurora TSUBASA system software programs are updated and the package for monitoring system is installed from the SX-Aurora TSUBASA Yum repository as described in " *2.10. Software Update on VHs* on page 29 ", " *3.1.3.2. Procedure with Ansible* on page 41 " and " *3.2.3.2. Setup Using Ansible* on page 59 ".

Please create the SX-Aurora TSUBASA Yum repository on the management server by the following procedure.

1. Create a directory for the repository.

   Create a directory for the repository in the DocumentRoot directory of the HTTP server.

   ```
   # mkdir -p /var/www/html/repos/TSUBASA/rpms
   ```

2. Store the SX-Aurora TSUBASA system software packages.

   Place the SX-Aurora TSUBASA system software packages you have prepared as instructed in " *2.1.1. Preparation of the software packages to install for the SX-Aurora TSUBASA system* on page 13 " in the directory created in the previous step.

   ```
   # ls -1 /var/www/html/repos/TSUBASA/rpms/
   libsignature-0.3.6-1.x86_64.rpm
   mmm-0.3.6-1.x86_64.rpm
   ```

```
 :
```

3. Store the group definition file.

   Place the group definition file of the SX-Aurora TSUBASA Yum repository (TSUBASA-groups.xml) you have prepared as instructed in " *2.1.1. Preparation of the software packages to install for the SX-Aurora TSUBASA system* on page 13 " in `[ /var/www/html/repos/TSUBASA ]`.

```
# ls /var/www/html/repos/TSUBASA
TSUBASA-groups.xml  rpms
```

4. Create a Yum repository.

   Run the createrepo command to create a Yum repository.

```
# createrepo -v -g TSUBASA-groups.xml /var/www/html/repos/TSUBASA
```

   This command creates 'repodata' directory under `[ /var/www/html/repos/TSUBASA ]` and also creates repomd.xml' file and database files under the 'repodata' directory.

Caution
Do not grant the write permission on the directories and files created in /var/www/html through the steps above to group and others. Set 755 for the directory permission and 644 for the file permission.

Caution
If you add or replace any package file, run the createrepo command to update the repository.

2.2.6. Creating an administrative user

Create an account for an administrative user admin. And it is required to give sudo privilege and to disable requiretty for admin user. admin is a user who runs Ansible commands.

Example

```
# useradd -m admin
# passwd admin
 Changing password for user admin.
 New UNIX password: (Input password)
Retype new UNIX password: (re-input password)
passwd: all authentication tokens updated successfully.
# visudo
     :
admin ALL=(ALL) NOPASSWD: ALL
Defaults:admin !requiretty
```

When you are using Ansible, an administrative user admin remotely logs in VHs from the management server over SSH. Because of this, you need to create the administrative user admin on the VHs as well. Create the administrative user for VHs when you install the OS on VHs by Cobbler. How to create the administrative user for VHs is described in " *2.3. Initial Setup of VHs* on page 20 ".

2.2.7. Installing the template package

Install the template package for the sample files and tools described in this document on the management server.

The template package is provided as TSUBASA-sysmng-soft-*X.X-Y*.noarch.rpm. Get the latest version of the TSUBASA-sysmng-soft package and install it on the management server.

```
# rpm -i TSUBASA-sysmng-soft-X.X-Y.noarch.rpm
```

The sample files and tools are installed under `[ /opt/nec/sysmng-soft/ ]`.

2.2.8. Setting administrative user's SSH public and private keys

Ansible used to setup the VH environment first logs into VHs over SSH for further operations. Generate public and private keys for the administrative user admin to log in VHs over SSH, and change the SSH configuration.

- Creating a public/private key pair

```
# su - admin
$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/admin/.ssh/id_rsa):
Enter passphrase (empty for no passphrase): (Inout passphrase)
Enter same passphrase again: (Re-input passphrase)
Your identification has been saved in /home/admin/.ssh/id_rsa.
Your public key has been saved in /home/admin/.ssh/id_rsa.pub.
The key fingerprint is:
      :
```

The public key is created in the ~admin/.ssh/id_rsa.pub file. This public key file is distributed to the VH at the time of creating administrative user on VH. Details are described in " *2.3. Initial Setup of VHs* on page 20 ".

- Changing the SSH configuration

Change the SSH configuration of administrative user admin, `[ ~admin/.ssh/config ]`, as follows to prevent the warning message from appearing when connecting to a VH over SSH.

```
$ cat ~/.ssh/config
Host *
  StrictHostKeyChecking  no
$ chmod 600 ~/.ssh/config
```

2.2.9. Creating the VH information file

You need information such as host names, IP addresses for BMC and MAC addresses of VHs to setup VHs environment and control power of VHs by Ansible. So create a VH information file containing information about VHs you use.

For the information file, use the `[ /opt/nec/sysmng-soft/etc/host-list.xlsx ]` template file installed by the template package, and enter information related to all VHs you use. Save the `[ /opt/nec/sysmng-soft/etc/host-list.xlsx ]` file on the management server.

Caution
If you have already installed using *SX-Aurora TSUBASA Installation Guide (with OSS)*, skip this section.

The VH information file is an Excel file and it includes the following information.

Table 1: Items in the VH information file

Items	category	Tools by which the item is used.
Host Name	Required	Ansible, Power management tool
Group Name	Optional	Ansible
IP address	Not required	none
IP address for BMC	Optional	Power management tool
MAC address	Optional	Power management tool
Netmask of IP address	Not required	none
Gateway	Not required	none

- Host name must be registered.

- Specify group names if you do not install the same software programs on all VHs but want to select software programs to be installed based on VH roles.
- Specify BMC IP addresses if your power control tool uses BMC for power control.

Caution

- Do not change the file format of the template file.
- IP address is not used in this document. Keep the cell blank.
- Use the AA:BB:CC:DD:EE:FF format to enter MAC addresses.
- For optional items you do not use, leave their cells blank.

The VH information file created here is converted to files according to entry formats of tools described in " *2.2.10. Creating host list file* on page 20 ".

2.2.10. Creating host list file

On the management server, create all host list files described in this document by using the VH information file created in " *2.2.9. Creating the VH information file* on page 19 ". The host list files contain information about VHs referenced by Ansible and the power control tool in the entry format of each tools.

Execute the following command as the administrative user admin.

```
$ /opt/nec/sysmng-soft/bin/setup-hostlist
```

This command creates the following files.

`[ /opt/nec/sysmng-soft/etc/ansible/vh-hosts ]`

Host list file in the inventory file format defining hosts to be managed when Ansible is used to manage the VH configuration.

`[ /opt/nec/sysmng-soft/etc/cobimport.txt ]`

The Host list file for the cobimport.sh tool, which registers systems at once when Cobbler is used to install the OS on VHs. (This file is not used in this document.)

`[ /opt/nec/sysmng-soft/etc/vh_host.conf ]`

The host list file defining those controlled by the powerctrl tool when BMC is used to control power.

`[ /opt/nec/sysmng-soft/etc/vh_host_w.conf ]`

The host list file defining those controlled by the powerctrl tool when WOL is used to control power.

Caution
When the VH information file is modified, be sure to re-create the host list files by `setup-hostlist` command.

2.3. Initial Setup of VHs

This section explains the configuration for the management user admin on the management server to access VHs. The following procedures create the management user on VHs using Ansible as the rootuser on the management server.

Caution
If you have already installed using *SX-Aurora TSUBASA Installation Guide (with OSS)*, skip this section.

Please refer to " *2.4.1. Files for Ansible* on page 22 " for the files Ansible uses.

1. Ansible Performance Tuning

   Configure Ansible settings to reduce execution time of ansible. Set up Ansible as described in " *2.4.2. Tuning Ansible performance* on page 24 ".

However, pipeline function should not be enabled. The pipeline function might cause an error when you execute ansible-playbook command because requiretty setting is needed for root user on VHs.

2. Registration of VHs

   Register the target VHs to Inventory file. Please do the settings described in " *2.4.4. Registering VHs* on page 25 ".

3. Registration of the SSH public key of the Rootuser

   Register the SSH public key of the rootuser on the management server with VHs in order to access VHs via SSH as follows:

   a. Creation of the SSH Public and Private Key Pair of the Rootuser on the Management Server.

   ```
   # ssh-keygen -t rsa
   ```

   The public key is created in the file [~root/.ssh/id_rsa.pub].

   b. Registration of the Public Key with VHs

   Please run the following command to every VH.

   ```
   # ssh-copy-id -i ~/.ssh/id_rsa.pub vh-name01
   root@vh-name01's password: (Enter password)
   ```

   c. Modification of the SSH Configuration of the Rootuser
   Modify the file [~root/.ssh/config] as follows so that warning messages are not displayed at the access to VHs via SSH.

   ```
   # cat ~/.ssh/config
   Host *
     StrictHostKeyChecking  no
   # chmod 600 ~/.ssh/config
   ```

4. Execution of a Playbook to Configure the Management User

   A Playbook to configure the management user on VHs is provided as the file vh-add-admin.yml. This playbook performs the following tasks:

   a. Create the management user admin on VHs.
   b. Register the SSH public key of the management user with VHs .
   c. Grant sudo privilege s to the management user on VHs.
   d. Disable requiretty of the management user on VHs.

   Please run ssh-agent command and register the private key of the rootuser using ssh-add command. Then, perform vh-add-admin.yml with ansible-playbook command. At the execution of vh-add-admin.yml, enter the password of the management user admin.

   ```
   # ssh-agent bash
   # ssh-add  ~/.ssh/id_rsa
   Enter passphrase for /root/.ssh/id_rsa: (Enter the private key
    passphrase.)
   Identity added: /root/.ssh/id_rsa (/root/.ssh/id_rsa)
   # cd /opt/nec/sysmng-soft/etc/ansible
   # ansible-playbook vh-add-admin.yml
   Enter the password of new user admin: (Enter a password of the
    management user, admin.)
   confirm Enter the password of new user admin: (Enter the password
    again.)
   ```

   At this point, the management user admin is created on VHs.

   Caution
   The password of the management user admin is encrypted with hash in the Playbook. The sample file vh-add-admin.yml uses SHA512 hash.

5. Finalization

As the rootuser on the management server, please remove the public key and private key pair of the rootuser to disable the public key authentication of the rootuser at the access to VHs.

```
# rm ~/.ssh/id_rsa ~/.ssh/id_rsa.pub
```

2.4. Preparing for using Ansible

This section mainly explains the files Ansible uses and VH registration.

Caution
If you have already installed VHs by using SX-Aurora TSUBASA Installation Guide (with OSS), skip the following chapters.

" *2.4.2. Tuning Ansible performance* on page 24 "
" *2.4.4. Registering VHs* on page 25 "
" *2.4.5. Checking VH connectivity* on page 25 "

2.4.1. Files for Ansible

The following files are used in the Ansible's operation.

Inventory file	Text file with which managed machines are registered.
Playbook files	Text files describing how to configure managed machines in the YAML format.

Ansible guarantees idempotence by running playbooks that describe desired state of managed machines. Idempotence is the concept that "the same result is obtained even if an operation is performed multiple times", which is achieved by the configuration management tool. All steps written in playbooks are performed against new machines while only differential steps are performed against running machines.

The playbooks described in this document are provided as sample files, which are installed under `[ /opt/nec/sysmng-soft/etc/ansible ]` by the procedures in " *2.2.7. Installing the template package* on page 18 ". The following table shows directory layout and roles of sample playbooks.

Table 2: Structure of Sample Playbooks

File Names	Descriptions
vh-update.yml	Master playbook updating VHs; this playbook calls each role
group_vars/all	File defining common variables used across all playbooks
roles/TSUBASA-local-repo/<roles-dir>	Role configuring the settings for Ansible to reference the Yum repository for the SX-Aurora TSUBASA system; `[ /etc/yum.repos.d/TSUBASA-local.repo ]` and the GPG public key for the SX-Aurora TSUBASA software programs are configured with this role
roles/aurlic-lib/<roles-dir>	Role installing and configuring License access library.
roles/update-veos-apprun/<roles-dir>	Role updating VEOS Application Runtime. Before updating, VE is set to MAINTENANCE mode and VEOS and MMM are stopped. After updating, VEOS and MMM are restarted and VE is set to ONLINE mode.
roles/veos-appdev/<roles-dir>	Role updating VEOS Application Development.

File Names	Descriptions
roles/update-TSUBASA-InfiniBand/<roles-dir>	Role updating SX-Aurora TSUBASA InfiniBand.
roles/update-mmm/<roles-dir>	Role updating MMM. Before updating, VE is set to MAINTENANCE mode and VEOS and MMM are stopped. After updating, VEOS and MMM are restarted and VE is set to ONLINE mode.
roles/update-vmcfw/<roles-dir>	Role updating VMC Firmware. After updating, Firmware is updated and VH is rebooted.
roles/update-ScaTeFS-Client/<roles-dir>	Role updating ScaTeFS Client. After updating, VH is rebooted. (Refer to the below note.)
roles/nec-mpi/<roles-dir>	Role updating NEC MPI.
roles/tuningtool/<roles-dir>	Role updating tuning tool.
roles/parallel-debugger/<roles-dir>	Role updating NEC Parallel Debugger.
roles/update-NQSV-JobServer/<roles-dir>	Role updating NQSV/JobServer.
roles/update-NQSV-Client/<roles-dir>	Role updating NQSV/Client.
roles/NumericLibraryCollection/<roles-dir>	Role updating Numeric Library Collection.
roles/binutils/<roles-dir>	Role updating binutils.
roles/zabbix-agent/<roles-dir>	Role installing and updating Zabbix agent. This file will be used in " *3.1.3.2. Procedure with Ansible* on page 41 " (not used in this section).
roles/ganglia-gmond/<roles-dir>	Role installing and updating Ganglia agent. This file will be used in " *3.2.3.2. Setup Using Ansible* on page 59 " (not used in this section).
vh-add-admin.yml	Playbook to configure the management user on VHs.

Caution

The update-ScaTeFS-Client role does not support stopping nfs/smb/nmb services. See SX-Aurora TSUBASA Installation Guide for details.

Playbooks contain steps and variables related to software programs. Put playbooks in <roles-dir> described in the above table. Playbook directory layout described in the above table comply with the ones recommended by the Ansible official document. Main layout is as follows. Some files may not exist according to software program configurations.

Table 3: Structure of <roles-dir>

Files	Description
tasks/main.yml	File defining configuration steps
meta/main.yml	File defining dependent roles
handler/main.yml	File defining steps that are performed according to state changes
vars/main.yml	Variable definitions
templates/*	Template files written in Jinja2 style.

2.4.2. Tuning Ansible performance

When Ansible manages a large number of VHs, running playbooks takes time. Run the Ansible performance tuning to reduce playbook run time. This section describes typical methods of performance tuning. For details, see the Ansible documents.

Ansible connects to VHs over SSH and runs tasks in parallel; changing configurations of SSH connections and parallelism level helps reduce time for Ansible to run playbooks.

Edit the following items to the specified values in the Ansible configuration file, `[ /etc/ansible/ansible.cfg ]`,for performance tuning.

Table 4: Settings of ansible.cfg

Values	Description	Default	etc.
[defaults]			
forks = 100	Number of parallel tasks	5	Adjust the value according to the management server performance and the number of VHs.
timeout = 180	Timeout of SSH (in sec.)	10	
gathering = explicit	Gathering information of managed hosts implicit: explicit: disabled smart: gathering information only when the cache is disabled	implicit	
gather_subset = facter	Type of information items to gather from target hosts	all	
strategy = free	Task running methods linear: Ansible waits for all hosts to complete running tasks. free: Ansible doesn't wait.	linear	
[ssh_connection]			
pipelining = True	Using the pipeline function	False	VHs' requiretty (/etc/sudoers) needs to be disabled.

Ansible uses the SSH control master function by default as defined by the default ssh_args (SSH command arguments) values in the [ssh_connection] section. The specified values are as follows.

```
ssh_args = -C -o ControlMaster=auto -o ControlPersist=60s
```

Caution
To specify SSH command arguments for configuring Ansible, add arguments after the ones specified in the above command line. If the ControlMaster and ControlPersist are not defined, running playbooks takes time.

You need to disable the requiretty(/etc/sudoers) of VH administrative users to use the pipeline function. When administrative users are added to VHs, this setting is disabled by default. For details, refer to " *2.3. Initial Setup of VHs* on page 20 ".

2.4.3. Starting ssh-agent

Ansible used for setting the VH environment first logs in VHs over SSH for further operations. Start the `ssh-agent` and run the `ssh-add` command to register a SSH private key before running the Ansible command. Private key registration enables remote login without entering a passphrase. The private key to be used is created in " *2.2.8. Setting administrative user's SSH public and private keys* on page 19 ".

```
$ ssh-agent bash
$ ssh-add  ~/.ssh/id_rsa
Enter passphrase for /home/admin/.ssh/id_rsa: (Input passphrase)
Identity added: /home/admin/.ssh/id_rsa (/home/admin/.ssh/id_rsa)
```

2.4.4. Registering VHs

Ansible manages machines registered with the inventory file. The inventory file is a text file where you can specify target machines with host names or IP addresses and you can group machines as necessary.

Use vh-hosts, the Inventory file created in " *2.2.10. Creating host list file* on page 20 ", and change the Ansible's default inventory file to vh-hosts. Edit inventory on the Ansible configuration file `[ /etc/ansible/ansible.cfg ]` as follows.

```
inventory = /opt/nec/sysmng-soft/etc/ansible/vh-hosts
```

2.4.5. Checking VH connectivity

Use the `ping` module of the `ansible` command to check that remotely logging in the VH is possible. Check that nothing appears after running the following command.

```
$ ansible all -m ping -o | grep UNREACHABLE
```

When the `ping` module fails, the following message will be displayed.

```
vh-name01 | UNREACHABLE!
```

To see details of the error message, run the `ping` module without specifying -o against the host that failed to respond to the ping.

An example when a target host is not started:

```
$ ansible vh-name01 -m ping
vh-name01 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh.",
    "unreachable": true
}
```

An example when SSH authentication is not configured:

```
$ ansible vh-name01 -m ping
vh-name01 | UNREACHABLE! => {
    "changed": false,
    "msg": "ERROR! SSH encountered an unknown error during the
 connection. We recommend you re-run the command using -vvvv, which
 will enable SSH debugging output to help diagnose the issue",
    "unreachable": true
}
```

2.4.6. Starting Apache HTTP Server

SX-Aurora TSUBASA system software packages are downloaded from the HTTP server on the management server to each VHs in " *2.10. Software Update on VHs* on page 29 ". So start HTTP server on the management server.

```
# systemctl start httpd.service
```

2.5. Execution of Commands on VHs

`ansible` enables execution of commands on VHs from the management server.

- Execution of Commands as the Management User on the Management Server

  ```
  $ ansible target -a "command args"
  ```
- Execution of Commands as the Rootuser of the Management Server

  Run ansible command with --become option.

  ```
  $ ansible target --become -a "command args"
  ```
- Use of Environment Variables, Redirection, or Pipe

  Run ansible command with shell module.

  ```
  $ ansible target -m shell -a "command args | command args"
  ```

In the examples above, please specify VHs on which commands are executed for *target*. Hostnames and group names defined in the inventory file *vh-hosts* can be specified. Multiple targets or range specification is possible. Please refer to the document of Ansible.

2.6. File Manipulation on VHs

Ansible performs a role, a series of tasks, which is specified in a Playbook. The following are Playbook samples to manipulate files on VHs from the management server. Please create roles for file manipulation based on the samples, and specify them in the master Playbook *vh.yml*. Playbooks are performed using `ansible-playbook` command.

The following example is the steps to add file-ope role.

1. Creation of the Master playbook *vh-set.yml*

   Make `[ /opt/nec/sysmng-soft/etc/ansible/vh-set.yml ]`. Describe the following contents :

   ```
   ---
   - name: Set VH machine
     hosts: all
     become: yes
     roles:
       - file-ope
   ```
2. Creation of file-ope Role

   Make `[ roles/file-ope ]` directory and then create the following files:

   a. `[ roles/file-ope/tasks/main.yml ]`

   Define tasks to manipulate files. Please create necessary tasks based on the samples below.

   - Copy Files (copy module)

     In the case of copying files placed in the directory `[ roles/file-ope/files ]`

     ```
     - name: Copy the file from files directory in file-ope roles
       copy: src=file1 dest=/tmp/ansible/file1 owner=admin group=admin mode=0644
     ```

     In the case of copying files on the management server

     ```
     - name: Copy the file from any directory
     ```

```
  copy: src=/etc/hosts dest=/tmp/ansible/hosts.sample mode=0644
```

- Change attributes of files (file module)

```
- name: Set attributes for the file
  file: path=/tmp/ansible/file1 owner=root group=root state=file
```

- Delete files (file module)

```
- name: Delete the file
  file: path=/tmp/ansible/hosts.sample state=absent
```

- Update contents in files (lineinfile module)

```
- name: Update contents in the file
  lineinfile: >
    dest=/tmp/ansible/file1
    regexp='oldstring'
    line='newstring'
    backup=yes
```

  The `lineinfile` module replaces the last line which matches the regular expression specified for regexp, if any, with the string specified for line; Otherwise , it just adds the string after the last line. In order to replace multiple lines which match a regular expression, please use `replace` module.

- Update the SSH public key (authorized_key module)

```
- name: Update authorized_key from a local file
  authorized_key: >
    user=admin
    key="{{ lookup('file', '/home/admin/.ssh/id_rsa.pub') }}"
```

b. files under `[ roles/file-ope/files ]` directory

In the case of copying files, put the files to be copied into the directory roles/file-ope/files.

3. Execution of a Playbook

Execution of a Playbook performs the specified file manipulation only on VHs where the manipulation is necessary.

```
$ ansible-playbook vh-set.yml
```

Please refer to the following documents for the details of the Ansible modules explained above. Other modules for file manipulation are also described. Please use them as necessary.

- Ansible Official Document
- `ansible-doc` command on the management server

2.7. Software Installation on VHs

Please refer to SX-Aurora TSUBASA Environment Construction in *SX-Aurora TSUBASA Installation Guide (with OSS)* for the installation of software required for the SX-Aurora TSUBASA system operation.

This section explains how to install other software on VHs from the management server. The following is an example of installing the http package and starting the service.

Add a role for installing software to the sample Playbook `[ /opt/nec/sysmng-soft/etc/ansible ]`.

1. Creation of the Master playbook *vh-set.yml*

Make [/opt/nec/sysmng-soft/etc/ansible/vh-set.yml]. Describe the following contents :

```
---
- name: Set VH machine
  hosts: all
  become: yes
  roles:
    - httpd
```

If connections via a proxy server are required to obtain packages on VH machines, add the following setting to configure the proxy server.

```
  environment:
    http_proxy: http://proxy.example.com:8080
```

2. Creation of http role

   Make [roles/httpd] directory and create the following files:

   a. [roles/httpd/tasks/main.yml]

      Define the tasks to install httpd.

      ```
      ---
      - name: Install httpd
        yum: name=httpd state=latest
        notify:
          - Restart httpd and enabled
      ```

   b. [roles/httpd/handlers/main.yml]

      Define a handler to autostart and restart httpd.

      ```
      ---
      - name: Restart httpd and enabled
        service: name=httpd state=restarted enabled=yes
      ```

3. Execution of the Playbook

   Execution of the Playbook installs and starts httpd only on VHs where it has not yet been installed.

   ```
   $ ansible-playbook vh-set.yml
   ```

2.8. Software Uninstallation from VHs

This section explains how to uninstall software from VHs from the management server.

Please modify the role created in the previous section " *2.7. Software Installation on VHs* on page 27 ". The following is an example of uninstalling httpd.

1. Modification of httpd role

   Modify the file [roles/httpd/tasks/main.yml] as follows:

   a. Remove the task s for the installation
      Please remove all the tasks added in the previous section.
   b. Add tasks for the uninstallation

      Specify "httpd" for name option and "absent" for state option respectively in the yum module as follows:

      ```
      ---
      - name: Uninstall httpd
      ```

```
  yum : name=httpd state=absent
```

2. Execution of the Playbook

   Execution of the Playbook uninstalls httpd only from VHs where it has been installed.

```
$ ansible-playbook vh-set.yml
```

2.9. Update of Other Software

This clause explains how to update software other than the SX-Aurora TSUBASA system software on VHs from the management server.

Please modify the role created in " *2.7. Software Installation on VHs* on page 27 " The following is an example of updating httpd.

1. Check of the File roles/httpd/tasks/main.yml

   Check the role `[ roles/httpd/tasks/main.yml ]`

   Please make sure that latest is specified for state option in the task Install httpd. If present is specified, please change it to latest.

```
---
- name: Install httpd
  yum : name=httpd state=latest
  notify:
    - Restart httpd and enabled
```

2. Execution of the Playbook

   Execution of the Playbook updates and restarts httpd on VHs.

```
$ ansible-playbook vh-set.yml
```

2.10. Software Update on VHs

At the update of SX-Aurora TSUBASA software, some pre-processing and post-processing are required. For details, please refer to the 'Update' section of *SX-Aurora TSUBASA Installation Guide*.

A sample playbook is provided to update the SX-Aurora TSUBASA software on VHs from the management server. The sample playbook executes the required pre-processing and post-processing for each software before/after the update. However, the following processes are not covered by the playbook.

- Remove target VHs from job scheduler
- Remove target VHs from monitoring
- Re-include target VHs to monitoring
- Re-include target VHs to the job scheduler

When you use a job scheduler or a system monitoring tool, execute appropriate processings before/after the update by referring to the 'Update' section of *SX-Aurora TSUBASA Installation Guide*.

2.10.1. Editing Playbook

This section describes about the sample playbook files that are needed to be modified according to each environment.

1. Master Playbook

   Specify roles of software to be updated to roles:. To specify the roles, remove the '#' characters at the beginning of the software role lines.

[vh-update.yml]

```
---
- name: Update SX-Aurora TSUBASA Software on VH machine
  hosts: all
  become: yes
  gather_facts: true
  roles:
#    - aurlic-lib
#    - update-veos-apprun
#    - veos-appdev
#    - update-TSUBASA-InfiniBand
#    - update-mmm
#    - update-vmcfw
#    - update-ScaTeFS-Client
#    - tuningtool
#    - parallel-debugger
#    - update-NQSV-JobServer
#    - update-NQSV-Client
#    - NumericLibraryCollection
#    - binutils
#    - nec-mpi
```

If you get packages on VHs via a proxy server, set a proxy server by adding the following line to the master playbook.

```
  environment:
    http_proxy: http://proxy.example.com:8080
```

2. Common variables for all Playbooks

   Append the information of HTTP server (management server) IP address and the Yum repository directory for SX-Aurora TSUBASA that is made at " *2.2.5. Creating a Yum repository for SX-Aurora TSUBASA* on page 17 " to the master playbook. The Yum repository directory is the directory which contains the repodata. And specify it as the relative path from the document root directory of the HTTP server.

   [group_vars/all]

```
---
httpserver: http-server-ipaddr
reposdir: repos/TSUBASA
```

3. Each software roles

   Set the following variables in [vars/main.yml] files of each software roles. Please refer to the software's document for the details of the variables.

   a. aurlic-lib

   Set the variables below, if you update the license server.

   [roles/aurlic-lib/vars/main.yml]

variable name	value
aurlic_serverhost	The hostname of the license server
aurlic_serverport	Port number (This value should be set when you change the port number from the default value, 7300.)

   b. update-veos-apprun
   No setting required.

   c. update-veos-appdev
   No setting required.

   d. update-TSUBASA-InfiniBand
   No setting required.

e. update-mmm
No setting required.

f. update-ScaTeFS-Client
No setting required. But set the variables of aurlic-lib role.

g. tuningtool
No setting required.

h. parallel-debugger
No setting required.

i. update-NQSV-JobServer
No setting required.

j. update-NQSV-Client
No setting required.

k. NumericLibraryCollection
`[ roles/NumericLibraryCollection/vars/main.yml ]`

Variable name	Value
nlc_version	The version number included in the group ID of NEC Numeric Library Collection (The X-X-X part in the group ID of nec-nlc-X.X.X).

l. binutils
No setting required.

m. nec-mpi

Set the following variables to specify which versions of NEC MPI library group packages to install, and which versions to remove.

`[ roles/nec-mpi/vars/main.yml ]`

variable name	Value
mpi_inst_vers	The version numbers to install NEC MPI library group packages. The version number is included in the group ID of NEC MPI (The X-Y-Z part in the group ID of nec-mpi-X-Y-Z).
mpi_uninst_vers	The version numbers to remove NEC MPI library group packages. The version number is included in the group ID of NEC MPI (The X-Y-Z part in the group ID of nec-mpi-X-Y-Z).

Also set the variables of aurlic-lib role.

2.10.2. Execution of the Playbook

Software of SX-Aurora TSUBASA can be updated by executing `ansible-playbook` command with specifying the target VHs. By executing `ansible-playbook`, pre-update and post-update procedures are performed. Please refer to the 'Update' section in *SX-Aurora TSUBASA Installation Guide* for the details of the pre-update and post-update procedures.

- To update software programs on all VHs

```
$ cd /opt/nec/sysmng-soft/etc/ansible
$ ansible-playbook vh-update.yml
```

- To update software programs on specified VHs

```
$ ansible-playbook vh-update.yml -l vh-name01 vh-name02

$ ansible-playbook vh-update.yml -l 'vh-name0*'
```

```
$ ansible-playbook vh-update.yml -l group-name
```

When playbook execution fails, an error message like the one below appears and host names of failed VHs are recorded in the [*playbookname*.retry] file.

```
TASK [update-veos-apprun : Update VEOS Application Runtime] ***
fatal: [vh-name01]: FAILED! => { ... }
 :
  to retry, use: --limit @/opt/nec/sysmng-soft/etc/ansible/vh-
update.retry
```

To run the command again only against failed VHs, run the following command.

```
$ ansible-playbook vh-update.yml \
   --limit @/opt/nec/sysmng-soft/etc/ansible/vh-update.retry
```

Chapter 3
Operational Status Monitoring

This chapter explains how to monitor operational status of the SX-Aurora TSUBASA system using Zabbix or Ganglia+Nagios. In addition to usual monitoring items on Linux, it is also possible to monitor the operational status of VH and VE such as the core temperature, memory usage, and voltage value using plugins provided for the SX-Aurora TSUBASA system.

3.1. Zabbix

The configuration of Zabbix in the SX-Aurora TSUBASA system is shown in the following figure.

Zabbix server runs on the management server, and Zabbix agent on each VH. When there are 1000 or more VHs, please make a layered structure by using Zabbix-proxy. To monitor the management server, please configure Zabbix agent on the management server, too.

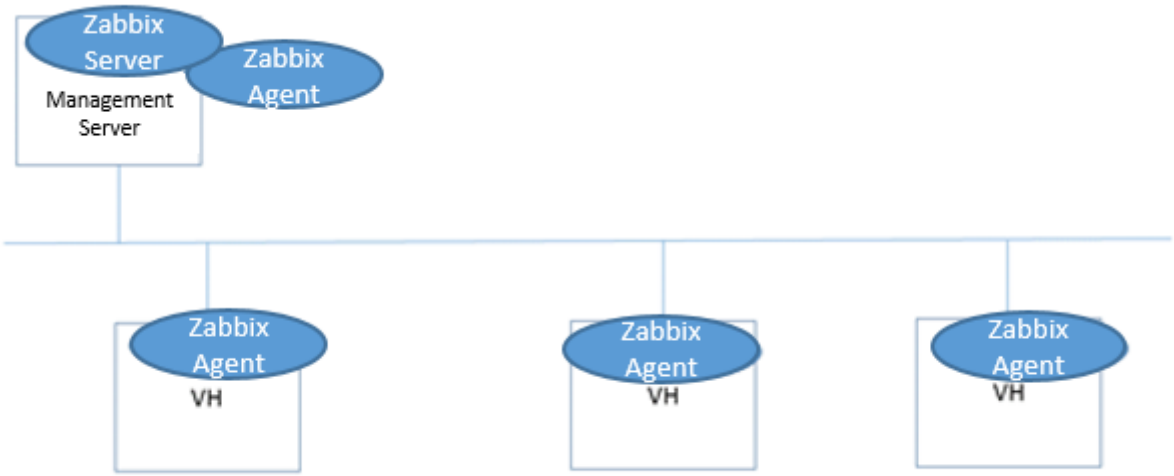

Figure 4: The Configuration of Zabbix

An example of a web interface screen shot (Dashboard) to set up Zabbix. This screen image is displayed for a user with administrative privileges. Displayed information and menus depend on user privileges. Please add users with appropriate privileges as needed. The work flow to enable the operational status monitoring of the SX-Aurora TSUBASA system using Zabbix is as follows:

1. Preparation
2. Initial Setup of Zabbix Server

   Set up Zabbix server on the management server.
3. Initial Setup of Zabbix Agent

   Set up Zabbix agent on VHs.
4. Setting of monitored VHs

   Specify monitored VHs via a Zabbix web interface.
5. Setup of monitoring items

   Set up monitoring items via a Zabbix web interface.

3.1.1. Preparation

1. Download

   Download the following Zabbix plugin package file onto an arbitrary directory on the management server.

   - TSUBASA-monitoring-Zabbix-<major>.<minor>.<release>.x86_64.rpm

     Download can be performed from the following website.

     - " *http://jpn.nec.com/hpc/aurora/ve-software/* " or
     - " *http://www.nec.com/en/global/prod/hpc/aurora/ve-software/* "

     The contents of the rpm package is as follows: Samples of loadable modules, configuration files, templates, value mapping files, and source files for each validated version of Zabbix are placed under the directory /opt/nec/sysmng-soft/monitoring-sample/zabbix/zabbix-3.0.11.

Table 5: List of sample files Installation Path

Installation Path				Description
etc/zabbix/ zabbix_agentd.d/			userparameter_veos.conf	A configuration file used on VHs for monitoring of services. Please refer to " *3.1.5.5. Monitoring of VEOS Services* on page 47 " for the usage.
selinux			Makefile	A makefile to create an SELinux policy module.
			zabbix-agent-TSUBASA.te	A type enforcement file in which a sample policy for execution of loadable modules is described.
			zabbix-agent-TSUBASA.pp	An SELinux policy module for the sample policy described in the above cell.
usr/	lib64/ zabbix/	modules/		The directory in which loadable modules are placed. Loadable modules are used on VHs. Please refer to " *3.1.5.3. Information Gathering with Loadable Modules* on page 46 " for the usage.
			ve_hw_item.so	A loadable module to measure the values of HW monitoring items.
			ve_os_item.so	A loadable module to measure the values of VEOS monitoring items.
		template/		The directory in which templates are placed. The templates are used on the management server. Please refer to " *3.1.5.1. Configuration of Monitoring Items Using Templates* on page 46 "for the usage.
			template_ve0.xml	Templates to monitor VEOS and HW.

Installation Path				Description
			...	
			template_ve7.xml	
			template_veos_service.xml	A template to monitor VEOS services
		mapping/	zbx_export_valuemaps.xml	A value mapping file, which is used on the management server. Please refer to " *3.1.5.1. Configuration of Monitoring Items Using Templates* on page 46 " for the usage.
	src/zabbix/modules/ve/			The directory in which sample sources are placed. You can create customized loadable modules from the sources to add arbitrary monitoring items. Please refer to " *3.1.9. Creation of Loadable Modules from Source Files* on page 49 " for the usage.
			ve_hw_item.c	A source file of functions to measure the values of HW monitoring items
			ve_os_item.c	A source file of functions to measure the values of VEOS monitoring items
			ve_common.c	A source file of commonly used functions
			ve_item.h	A header file
			Makefile	A makefile

3.1.2. Initial Setup of Zabbix Server

This clause explains how to install and configure Zabbix server on the management server.

3.1.2.1. Preparation

Before the installation, please set up the following packages.

1. Setting of firewall

   If firewall is running, only ssh is allowed in the public zone by default. Please add HTTP (80/tcp) and zabbix-trapper (10051/tcp). In the case of installing Zabbix agent on the management server, add Zabbix-agent (10050/tcp), too.

```
# firewall-cmd --zone=public --add-port=80/tcp --permanent
success
# firewall-cmd --zone=public --add-port=10051/tcp --permanent
success
# firewall-cmd --zone=public --add-port=10050/tcp --permanent
success
# firewall-cmd --reload
success
```

If firewall is not needed, please stop it.

2. Configuration of Time Synchronization

   Please configure time synchronization using ntp or chrony, since Zabbix server obtains and stores information based on the OS time.

3. Setup of Email

   Please set up an email server on the management server so as to receive email notifications from Zabbixserver.

4. Setup of Apache HTTP Server

   Zabbix server needs HTTP server for Zabbix Web Interface. Please install the Apache HTTP server package and start httpd. The following is an example of the installation using yum:

   ```
   # yum -y install httpd
   # systemctl start httpd.service
   ```

5. Setup of Database

   Zabbix server stores information about the configuration and data gathered from monitored VHs in a database. MariaDB, which is compatible with MySQL, is the standard database in Red Hat Enterprise Linux 7 (RHEL7), and the following are how to set up MariaDB.

   a. Installation of MariaDB

   Please install the following packages.

   ```
   # yum -y install mariadb-server mariadb mariadb-devel mariadb-libs
   ```

   b. Performance Tuning

   It is recommended that MariaDB be configured so as to reduce I/O load, because a lot of update queries occur in Zabbix.

   Add the following lines in [mysqld] section in the file `[ /etc/my.cnf ]`. Note that if MariaDB is already put to other uses, the configuration for Zabbix can conflict with that for the others. Please adjust the settings according to the environment.

   The first two lines are strongly recommended for Zabbix server.

   ```
   [mysqld]
   innodb_file_per_table
   character-set-server=utf8

   innodb_log_buffer_size=16M
   innodb_buffer_pool_size=1024M
   innodb_log_file_size=256M
   innodb_log_files_in_group=2
   skip-character-set-client-handshake
   ```

   c. Start of MariaDB Service

   Start the MariaDB service as follows:

   ```
   # systemctl start mariadb.service
   ```

   d. Autostart of the MariaDB Service

   Enable autostart of the MariaDB service on the boot of the management server as follows:

   ```
   # systemctl enable mariadb.service
   ```

   e. Setting of the Root Password

   Set the root password with the security improvement tool mysql_secure_installation at the initial installation of MariaDB.

   ```
   # mysql_secure_installation
   ```

If MariaDB is already in operation, just check the root password.

6. PHP

   Set up PHP so as to enable Zabbix web interfaces as follows:

   a. Installation

      Install the following packages

```
# yum -y install php php-devel php-pdo php-mysql php-mbstring
 php-gd php-pear zlib-devel
```

   b. Configuration

      Specify the following values, which are recommended for Zabbix, in the file `[ /etc/php.ini ]`, and then restart httpd. Please specify your time zone for the timezone.

```
date.timezone = Asia/Tokyo
post_max_size = 16M
max_execution_time = 300
max_input_time = 300
```

3.1.2.2. Installation of Zabbix Server

The following are how to install Zabbix server using Zabbix repository.

1. Installation of the Zabbix Repository Configuration Package

   Please get the repository configuration package (zabbix-release-3.0-1.el7.noarch.rpm) from Zabbix official website, and install the package by rpm command.

```
# rpm -ivh zabbix-release-3.0-1.el7.noarch.rpm
```

2. Installation of Zabbix Server

   Install the following packages. Please note that the installation of packages provided by Red Hat Network (RHN) can be required to eliminate dependency among packages. In this case, install Zabbix server after the installation of the required packages.

```
# yum -y install zabbix-server-mysql-3.0.11 zabbix-web-mysql-3.0.11
```

   If you want to display menus in Japanese, install the following package, too.

```
# yum -y install zabbix-web-japanese-3.0.11
```

   If you want to monitor the management server itself, install Zabbix agent as follows. Please refer to " *3.1.3. Initial Setup of Zabbix Agent* on page 38 " for the configuration of Zabbix agent.

```
# yum -y install zabbix-agent-3.0.11
```

3. Creation of a Database for Zabbix Server

   Access MariaDB as the root user, and create a user zabbix and a database for Zabbix server. The password for the user zabbix also needs to be set appropriately. The password *zabbix-pass* is set in the following example.

```
# mysql -u root -p
Enter password : Enter the root password
mysql> create database zabbix character set utf8 collate utf8_bin;
mysql> grant all privileges on zabbix.* to zabbix@localhost
 identified by 'zabbix-pass';
mysql> quit;
```

4. Import of the Initial Data for Zabbix Server

   Import the initial data for the database for Zabbix server as follows, specifying the username zabbix and the password *zabbix-pass*.

```
# cd /usr/share/doc/zabbix-server-mysql-3.0.11
# gzip -d create.sql.gz
# mysql -u zabbix -p zabbix < create.sql
Enter password:   Input zabbix-pass
```

5. Setting of a Password for the Database

   Specify a password to access the database in the file `[ /etc/zabbix/zabbix_server.conf ]`. The password *zabbix-pass* is specified in the following example.

```
DBPassword=zabbix-pass
```

6. Setting of the Timezone

   Specify your time zone in the file `[ /etc/httpd/conf.d/php.conf ]`.

```
php_value date.timezone Asia/Tokyo
```

3.1.2.3. Start of Zabbix Server

1. Start of Zabbix Server

   Start Zabbix server as follows. Restart httpd as the files related to PHP are updated at the installation of Zabbix server.

```
# systemctl start zabbix-server
# systemctl restart httpd
```

2. Autostart of Zabbix Server

   Enable autostart of Zabbix server on the boot of the management server as follows:

```
# systemctl enable zabbix-server
```

3. Restart of Zabbix Server

   After update of the configuration file of Zabbix server, please restart Zabbix server as follows:

```
# systemctl restart zabbix-server
```

3.1.2.4. Initial Setup of Zabbix Server

1. Confirmation of Connection to Zabbix Server

   Please confirm that the following URL is accessible using a browser.

```
  http://the IP address or hostname of Zabbix server/zabbix/
```

2. Configuration of Zabbix Server

   The above URL displays the configuration screen of Zabbix server. Please configure Zabbix server while referring to the Zabbix official document.

3. Configuration of Users

   The account Admin, which has administrative privileges, is initially registered with Zabbix server. The initial password is zabbix. Please change the password appropriately and add users with appropriate privileges to access and change the configuration of monitored VHs while referring to the Zabbix official document.

3.1.3. Initial Setup of Zabbix Agent

This clause explains how to install and configure Zabbix agent.

It can be done on each VH manually or on all VHs at once from the management server using Ansible.

3.1.3.1. Manual Setup

Zabbix Agent can be manually configured on each VH as follows:

Installation

1. Setting of firewall

   If firewall is running, only ssh is allowed in the public zone by default. Please add zabbix-agent (10050/tcp).

   ```
   # firewall-cmd --zone=public --add-port=10050/tcp --permanent
   success
   # firewall-cmd --reload
   success
   ```

   If firewall is not needed, please stop it.
2. Installation of the Zabbix Repository Configuration Package

   Please get the repository configuration package (zabbix-release-3.0-1.el7.noarch.rpm) from Zabbix official website, and install the package by rpm command.

   ```
   # rpm -ivh zabbix-release-3.0-1.el7.noarch.rpm
   ```
3. Installation of Zabbix Agent

   Please install the following package.

   ```
   # yum -y install zabbix-agent-3.0.11
   ```
4. Installation of libyaml

   Please install the following package.

   ```
   # yum -y install libyaml
   ```
5. Installation of the Zabbix Plugin Package

   Please install the rpm package `[ TSUBASA-monitoring-Zabbix-<major>.<minor>.<release>.x86_64.rpm ]`, which is downloaded in" *3.1.1. Preparation* on page 34 ".

   ```
   # rpm -ivh TSUBASA-monitoring-Zabbix-<major>.<minor>.<release>.x86_64.rpm
   ```
6. Configuration of a SELinux Policy

   If SELinux is configured to be Enforcing, configure an SELinux policy using the Zabbix plugin so that Zabbix agent can access VE information. If SELinux is configured to be Permissive or Disabled, this step can be skipped.

   The plugin includes the following three sample files for the SELinux policy:

   - `[ zabbix-agent-TSUBASA.te ]` (A policy definition file)
   - `[ Makefile ]` (A makefile to generate a policy module from the above file)
   - `[ zabbix-agent-TSUBASA.pp ]`(A policy package)

   The samples have the definitions of the policy required for Zabbix agent to have access to VE information.

   If a site specific policy for Zabbix agent is not needed, execute the following command to apply the sample policy:

   ```
   # semodule -i /opt/nec/sysmng-soft/monitoring-sample/zabbix/
   zabbix-3.0.11/selinux/zabbix-agent-TSUBASA.pp
   ```

   If a site specific policy for Zabbix agent is needed, please create a policy package for the SELinux policy as follows:

a. Modification of the Policy Definition File

Copy the sample policy definition file and makefile into a working directory, and modify the policy definition.

In the following example, the working directory is [/tmp/selinux].

```
# cd /opt/nec/sysmng-soft/monitoring-sample/zabbix/
zabbix-3.0.11/selinux/
# cp zabbix-agent-TSUBASA.te /tmp/selinux
# cp Makefile /tmp/selinux
# cd /tmp/selinux
# vi zabbix-agent-TSUBASA.te
```

b. Creation of a Policy Package

Execute make command at the working directory to create a policy package whose name is [zabbix-agent-TSUBASA.pp]

```
# cd /tmp/selinux
# make
```

c. Application of the Policy Package

Execute semodule command to enable the SELinux policy.

```
# semodule -i zabbix-agent-TSUBASA.pp
```

Zabbix Agent Configuration File

1. Association of Zabbix Agent with Zabbix Server

Please specify the IP address or hostname of Zabbix server in the file [/etc/zabbix/zabbix_agentd.conf] to allow access from Zabbix server.

```
Server=the IP address or hostname of Zabbix server
```

2. Activation of the Active check Function

If you want to make Zabbix agent notify Zabbix server, which is called active check, please add the following lines:

```
Hostname=the hostname of Zabbix agent
ServerActive=the IP address of Zabbix server
```

The specification of "Hostname" must be the same as the hostname of the VH specified in " *3.1.4.1. Registration of Hosts* on page 45 ".

Start of Zabbix Agent

1. Start of Zabbix Agent

Start Zabbix agent as follows:

```
# systemctl start zabbix-agent
```

2. Autostart of Zabbix Agent

Enable autostart of Zabbix agent on the boot of the VH as follows:

```
# systemctl enable zabbix-agent
```

3. Restart of Zabbix Agent

After update of the configuration file of Zabbix Agent, please restart Zabbix agent as follows:

```
# systemctl restart zabbix-agent
```

3.1.3.2. Procedure with Ansible

By the following procedure, the settings of Zabbix agent for multiple VHs can be executed from the management server in a lump.

1. Settings to use Ansible

   Execute the following settings to use Ansible. These settings should be executed just once.

   a. " *2.2. Initial Setup of the Management Server* on page 16 "
   b. " *2.3. Initial Setup of VHs* on page 20 "
   c. " *2.4. Preparing for using Ansible* on page 22 "

   In this procedure, the files under `[ /opt/nec/sysmng-soft/etc/ansible ]` that are installed at " *2.2.7. Installing the template package* on page 18 "

   - Master Playbook `[ vh.yml ]`
   - Common variables for all Playbooks `[ group_vars/all ]`
   - Role for settings of Zabbix agent `[ roles/zabbix-agent ]`

   The target VHs are specified in the Inventory file `[ /opt/nec/sysmng-soft/etc/ansible/vh-hosts ]` that is made in " *2.2.10. Creating host list file* on page 20 ".

2. Locating TSUBASA-monitoring-Zabbix-X.X-X.x86_64.rpm and creating yum repository

   The zabbix-agent role installs TSUBASA-monitoring-Zabbix-X.X-X.x86_64.rpm in the Yum repository on the management server to each VHs. In order to do that, locate the TSUBASA-monitoring-Zabbix-X.X-X.x86_64.rpm that is downloaded in " *3.1.1. Preparation* on page 34 " to the yum repository for SX-Aurora TSUBASA on the management server. And update the repository using createrepo command.

```
# cd /var/www/html/repos/TSUBASA
# ls rpms
 :
TSUBASA-monitoring-Zabbix-X.X-X.x86_64.rpm
 :
# createrepo -v -g TSUBASA-groups.xml /var/www/html/repos/TSUBASA
```

3. Settings of the common Playbook variable

   Set the http server (= the management server) and the yum repository directory in `[ group_vars/all ]`. The yum repository directory means the directory under which the repodata exists. And it should be specified by a relative path from the document root directory of the http server.

```
---
httpserver: IP address of the http server
reposdir: repos/TSUBASA
```

4. Update master playbook vh.yml

   Append the role of zabbix-agent

```
---
- name: Set VH machine
  hosts: all
  become: yes
  roles:
    - TSUBASA-local-repo
    - mmm
    - zabbix-agent
    :
```

If connections via a proxy server are required to obtain packages on VH machines, add the following setting to configure the proxy server.

```
environment:
  http_proxy: http://proxy.example.com:8080
```

5. Settings of variables for zabbix-agent role

Set the following variables in [roles/zabbix-agent/vars/main.yml]

Name of variable	Value
zabbix_agent_version	Version of zabbix-agent
zabbix_agent_server	The value set for 'Server' variable in the Zabbix configuration file, [/etc/zabbix/zabbix_agentd.conf]. For details, please refer to " *Zabbix Agent Configuration File* on page 40 ".

Please enable the following variable if you use the Active Check. For details, please refer to " *Zabbix Agent Configuration File* on page 40 ".

variable	Value
zabbix_agent_hostname	The value set for the 'Host' variable in Zabbix configuration file, [/etc/zabbix/zabbix_agentd.conf]. The *inventory_hostname* used in this variable is a built-in variable of Ansible. This is expanded to a hostname of the target VH.
zabbix_agent_serveractive	The value set for the 'ServerActive' variable in Zabbix configuration file, [/etc/zabbix/zabbix_agentd.conf].

Confirm the settings of " *3.1.5.3. Information Gathering with Loadable Modules* on page 46 ". Check the following variables are set to enable.

variable	value
zabbix_agent_loadmodulepath	The value set for the 'LoadModulePath' variable in Zabbix configuration file, [/etc/zabbix/zabbix_agentd.conf]. No deed to change.
zabbix_agent_loadmodules	The value set for the 'LoadModule' variable in Zabbix configuration file, [/etc/zabbix/zabbix_agentd.conf]. No deed to change.

Set the following variables for the function of " *3.1.5.5. Monitoring of VEOS Services* on page 47 ".

Variable	Value
zabbix_agent_mon_veos	*enabled* or *disabled*

6. Execution of Playbook

Run playbook by administrative user admin, so the settings described in " *3.1.3.1. Manual Setup* on page 39 ", " *3.1.5.3. Information Gathering with Loadable Modules* on page 46 " and " *3.1.5.5. Monitoring of VEOS Services* on page 47 " are performed and Zabbix agent is started on each VHs.

```
$ cd /opt/nec/sysmng-soft/etc/ansible
$ ansible-playbook vh.yml
```

3.1.4. Configuration of Host Information

Monitored equipment needs to be registered as a host in Zabbix.

There are two methods of monitoring sensor information of VEs. One is to register VEs as hosts in Zabbix, in addition to VHs. The other is to specify sensor information of VEs as monitoring items of the VH to which the VEs are connected.

In the case of registering VEs as hosts, hosts are configured as shown in orange in the following figure.

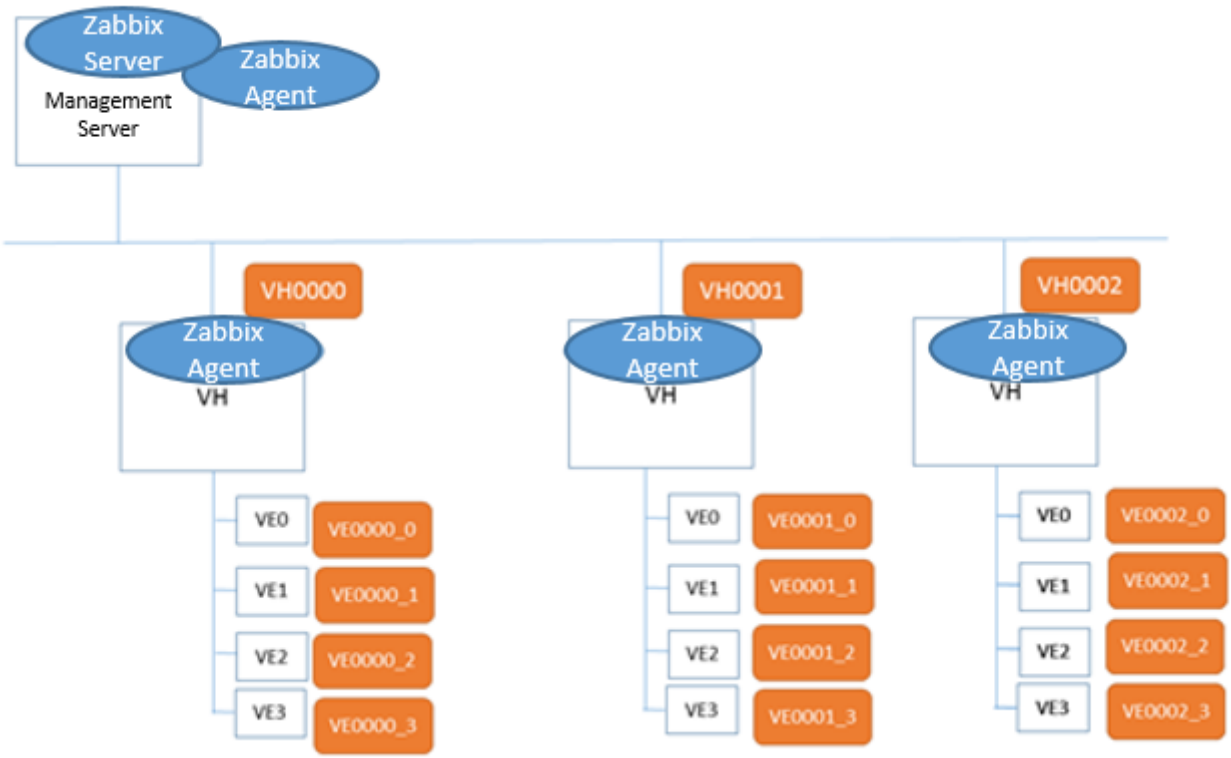

Figure 5: VEs are registered as hosts

In the case of not registering VEs as hosts, hosts are configured as shown in orange in the following figure.

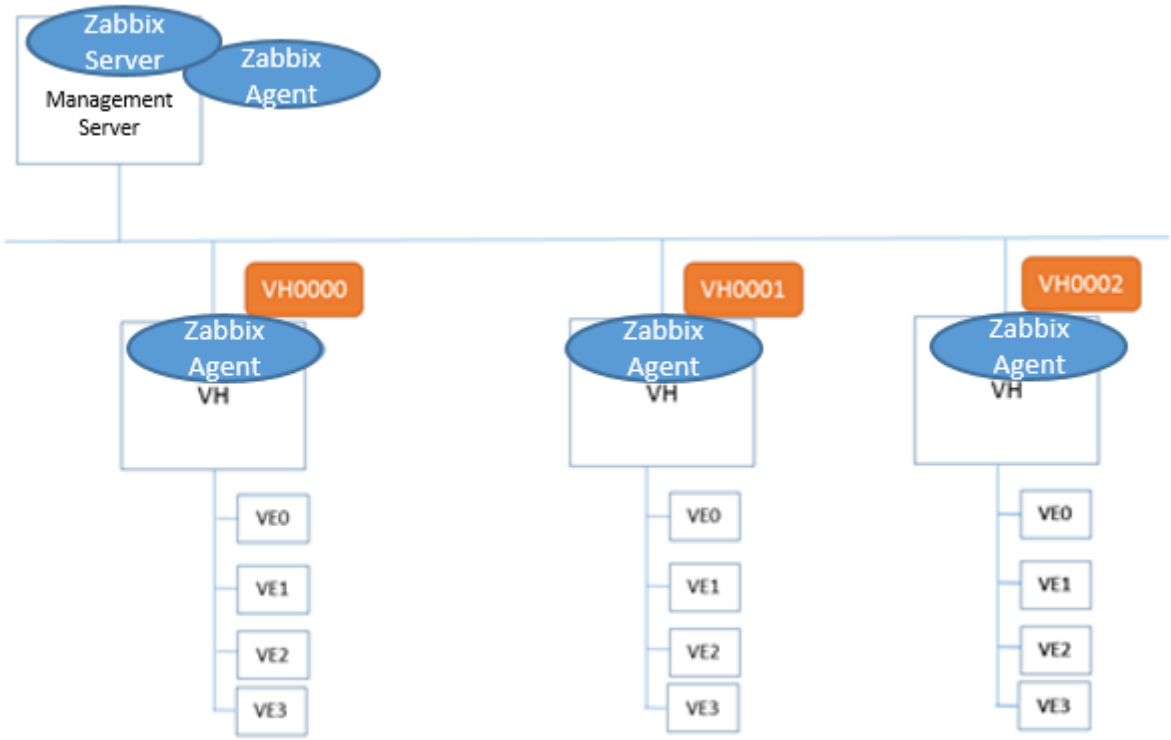

Figure 6: VEs are not Registered as Hosts

Registered hosts can be organized into multiple host groups and need to belong to one or more host groups in Zabbix. Two examples of the configurations of host groups are shown in the red frames in following figures.

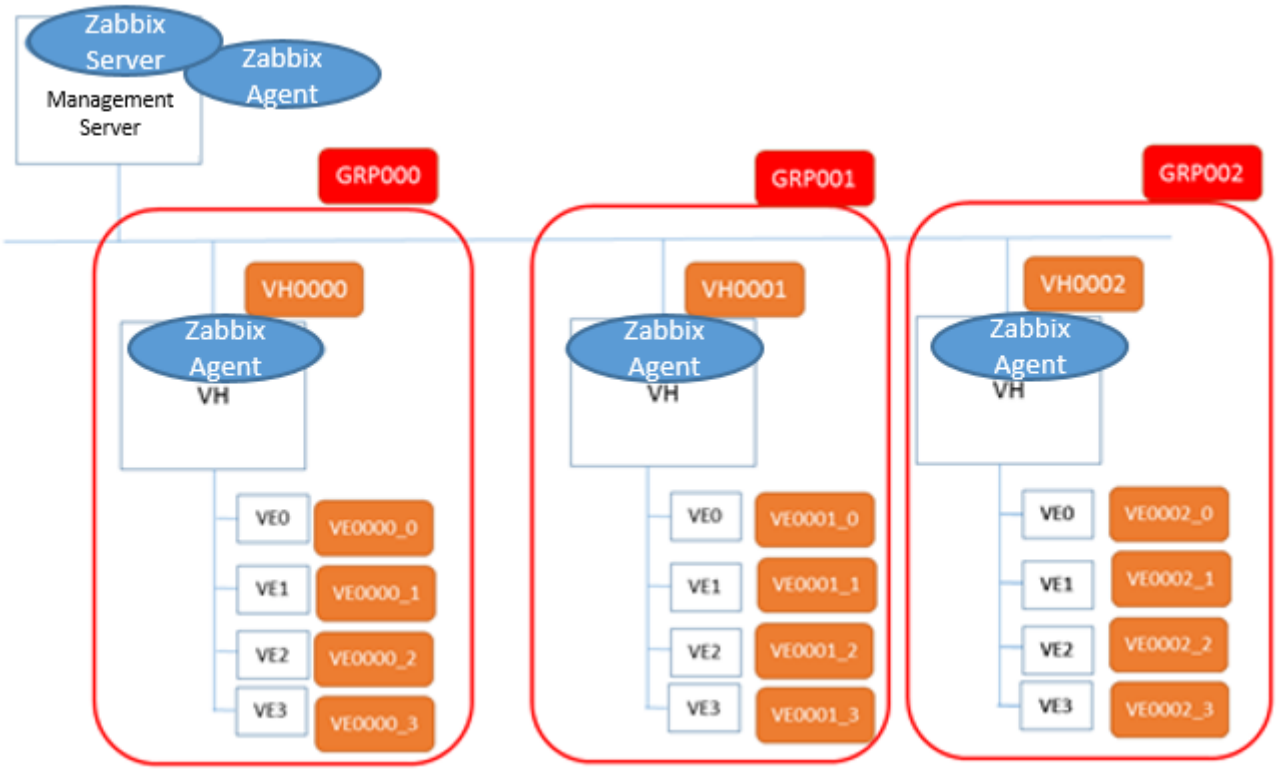

Figure 7: Example 1: Configuration of Host Groups

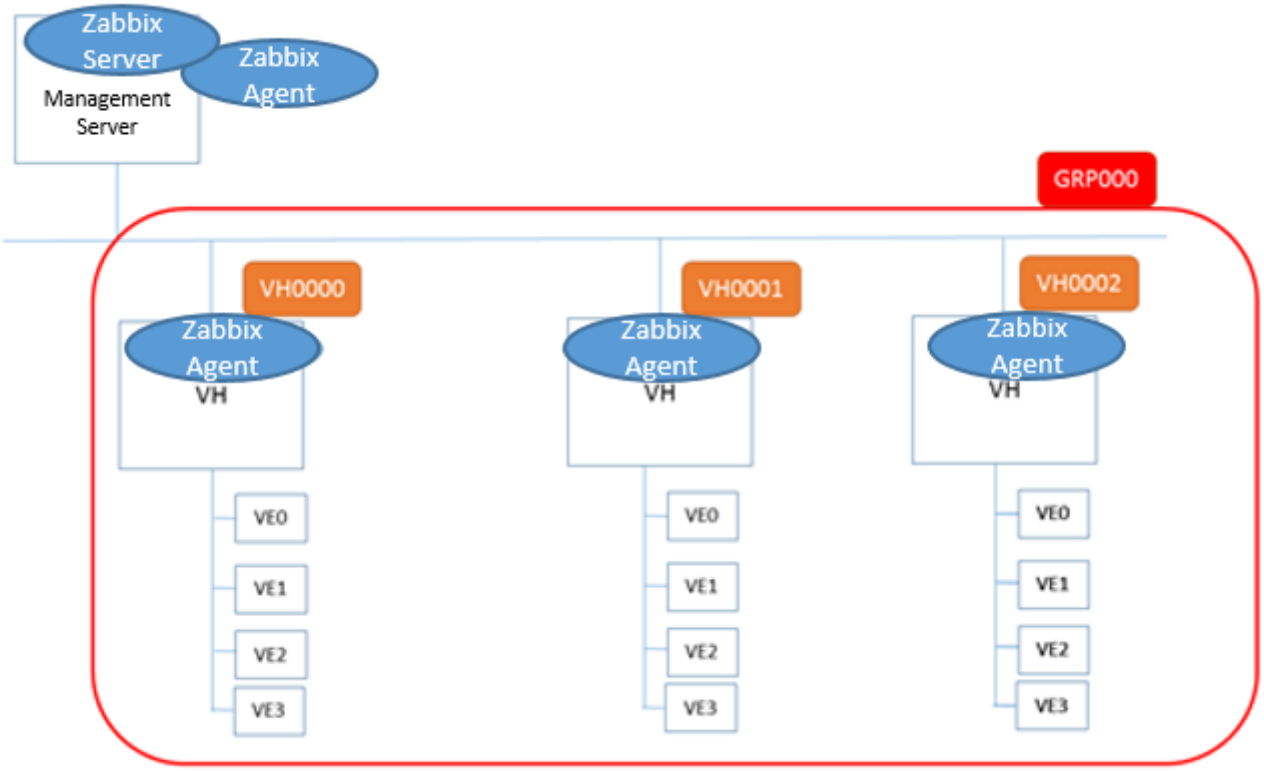

Figure 8: Example 2: Configuration of Host Groups

Various management idea can make by the host information and group information. Please configure to suit your site.

3.1.4.1. Registration of Hosts

Register VHs as hosts via the Zabbix web interface on the management server. If you want to configure VEs as hosts, register VEs as hosts, too.

1. Registration of VHs as Hosts

   Please register the IP addresses or hostnames of VHs managed by the management server. Each hostname needs to be the same as that specified for Hostname in the file `[ /etc/zabbix/zabbix_agentd.conf ]` so that Zabbix server recognizes the correspondence between VHs and Zabbix agents.

   Also specify one or more host groups to which each VH belongs. Host groups can be added or changed after the beginning of operation. Please refer to the Zabbix official document.
2. Registration of VEs as Hosts

   In the registration of VEs as hosts, please specify the IP address of the VH to which each VE is connected because Zabbix server obtains information of VEs from the VH. VEs are associated with their host names using monitoring items.

   Also, specify one or more host groups to which each VE belongs. Host groups can be added or changed after the beginning of operation. Please refer to the Zabbix official document.

3.1.5. Configuration of Monitoring Items

To configure monitoring items for VHs and VEs, templates of monitoring items need to be linked with monitored devices via the Zabbix server web interface, and the load module needs to be attached to zabbix agent. Please refer to " *3.1.5.1. Configuration of Monitoring Items Using Templates* on page

46 " and " *3.1.5.3. Information Gathering with Loadable Modules* on page 46 " for how to link templates and attach the module.

Please also refer to " *3.1.5.2. Addition of Monitoring Items* on page 46 " for how to add monitoring items that are not included in the templates and

" *3.1.5.5. Monitoring of VEOS Services* on page 47 " for the configuration of monitoring items for VEOS services.

3.1.5.1. Configuration of Monitoring Items Using Templates

Link the Template for Linux OS with Host. And, Link the Templates for VEs which are included in the Zabbix plugin package file.

1. Linking of the Template for Linux OS with Hosts

   Link the template for Linux OS "Template OS Linux", which is included in the Zabbix standard release, with VH hosts. Please refer to the Zabbix official document.
2. Import of Value Mappings

   The templates for VEs use value mappings, which are the correspondence between monitoring items and displayed strings, specific to VEs. The value mappings are defined in the file "zbx_export_valuemaps.xml" included in the Zabbix plugin package file. Please import the file. Please refer to the Zabbix official document.
3. Linking of Templates for VEs

   Link templates for VEs, which are included in the Zabbix plugin package file, with VEs. In the case that VEs are registered as hosts as shown in " *Figure 5: VEs are registered as hosts* on page 43 ", please link each template with the VE of the corresponding VE number. In the case that monitoring items for VEs are included in those for VHs as shown in " *Figure 6: VEs are not Registered as Hosts* on page 44 ", please link each template for VEs with the VH to which the corresponding VE is connected.

   Please refer to " *A.2. The Items and Triggers Provided by the Templates* on page 82 " or the details of monitoring items and triggers defined in the templates. Triggers are the conditions to refresh the current system status.

   Please refer to " *3.1.5.2. Addition of Monitoring Items* on page 46 " for how to add monitoring items.

   Please refer to " *3.1.5.6. Performance Tuning* on page 48 "for how to delete monitoring items or change monitoring intervals.

3.1.5.2. Addition of Monitoring Items

Please add monitoring items which are not included in the templates while referring to the Zabbix official document. For example, the monitoring items in the templates do not include monitoring of specific processes or logs.

3.1.5.3. Information Gathering with Loadable Modules

The Zabbix plugin package file " *3.1.1. Preparation* on page 34 " includes loadable modules(ve_hw_item.so, ve_os_item.so) to gather values of monitoring items specific to the TSUBASA system. The loadable modules include all the monitoring items described in " *A.1. The Item Keys Provided by the Loadable Modules* on page 75 ".

1. Installation of the Loadable Modules

   Please install the Zabbix plugin package file.
2. Configuration

Please specify the path to the loadable modules for "LoadModulePath" and the file names of the loadable modules for "LoadModule" in the file [/etc/zabbix/zabbix_agentd.conf] as follows:

```
# LoadModulePath=${libdir}/modules
LoadModulePath=/opt/nec/sysmng-soft/monitoring-sample/zabbix/
zabbix-3.0.11/usr/lib64/zabbix/modules/

# LoadModule=
LoadModule=ve_hw_item.so
LoadModule=ve_os_item.so
```

3. Restart of Zabbix Agent

   Restart Zabbix agent for the configuration to take effect.

```
#systemctl restart zabbix-agent
```

3.1.5.4. Information Gathering with User Parameters

Monitoring items can also be specified using the command described in Appendix A, instead of using the loadable modules.

It will be useful in the case that the number of monitoring items is small.

1. Configuration

   To monitor the number of VEs and cores on each VE, for example, please add the following lines in the file [/etc/zabbix/zabbix_agentd.conf]. Please configure the monitoring items using the command according to the user site.

```
# UnsafeUserParameters=0
UserParameter=ve_Attached_num[*],/opt/nec/ve/bin/vecmd info
 | awk /Attached Ves/ ' { n=split($$0,a," : ");  print a[2];
 exit }'
UserParameter=ve_cores[*],/opt/nec/ve/bin/vecmd -N $1 info |
 awk / Cores/ ' { n=split($$0,a,": ");  print a[2]; exit }'
```

3.1.5.5. Monitoring of VEOS Services

To monitor services or logs of VEOS, please configure as follows:

1. Installation of the Zabbix Plugin Package File

   Install the Zabbix plugin package file.

2. Copy of Configuration File

   The configuration file for monitoring of VEOS services is [/opt/nec/sysmng-soft/monitoring-sample/zabbix/zabbix-3.0.11/etc/zabbix/zabbix_agentd.d/userparameter_veos.conf]. Copy it into the directory [/etc/zabbix/zabbix_agentd.d].

```
# cp /opt/nec/sysmng-soft/monitoring-sample/zabbix/zabbix-3.0.11/
etc/zabbix/zabbix_agentd.d/userparameter_veos.conf /etc/zabbix/
zabbix_agentd.d
```

3. Activation of the Active Check Function of Zabbix Agent

Please enable the active check function of Zabbix agent while referring to " *Zabbix Agent Configuration File* on page 40 "

4. Restart of Zabbix Agent

   Restart Zabbix agent for the configuration to take effect.

```
# systemctl restart zabbix-agent
```

5. Linking of the Templates for Monitoring VEOS Services

   ease link the templates for monitoring VEOS services via the Zabbix server web interface. Please refer to the Zabbix official document how to link templates. The monitoring items defined in the templates are described in " *A.2. The Items and Triggers Provided by the Templates* on page 82 ".

Caution
Please give the root privileges or read permission of `[ /var/log/messages ]` to Zabbix agent so that it can monitor the logs of MMM.

3.1.5.6. Performance Tuning

A lot of monitoring items are configured by linking the templates for the SX-Aurora TSUBASA system. The loadable modules enable all the possible monitoring items. However, monitoring too many items will burden the system. Therefore, please reduce monitoring items or change monitoring intervals while referring to the Zabbix official document.

Zabbix server stores history data, trend data, and event data in MariaDB. The amount of disk space required for the management server can be estimated as follows:

1. History Data

   History data is raw monitoring data gathered by Zabbix. The amount of disk space for history data can be estimated as follows:

   (The number of monitoring items * the amount per monitoring item [Byte]) / Average monitoring intervals [second] * (3600 [second] * 24 [hour] * the retention period [day])

   In addition, the amount for log data is needed to monitor logs or SNMP traps.

2. Trend Data

   Trend data is the statistical data per hour of numerical values in the history data (maximum, minimum, and average), and used for drawing graphs. The amount can be estimated as follows:

   (The number of monitoring items * the amount per trend data [Byte]) * 24 [hour] * the retention period [day])

   The amount per trend data depends on the database, but is normally around 128 bytes.

3. Event Data

   The amount of event data can be estimated as follows. As it is difficult to predict the number of failures, the worst case estimation is recommended.

   (The number of failures and recoveries per day * the amount per event [Bye] * the retention period [day])

   The amount per event depends on the database, but is normally around 130 bytes.

3.1.6. Configuration of Triggers

Triggers are the conditions to refresh the current system status. Please add triggers which are not included in the templates, or modify the thresholds or conditions for the preset triggers as necessary while referring to the Zabbix official document. Please refer to " *A.1. The Item Keys Provided by the Loadable Modules* on page 75 " for the triggers preset in the templates.

3.1.7. Configuration of Actions

Please configure actions, which is the actions of Zabbix to the system administrator on detection of failure. Templates for actions are not included in the Zabbix plugin package. Please configure actions while referring to the Zabbix official document as necessary.

3.1.8. Customization of the Web Interface

It is possible to customize the web interface so as to display various data such as the network map of a system in Zabbix.

It is also possible to display data in hierarchical representation such as (Cluster map) - (Rack map) - (VH and VE map). Please create interfaces for users with various privileges such as the system administrator, maintenance service staff, and administrators of a small set of VHs so that they can easily monitor the system status.

It is helpful to create a screen in which multiple monitoring items are summarized to check information which dynamically changes and is regularly updated. For example, a screen to see the status of all VEs at a glance can make it unnecessary to find the information among a lot of monitoring data.

It is possible to see created maps and screens easily by placing them on the dashboard. Please refer to the Zabbix official document.

3.1.9. Creation of Loadable Modules from Source Files

Source files of loadable modules are placed in the directory `[ /opt/nec/sysmng-soft/monitoring-sample/zabbix/zabbix-3.0.11/usr/src/zabbix/modules/ve ]` by installing the package file. It is possible to add monitoring items by editing the source files to customize the loadable modules.

Creation of Loadable modules requires the development environment for Zabbix. Please prepare the environment in advance while referring to the Zabbix official document. In the following example, the environment is prepared in the directory `[ /tmp/zabbix_src ]`

1. Installation of libyaml and libyaml-devel

   Please install the Zabbix plugin package file.

   ```
   # yum -y install libyaml libyaml-devel
   ```
2. Installation of the Zabbix Plugin Package

   Please install the Zabbix plugin package file.
3. Copy of Source Files of Loadable Modules

   Copy the source files of loadable modules into the directory src/modules under the Zabbix development environment as follows:

   ```
   $ cp -r /opt/nec/sysmng-soft/monitoring-sample/zabbix/
   zabbix-3.0.11/usr/src/zabbix/modules/ve /tmp/zabbix_src/
   zabbix-3.0.11/src/modules/
   ```
4. Editing of the Source Files

   Please edit the source files.

   ```
   $ cd /tmp/zabbix_src/zabbix-3.0.11/src/modules/ve
   ```

   ```
   $ vi ve_hw_item.c
   ```
5. Generation of Loadable Modules

   Make the loadable modules as follows:

   ```
   $ make
   ```
6. Copy of the Loadable Modules

Copy the created loadable modules into VHs.

7. Registration of the Loadable Modules with Zabbix Agent

   Please register the loadable modules with Zabbix agent on VHs. Please refer to " *3.1.5.3. Information Gathering with Loadable Modules* on page 46 " for the registration. Please restart Zabbix agent after the registration.

3.1.10. Exclusion of monitoring hosts for updating SX-Aurora TSUBASA software

This clause explains how to exclude monitoring hosts when updating SX-Aurora TSUBASA software.

1. Log in to Zabbix via Web interface as administrator.
2. Click on 'Configuration' and 'Maintenance' of upper part. Display a screen of Maintenance Mode.
3. Click on 'Create maintenance period'. Display a screen of creating maintenance period.
4. Input a maintenance information to 'Maintenance' tab. Select 'No data collection' as Maintenance Type.
5. Click on 'Periods' tab and 'New'. Input the maintenance period.
6. Click on 'Add' of 'Maintenance Period' field, create the period.
7. Click on 'Hosts & Groups' tab, select the target hosts and groups. And click on 'add' of the screen lower part, create the maintenance period.

3.1.11. Inclusion of monitoring hosts updating SX-Aurora TSUBASA software

This clause explains how to include monitoring hosts when updating SX-Aurora TSUBASA software.

1. Log in to Zabbix via Web interface as administrator.
2. Click on 'Configuration' and 'Maintenance' of upper part. Display a screen of Maintenance Mode.
3. Click on the checkbox of maintenance period which was created on " *3.1.10. Exclusion of monitoring hosts for updating SX-Aurora TSUBASA software* on page 50 ", and click on 'Delete'.

3.2. Ganglia and Nagios

Ganglia is an OSS monitoring tool, which enables realtime monitoring of information such as CPU and memory usage on hosts via a web interface.

Nagios is an OSS monitoring tool, which monitors hosts or services on a network and makes notifications.

In the SX-Aurora TSUBASA system, Ganglia monitors VHs and Nagios makes notifications.

Ganglia and Nagios are configured as shown in next figure, where Nagios and Ganglia server (gmetad) run on the management server and Ganglia agents on monitored VHs. If there are many monitored VHs, please organize them into multiple small groups.

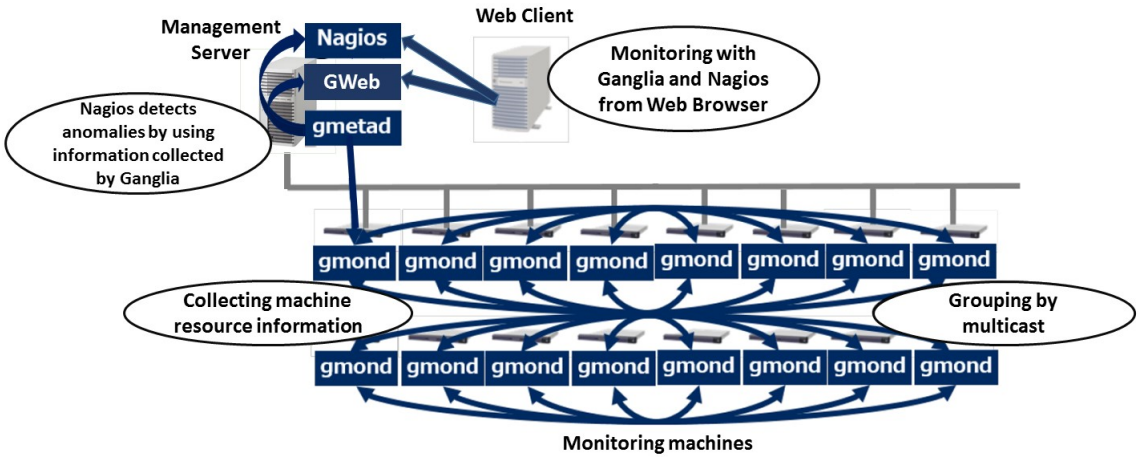

Figure 9: Configuration of Ganglia and Nagios

3.2.1. Preparation

1. Installation of the Sample Plugins

   Download the plugin package files for Ganglia and Nagios, which include configuration and plugin sample files, from the following web site.

   - " *http://jpn.nec.com/hpc/aurora/ve-software/* " or
   - " *http://www.nec.com/en/global/prod/hpc/aurora/ve-software/* "

   And install the package file for Nagios on the management server and for Ganglia on VHs.

```
# rpm -ivh TSUBASA-monitoring-Nagios-X.X-Y.x86_64.rpm
# rpm -ivh TSUBASA-monitoring-Ganglia-X.X-Y.x86_64.rpm
```

   After the installation, configuration and plugin sample files for each validated version of Ganglia and Nagios are placed in /opt/nec/sysmng-soft/monitoring-sample/nagios and /opt/nec/sysmng-soft/monitoring-sample/ganglia, respectively.

Table 6: List of sample files Installation Path

Installation Path	Description
.../ganglia-3.7.2/etc/ganglia/conf.d/ve.pyconf	Ganglia plugin configuration files on VH.
.../ganglia-3.7.2/etc/ganglia/conf.d/veos.pyconf	
.../ganglia-3.7.2usr/lib64/ganglia/python_modules/ve.py	Plugin modules for Ganglia to gather monitoring data of VEs
.../ganglia-3.7.2/usr/lib64/ganglia/python_modules/veos.py	
.../ganglia-3.7.2/usr/lib64/ganglia/python_modules/ve_common.py	
.../ganglia-3.7.2/usr/lib64/ganglia/python_modules/ymlparse.so	
.../nagios-4.0.8/etc/nagios/ve/ganglia-services.cfg	Nagios configuration files
.../nagios-4.0.8/etc/nagios/ve/nodes.cfg	
.../nagios-4.0.8/usr/lib64/nagios/plugins/check_ganglia	Plugin modules for interfacing between Nagios and Ganglia
.../nagios-4.0.8/usr/lib64/nagios/plugins/check_ganglia2	
.../nagios-4.0.8/usr/lib64/nagios/plugins/check_ganglia3	

Please use the sample files corresponding to the versions of Ganglia and Nagios you use.

3.2.2. Configuration of the Management Sever

This clause explains how to install and configure Ganglia server and Nagios on the management server.

3.2.2.1. Preparation on the Management Server

Before the installation of Ganglia server and Nagios, please install and configure necessary packages as follows:

1. Configuration of SELinux

   Modify the value of SELinux in the file `[ /etc/sysconfig/selinux ]` as follows and reboot the management server.

   Before :

   ```
   SELINUX=enforcing
   ```

   After :

   ```
   SELINUX=disabled
   ```

2. Setting of firewall

   If firewall is running, only ssh is allowed in the public zone by default. Please add 80/tcp (HTTP), 8649/tcp (for Ganglia to gather information of VHs), and 8651/tcp (for Nagios plugins to receive data from Ganglia).

   ```
   # firewall-cmd --zone=public --add-port=80/tcp --permanent
   success
   # firewall-cmd --zone=public --add-port=8649/tcp --permanent
   success
   # firewall-cmd --zone=public --add-port=8651/tcp --permanent
   success
   # firewall-cmd --reload
   success
   ```

   If firewall is not needed, please stop it.

3. Configuration of Time Synchronization

   Please configure time synchronization using ntp or chrony, since Ganglia obtains and stores information based on the OS time.

4. Setup of Apache HTTP Server

   Please install the Apache HTTP server package and start httpd to access the Ganglia web interface as follows:

   ```
   # yum -y install httpd
   # systemctl start httpd
   ```

5. Autostart of Httpd

   Enable autostart of httpd on the boot of the management server as follows:

   ```
   # systemctl enable httpd
   ```

6. Installation of a Round Robin Database (RRD)

   Ganglia sever stores monitoring data in a RRD. A RRD overwrites the oldest data with new-coming data using a circular buffer to keep the storage footprint constant over time. Please install the following package.

```
# yum -y install rrdtool
```

3.2.2.2. Installation of Ganglia Server

Install Ganglia sever on the management server as follows:

1. Installation of the Extra Packages for Enterprise Linux (EPEL)

   The Ganglia package is included in the EPEL provided by the Fedora project. Please install the EPEL as follows:

   ```
   # yum -y install epel-release
   ```
2. Installation of libyaml

   Please install the following package.

   ```
   # yum -y install libyaml
   ```
3. Installation of Ganglia Server

   Please install Ganglia sever (gmetad, ganglia-web) as follows:

   ```
   # yum -y install ganglia-3.7.2 ganglia-gmetad-3.7.2 ganglia-
   web-3.7.1
   ```

   Please note that the installation of packages provided by Red Hat Network (RHN) can be required to eliminate dependency among packages. In this case, install Ganglia after the installation of the required packages.

3.2.2.3. Configuration of Ganglia Server

Please update the following two files on the management server:

- gmetad.conf
- ganglia.conf of httpd

1. Update of /etc/ganglia/gmetad.conf

   Please insert data_source lines to specify the IP addresses or hostnames of monitored VHs and their group names, one group per line. The group name which updated in this section also use in section of " *Configuration of Ganglia Agent (gmond)* on page 57 ".The VH hosts also can be management by split multiple groups. The following example specifies a list of monitored hosts VH-host1 and VH-host2, and the group name VHgroup.

   ```
   data_source "my cluster" localhost
   data_source "VHgroup" VH-host1 VH-host2
   ```

   It is recommended to specify multiple VHs from one multicast group, so that failure of one Ganglia agent on a VH will prevent Ganglia server from gathering monitoring data.
2. Update of /etc/httpd/conf.d/ganglia.conf

   Please specify the IP address or hostname of a web client which is allowed to access Ganglia sever. The default configuration allows only the local host to access Ganglia server. With no specifications, all web clients can access Ganglia server. Therefore, explicit specifications are strongly recommended.

   ```
   <Location /ganglia>
     Require local
     Require ip   <The IP address of a web client>
   ```

```
  Require host <The hostname of a web client>
</Location>
```

3.2.2.4. Installation of Nagios

Please install Nagios on the management server as follows:

1. Installation of Nagios

   Install Nagios packages as follows:

   ```
   # yum -y install nagios-4.3.2 nagios-plugins-all-2.2.1 nagios-
   plugins-nrpe-3.1.1
   ```

   Please note that the installation of packages provided by Red Hat Network (RHN) can be required to eliminate dependency among packages. In this case, install Nagios after the installation of the required packages.
2. Setup of the Nagios Web Interface

   It is needed to preparation as follows for using Nagios web interface. Please set a password of the default user nagiosadmin of the Nagios web interface, and then restart httpd as follows:

   ```
   # htpasswd /etc/nagios/passwd nagiosadmin
   New password: <password>
   Re-type new password: <password>
   Updating password for user nagiosadmin
   # systemctl restart httpd
   ```

3.2.2.5. Configuration of Nagios

It is possible for Nagios to access monitoring data of VHs gathered by Ganglia using the Nagios configuration and plugin sample files as follows:

1. Copy of the Sample Files

   Please copy the files ganglia-services.cfg and nodes.cfg, which are included in the Nagios configuration and plugin sample files installed on the management server, into the directory /etc/nagios/ve.

   ```
   # mkdir /etc/nagios/ve
   # cd /opt/nec/sysmng-soft/monitoring-sample/nagios/nagios-4.0.8/
   etc/nagios/ve
   # cp ganglia-services.cfg /etc/nagios/ve/ganglia-services.cfg
   # cp nodes.cfg /etc/nagios/ve/nodes.cfg
   ```

   In addition, copy the plugin modules for interfacing between Nagios and Ganglia into the directory of Nagios plugin modules, which is /usr/lib64/nagios/plugins by default.

   ```
   # cd /opt/nec/sysmng-soft/monitoring-sample/nagios/nagios-4.0.8/
   usr/lib64/nagios/plugins
   # cp check_ganglia check_ganglia2 check_ganglia3 /usr/lib64/nagios/
   plugins
   ```
2. Modification of /etc/nagios/nagios.cfg

   Please insert the following line in the file /etc/nagios/nagios.cfg:

   ```
   cfg_dir=/etc/nagios/ve
   ```
3. Modification of /etc/nagios/ve/nodes.cfg

For Ganglia to use multicast communication for monitoring, please specify the IP addresses of the network interfaces used for monitoring for "address" and the hostnames into which the IP addresses are resolved for "host_name". Please specify all the monitored VHs in the file /etc/nagios/ve/nodes.cfg as follows.

```
define host {
 use vh-server
 host_name VH-host1
 address the IP address of monitored VH(VH-host1)
}

define host {
 use vh-server
 host_name VH-host2
 address the IP address of monitored VH(VH-host2)
}
   :
   :
```

4. Modification of /etc/nagios/ve/ganglia-services.cfg

   Configuration for Nagios to obtain the values of monitoring 2VE items of Ganglia is specified in the file /etc/nagios/ve/ganglia-services.cfg.

   Monitoring items in the Ganglia sample plugins described in " *Configuration of Ganglia Agent (gmond)* on page 57 " are preset. If you change the monitoring items, or monitoring VHs other than 2VE configuration, please also modify the file ganglia-services.cfg.

3.2.2.6. Configuration of Actions

Please configure the notification function of Nagios on detection of failure according to your environment.. Notification by email can be specified as follows:

1. Modification of /etc/nagios/objects/contacts.cfg

   Please specify the destination address for "email" in /etc/nagios/objects/contacts.cfg.

```
email                                   nagios@localhost
```

3.2.2.7. Start of Ganglia Server and Nagios

1. Start of the Services

   Pleases start Ganglia server and Nagios as follows, and then restart httpd for the configuration to take effect.

```
# systemctl start gmetad
# systemctl start nagios
# systemctl restart httpd
```

2. Autostart of Ganglia Sever and Nagios

   Enable autostart of Ganglia server and Nagios on the boot of the management server as follows:

```
# systemctl enable gmetad
# systemctl enable nagios
```

3. Reboot of the Services

After update of the configuration files or plugins of Ganglia server or Nagios, please restart Ganglia server or Nagios as follows:

```
# systemctl restart gmetad
# systemctl restart nagios
```

4. Confirmation of Connection

   Please access the following URSs using a web browser to confirm that the web interfaces of Ganglia server and Nagios are accessible. To access the Nagios web interface, please input the username and password set in " *3.2.2.4. Installation of Nagios* on page 54 ".

   http://the IP address or hostname of the management server/ganglia/

   http://the IP address or hostname of the management server/nagios/

   At this point, the initial setup of Ganglia server and Nagios is completed.

3.2.3. Configuration of Ganglia Agent

This clause explains how to install and configure Ganglia agent.

It can be done on each VH manually or on all VHs at once from the management server using Ansible.

3.2.3.1. Manual Setup

Ganglia Agent can be manually configured on each VH as follows:

Installation

1. Setting of firewall

   If firewall is running, only ssh is allowed in the public zone by default. Please add 8649/udp (for Ganglia agent to send and receive data) and 8649/tcp (for Ganglia agent to send data to Ganglia server).

   ```
   # firewall-cmd --zone=public --add-port=8649/udp --permanent
   success
   # firewall-cmd --zone=public --add-port=8649/tcp --permanent
   success
   # firewall-cmd --reload
   success
   ```

   If firewall is not needed, please stop it.

2. Installation of the Extra Packages for Enterprise Linux (EPEL)

   The Ganglia package is included in the EPEL provided by the Fedora project.

   Install the EPEL as follows:

   ```
   # yum -y install epel-release
   ```

3. Installation of libyaml

   Install the following package:

   ```
   # yum -y install libyaml
   ```

4. Installation of Ganglia Agent and its Python Module

   Install the following packages:

   ```
   # yum -y install ganglia-gmond-3.7.2 ganglia-gmond-python-3.7.2
   ```

Configuration of Ganglia Agent (gmond)

Please modify /etc/ganglia/gmond.conf as follows, to divide monitored VHs into multicast groups and gather information using multicast communication.

1. Configuration of Groups

   Please specify the group name to which VHs belong in the "cluster" block as follows. The group name needs to be the same as that specified in " *3.2.2.3. Configuration of Ganglia Server* on page 53 ".

   ```
   cluster {
           name = "VHgroup"
   }
   ```

   Please specify other configuration values for the groups according to your environment.
2. Configuration of Send Channels

   Please specify the network interface to send monitored data in the "udp_send_channel" block as follows. Please note that multicast needs to be enabled on the network interface.

   ```
   udp_send_channel {
     mcast_if = eth0
   }
   ```

   Please specify other configuration values for send channels according to your environment.
3. Configuration of Receive Channels

   Please specify the network interface to receive monitored data in the "udp_recv_channel" block as follows. Please note that too many monitoring items can lead to starvation of the buffer used for communication between Ganglia server and Ganglia agent. In that case, please reduce the number of monitoring items or increase the UDP receive buffer size of Ganglia.

   ```
   udp_recv_channel {
     mcast_if = eth0
     buffer = 10485760
   }
   ```

   Since the UDP receive buffer size of Ganglia cannot exceed the limit of the UDP buffer size of OS, please increase the limit as necessary. The following example checks the limit of the UDP buffer size of OS and sets it to 15MB.

   ```
   # /sbin/sysctl -n net.core.rmem_max
   212992
   # /sbin/sysctl -w net.core.rmem_max=15000000
   net.core.rmem_max = 15000000
   # /sbin/sysctl -n net.core.rmem_max
   15000000
   ```

   Please specify other configuration values for receive channels according to your environment.

Configuration of Plugins

A monitoring item is called a metric in Ganglia. A metric has attributes such as numerical, string, or constant. The Ganglia package file installed in " *3.2.1. Preparation* on page 51 " includes plugin and configuration sample files to gather values of metrics. Please refer to " *B.1. The List of the Monitoring Items (Ganglia)* on page 88 " for the metrics configured in the plugins.

1. Copy of the Configuration Files

   Please copy the following two configuration files installed on VHs into the directory /etc/ganglia/conf.d/ where configuration files for Ganglia plugins are placed.

- ve.pyconf
- veos.pyconf

```
# cd /opt/nec/sysmng-soft/monitoring-sample/ganglia/
ganglia-3.7.2/etc/ganglia/conf.d/
# cp ve.pyconf /etc/ganglia/conf.d/ve.pyconf
# cp veos.pyconf /etc/ganglia/conf.d/veos.pyconf
```

2. Configuration of Plugins

   Please specify the following four files installed on VHs in the Ganglia plugin configuration file `[ /etc/ganglia/conf.d/modpython.conf ]` as follows:

   - ve.py
   - veos.py
   - ve_common.py
   - ymlparse.so

```
modules {
  module {
    name = "python_module"
    path = "modpython.so"
    params = "/usr/lib64/ganglia/python_modules"
    params = "/opt/nec/sysmng-soft/monitoring-sample/ganglia/
ganglia-3.7.2/usr/lib64/ganglia/python_modules"
  }
}
include ("/etc/ganglia/conf.d/*.pyconf")
```

3. Modification of /etc/ganglia/conf.d/ve.pyconf

   Please change the refresh rate (second) specified in /etc/ganglia/conf.d/ve.pyconf according to your environment as follows. Please note that the values of the refresh rate and collect_every need to be the same.

```
modules {
  module {
     :
    param RefreshRate {
      value = 30
    }
     :
  }
  collection_group {
    collect_every = 30
     :
  }
```

4. Modification of veos.pyconf

   Please also change the refresh rate (second) specified in veos.pyconf as needed like " *3* on page 58 ".

5. Restart of Ganglia Agent

   Please restart Ganglia agent as follows:

```
# systemctl restart gmond
```

Customization of Monitoring Items

Application of the sample plugins enables all the monitoring items described in " *B.1. The List of the Monitoring Items (Ganglia)* on page 88 ". If the number of the monitoring items is too many and

causes too heavy workload on the system, please reduce monitoring items as follows. If you need to add monitoring items, please create the corresponding plugin.

1. Round Robin Database (RRD) files

   Monitoring items which gathered by Ganglia is aggregated on management server, and saved to the directory /var/lib/ganglia/rrds as RRD format.

2. Reduction of Monitoring Items

   Comment out the specifications of monitoring items in the following files in the directory `[ /etc/ganglia/conf.d/ ]`. Please refer to " *B.1. The List of the Monitoring Items (Ganglia)* on page 88 " for each monitoring item.

   - veos.pyconf
   - ve.pyconf

   For example, if you comment out the following three lines in ve.pyconf, the monitoring of 0.9V_AVDD voltage sensor is disabled.

   ```
   #metric {
   #    name_match = "ve_(.+)_avdd_0_90v"
   # }
   ```

   After the reduction of monitoring items, please restart Ganglia agent.

   If disabled monitoring items returning to enable again, please delete comment out and restart Ganglia agent.

3. Addition of Monitoring Items

   The following is how to add monitoring items to the sample plugins.

   - Define procedures to monitor new items in the plugin modules (ve.py, veos.py)
   - Define the monitoring items in the configuration files (ve.pyconf, veos.pyconf)
   - Restart Ganglia agent

3.2.3.2. Setup Using Ansible

It is possible to configure Ganglia agent on multiple VHs at once from the management server using Ansible as follows:

1. Setup of Ansible

   Please follow the instructions in section to use Ansible if they are not done yet.

   a. " *2.2. Initial Setup of the Management Server* on page 16 "
   b. " *2.3. Initial Setup of VHs* on page 20 "
   c. " *2.4. Preparing for using Ansible* on page 22 "

   The following files in `[ /opt/nec/sysmng-soft/etc/ansible ]`, which are installed in " *2.2.7. Installing the template package* on page 18 ", are needed here.

   - Common Variables Definition across all Playbooks: group_vars/all
   - A role to configure Ganglia agent: roles/ganglia-gmond

   VHs to be configured here are specified in the inventory file `[ /opt/nec/sysmng-soft/etc/ansible/vh-hosts ]`, which is created in " *2.2.10. Creating host list file* on page 20 ".

2. Copy of the TSUBASA-monitoring-Ganglia-X.X-X.x86_64.rpm file and Update of the yum Repository

   Please copy the Ganglia package file Aurora-monitoring-Ganglia-X.X-X.x86_64.rpm, which is downloaded in " *3.2.1. Preparation* on page 51 ", into the SX-Aurora TSUBASA yum repository on the management server, and update the repository with createrepo command as follows. Ganglia-gmond role installs the package file on each VH from the yum repository.

```
# cd /var/www/html/repos/TSUBASA
# ls rpms
 :
TSUBASA-monitoring-Ganglia-X.X-X.x86_64.rpm
 :
# createrepo -v -g TSUBASA-groups.xml /var/www/html/repos/TSUBASA
```

3. Settings of the common Playbook variable

   Set the http server (= the management server) and the yum repository directory in `[ group_vars/all ]`. The yum repository directory means the directory under which the repodata exists. And it should be specified by a relative path from the document root directory of the http server.

```
---
httpserver: IP address of the http server
reposdir: repos/TSUBASA
```

4. Update of the Master Playbook vh-set.yml

   Please make file of `[ /opt/nec/sysmng-soft/etc/ansible/vh-set.yml ]`, then add the reference as follows:

```
---
- name: Set VH machine
  hosts: all
  become: yes
  roles:
    - ganglia-gmond
```

   If connections via a proxy server are required to obtain packages on VH machines, add the following setting to configure the proxy server.

```
  environment:
    http_proxy: http://proxy.example.com:8080
```

5. Definitions of Variables for Ganglia-gmond Role

   Please specify the version of ganglia-gmond to be installed in the file roles `[ roles/ganglia-gmond/vars/main.yml ]`

Variable	Value
ganglia_gmond_version	The version of ganglia-gmond to be installed

   Also, define the following variables in the file `[ /etc/ganglia/gmond.conf ]`. Please refer to " *Configuration of Ganglia Agent (gmond)* on page 57 " for details.

Variable	Value
ganglia_gmond_clname	The group name to which VHs belong. This is the value specified for "name" in "cluster" block.
ganglia_gmond_send_mcast_if	The network interface for sending data. This is the value specified for "mcast_if" in "udp_send_channel" block.
ganglia_gmond_recv_mcast_if	The network interface for receiving data. This is the value specified for "mcast_if" in "udp_recv_channel" block.
ganglia_gmond_recv_buffer	The buffer size of the receive channel. To explicitly specify the buffer size, enable this variable. This is the value specified for "buffer" in "udp_recv_channel" block.

Please define the following variable in the file /etc/ganglia/conf.d/ve.pyconf. Please refer to " *Configuration of Plugins* on page 57 " for details.

Variable	Value
ganglia_gmond_ve_rate	The refresh rate (second) of monitoring data. This is the value specified for "value" in "param RefreshRate" block.

Please define the following variable in the file `[ /etc/ganglia/conf.d/veos.pyconf ]`. Please refer to " *Configuration of Plugins* on page 57 " for details.

Variable	Value
ganglia_gmond_veos_rate	Refresh rate (second) of monitoring data. This is the value specified for "value" in "param RefreshRate" block.

6. Execution of the Playbook

   Execution of the Playbook on the management server as the management user admin performs the configurations described in " *3.1.3.1. Manual Setup* on page 39 " , and starts Ganglia agent on each VH.

```
$ cd /opt/nec/sysmng-soft/etc/ansible
$ ansible-playbook vh-set.yml
```

Definition of Variables on Different VHs with different Values

The group definition and group variable functions of Ansible enable to define variables for ganglia-gmond role in the file roles/ganglia-gmond/vars/main.yml with respectively different values on different VHs. For example, to specify respectively different values for the variables ganglia_gmond_send_mcast_if and ganglia_gmond_recv_mcast_if on different VHs, please execute the following steps, in addition to the steps 1-5 described above.

- Deletion of Variables in Ganglia-gmond Role

  Delete the variables ganglia_gmond_send_mcast and ganglia_gmond_recv_mcast_if in the file roles/ganglia-gmond/vars/main.yml .
- Definition of Groups

  Please define groups in the inventory file `[ /opt/nec/sysmng-soft/etc/ansible/vh-hosts ]`. Each variables on the VHs in the same group will have the same value.

```
[group0]  # mcast_if = eth0
vh-name000
vh-name001
vh-name002

[group1]  # mcast_if = eth1
vh-name100
vh-name101
vh-name102
```

- Definition of Group Variables

  Please create the files, each of which has the same name as the group name defined in the inventory file, in the directory group_vars and define the values of the variables ganglia_gmond_send_mcast_if and ganglia_gmond_recv_mcast_if in each file.

  `[ group_vars/group0 ]`

```
---
ganglia_gmond_send_mcast_if: eth0
ganglia_gmond_recv_mcast_if: eth0
```

[group_vars/group1]

```
---
ganglia_gmond_send_mcast_if: eth1
ganglia_gmond_recv_mcast_if: eth1
```

- Execution of the Playbook

  Execution of the Playbook on the management server as the management user admin defines the variables "mcast_if" in "udp_send_channel" block and "udp_recv_channel" block in the file [/etc/ganglia/gmond.conf] on each VH with the values of the corresponding group variables.

```
$ ansible-playbook vh-set.yml
```

Deletion and Addition of Monitoring Items

To delete monitoring items described in " *Customization of Monitoring Items* on page 58 " or add new monitoring items, please modify the following template files:

- [roles/ganglia-gmond/templates/ganglia-3.7.2/ve.pyconf.j2]
- [roles/ganglia-gmond/templates/ganglia-3.7.2/veos.pyconf.j2]

Execution of ansible-playbook command makes the modification take effect on all VHs.

3.2.3.3. Exclude the VH from monitoring

At the Updating SX-Aurora TSUBASA monitoring software, please exclude the VH from monitoring as following:

1. Please check the following setting is done in the /etc/nagios/cgi.cfg. By default, following setting is already done after installation.

   ```
   use_authentication=1
   ```
2. Access to Web I/F of Nagios with Web browser and login as user whom registered at the time of environment setting. Then, disable the notification function with following steps.
3. Click the "Hosts" link from top window. It displays a list of all monitoring target VHs.
4. Click the VH that need to exclude from monitoring.
5. Click "Disable notifications for all services on this host" link.
6. Check to "Disable For Host Too" checkbox, and click "Commit" button.
7. Click "Done" link to complete settings.
8. Back to the window of step 4 and check whether the settings are reflected.
9. Stop the Ganglia monitoring agent service on the VH by using systemctl.

   ```
   # systemctl stop gmond
   ```

3.2.3.4. Restart monitoring of the VH.

After the Updating SX-Aurora TSUBASA monitoring software, please release the excluded VH to monitoring as following:

1. Please check the following setting is done in the /etc/nagios/cgi.cfg. By default, following setting is already done after installation.

   ```
   use_authentication=1
   ```
2. Access to Web I/F of Nagios with Web browser and login as user whom registered at the time of environment setting.
3. Click the "Hosts" link from top window. It displays a list of all monitoring target VHs.

4. Click the VH that need to release to monitoring target.
5. Click "Enable notifications for all services on this host" link.
6. Check to "Enable For Host Too" checkbox, and click "Commit" button.
7. Click "Done" link to complete settings.
8. Back to the window of step 4 and check whether the settings are reflected.
9. Start the Ganglia monitoring agent service on the VH by using systemctl.

```
# systemctl start gmond
```

Chapter 4
Power Control

You can control VH power (on and off) by:

- Powering on and off VHs manually
- Powering on and off VHs from the management server with the Baseboard Management Controller (BMC).
- Powering on and off VHs from the management server with Wake-On-LAN(WOL)

When manually powering on and off VHs, skip this chapter.

When powering on and off VHs from the management server, control power with `powerctrl` and `powerctrl_w`, the tools provided in the template package, TSUBASA-sysmngsoft-X.X-Y.noarch.rpm. The `powerctrl` uses the BMC to control power while the `powerctrl_w` uses the WOL. Only the rootuser can perform these power control tools. Use the `powerctrl` or the `powerctrl_w` according to VHs you use.

4.1. Controlling power with BMC

This section explains how to use the Baseboard Management Controller (BMC) to set up environments for controlling VH power from the management server and to power on and off VHs.

The after explanation premises that the BMC operation user is initialized.

4.1.1. Configuration for controlling power with BMC

4.1.1.1. Installing IPMITool on the management server

Install IPMITool on the management server.

Example:

```
$ su
# yum -y install ipmitool
```

4.1.1.2. Configuration for Power control tool `powerctrl`

`powerctrl` refers the host list file which includes the list of target VH hosts, `[ /opt/nec/sysmng-soft/etc/vh_host.conf ]`. This file has been already created in " *2.2.10. Creating host list file* on page 20 ".

4.1.2. Powering on/off VHs with BMC

Powering on VHs

Run the `on` command to power on the hosts listed in the host list file `[ /opt/nec/sysmng-soft/etc/vh_host.conf ]`.

```
# /opt/nec/sysmng-soft/bin/powerctrl on host001
Total host count is 1.
Do you want to execute "powerctrl on host001"? [y/n] y
BMC user name: ADMIN
```

```
BMC password: ************
send power control command to host001
```

You can specify multiple hosts to be powered on. When sequential numbers are assigned at the end of hostnames, you can use a range of sequential numbers to specify hostnames to control their powers. See " *5.2. powerctrl* on page 68 " for details.

>Powering off VHs

In general, use the configuration management function to shut down and power off VHs. When you need to forcibly power off VHs for any reasons, you can do so from the management server.

Run the `off` command to power off the host machines registered in the host list file, `[ /opt/nec/sysmng-soft/etc/vh_host.conf ]`.

```
# /opt/nec/sysmng-soft/bin/powerctrl off host001
Total host count is 1.
Do you want to execute "powerctrl off host001"? [y/n] y
BMC user name: ADMIN
BMC password: ************
send power control command to host001
```

As with powering on, you can specify multiple hosts to be powered off.

Other functions

You can use `powerctrl` to see the power status of VHs and to perform cold reboot Vhs.

```
# /opt/nec/sysmng-soft/bin/powerctrl status host001
```

```
# /opt/nec/sysmng-soft/bin/powerctrl reset host001
```

4.2. Powering on/off VHs with WOL

This section explains the configuration of the power control function with Wake-On-LAN (WOL) and how to use the tool to control VH power from the management server.

4.2.1. Configuration of controlling power with WOL

4.2.1.1. Check the network card

Check that VH machines' network cards which connect the machines to the management server support Wake-On-LAN (WOL). If network cards do not support the WOL, the functions described in this section are not available.

4.2.1.2. Configuration for the power control tool `powerctrl_w`

`powerctrl_w` refers the host list file which contains the target VH hosts, `[ /opt/nec/sysmng-soft/etc/vh_host_w.conf ]`. This file has been already created in " *2.2.10. Creating host list file* on page 20 ".

4.2.2. Powering on/off VHs with WOL

Powering on VHs

`powerctrl_w` tool can power on the hosts listed in the host list file `[ /opt/nec/sysmng-soft/etc/vh_host_w.conf ]`.

```
# /opt/nec/sysmng-soft/bin/powerctrl_w host001
Total host count is 1.
Do you want to execute "powerctrl_w host001"? [y/n] y
send power control command to host001
```

You can specify multiple hosts to be powered on. When sequential numbers are assigned at the end of hostnames, you can use a range of sequential numbers to specify hostnames to control their powers. See " *5.3. powerctrl_w* on page 70 " for details.

Powering off VHs

`powerctrl_w` cannot power off VHs. Use the configuration management function to shut down and power off VHs.

Chapter 5 Reference

5.1. create-hostlist.py

NAME

create-hostlist.py -- Tool for converting VH information files at once

SYNOPSIS

create-hostlist.py --type=*TYPE* [--infile=*INPUTFILE*] [--outfile=*OUTPUTFILE*]

DESCRIPTION

`create-hostlist.py` creates host list files using the VH information file `[ /opt/nec/sysmng-soft/etc/host-list.xlsx ]`. Specify types of host list files to be created in *TYPE* of the --type option. The types of host list files you can specify in *TYPE* are:

ansible	The tool creates the host list file in the inventory file format defining hosts to be managed when Ansible is used to manage the VH configuration.
cobbler	The tool creates the host list file for the cobimport.sh tool, which registers system records at once when Cobbler is used to install the OS on VHs.
power_bmc	The tool creates the host list file defining those controlled by the powerctrl tool when BMC is used to control power.
power_wol	The tool creates the host list file defining those controlled by the powerctrl_w tool when WOL is used to control power.

A created file is displayed on the console window as standard output by default. If the --outfile option is specified, a file is created as the file specified in *OUTPUTFILE*. If the specified file already exists, file creation fails.

Similarly, a VH information file that is used as an input source can be changed to the file specified in *INPUTFILE* of the --infile option.

VH information file

The following data is included in the VH information file.

hostname	Host names of VHs
host-group	Group names when VHs are grouping
IP-address	IP addresses of VHs
IP-address(BMC)	BMC IP address of VHs
MAC-address	MAC address (AA:BB:CC:DD:EE:FF style) of VHs
netmask	Netmask of the VH IP address
gateway	Gateway's address

The following data are used for each *TYPE*.

	hostname	host-group	IP-address	IP-address (BMC)	MAC-address	netmask	gateway
ansible	*	*					
cobbler	*		*		*	*	*
power_bmc	*			*			
power_wol	*				*		

The VH information file is provided as a sample file. Do not change the file type of the sample file, and leave unused cells blank.

RETURN VALUE

Zero is returned for a successful host list file creation and any number other than zero is returned for a failed creation.

FILES

/opt/nec/sysmng-soft/etc/host-list.xlsx VH information file

NOTE

`create-hostlist.py` uses Python. Make sure that Python 2.7 or higher version is installed. And python-excel library is required. Also make sure xlrd package is installed.

SEE ALSO

" *powerctrl* ", " *powerctrl_w* ", " *setup-hostlist* "

5.2. powerctrl

NAME

powerctrl -- Power control tool for VH (with BMC)

SYNOPSIS

powerctrl [*options*] *command target*

powerctrl { --all | -a } *command*

powerctrl { --help | -h }

DESCRIPTION

The `powerctrl` uses the ipmitool commands to control power of the hosts specified in *target*. It performs power control operations such as booting, shutting down, and rebooting hosts according to a value specified in *command*. Only the rootuser can run this tool.

Behavior of the `powerctrl` command and information displayed after the execution comply with the `chassis power` sub-commands of the `ipmitool`.

OPTIONS

--user=*user*
-u *user* Specify the BMC management username.

--password=*password* Specify the password of the BMC management user.

-p *password*	When the -user (or -u) or -password (or -p) options are omitted, the interactive mode prompts you for your next input. Enter the username and password of the BMC management user in the interactive mode.
	`BMC user name:`
	`BMC password:`

--yes -y	The `powerctrl` displays the entered command and the number of target hosts before running the command, and asks users whether or not to run the command. When this option is specified, "y" is automatically entered. When you omit this option, if there is no problem, enter "y". Entering "n" cancel running the command.
--all -a	A power control operation specified in *command* runs on all the hosts listed in the host information file `[ vh_host.conf ]`. When this option is specified, you do not need to specify *target*.
--toolpath=*ipmitool_path* -t *ipmitool_path*	Specify the absolute path of the `ipmitool` commands. If it is not specified, `[ /usr/bin/ipmitool ]` is used.
--help -h	Help message is displayed.
--file=*log_path* -f *log_path*	Specify the file where execution results are written. When this option is omitted, results are displayed on the console window as standard output. The `powerctrl` displays target hostnames and operation results as execution results.

command

status	Displays power status of target hosts.
on	Powers on target hosts.
off	Powers off target hosts.
reset	Cold reboot is performed on target hosts.

Specifying host(*target*)

Specify the names of the hosts targeted for power control. The hostnames specified here need to be registered in the host information file `[ vh_host.conf ]`.

When hostnames end with an integer greater than zero, you can enter a range of integers to specify hosts to be operated.

See below to specify hostnames.

(1)	Specifying a single host	host0
(2)	Specifying multiple hosts	host0 host1

(3)	Specifying a range of hosts	host[0-100]
(4)	Combining (2) and (3)	host[0-100] host200 host300

Example

```
# /opt/nec/sysmng-soft/bin/powerctrl on host00[1-8]
Total host count is 8.
Do you want to execute "powerctrl on host00[1-8]" ? [y/n] y
BMC user name: ADMIN
BMC password: ************
send power control command to host001
send power control command to host002
send power control command to host003
send power control command to host004
send power control command to host005
send power control command to host006
send power control command to host007
send power control command to host008
```

NOTE

Use the same BMC administrative user and password on all hosts.

FILES

[/opt/nec/sysmng-soft/etc/vh_host.conf]	Host information file that defines the relation between host name and BMC IP address of VHs.

SEE ALSO

ipmitool, " *powerctrl_w* ", " *vh_host.conf* "

5.3. powerctrl_w

NAME

powerctrl_w -- Power control tool for VH (with WOL)

SYNOPSIS

powerctrl_w [*options*] *target*

powerctrl_w { --all | -a }

powerctrl_w { --help | -h }

DESCRIPTION

The `powerctrl_w` uses the `ether-wake` commands to control power (power on) of the hosts specified in *target*. Only the rootuser can run this tool.

Behavior of the `powerctrl_w` command and information displayed after the execution comply with the `ether-wake` commands.

OPTIONS

--yes -y	The `powerctrl_w` displays the entered command and the number of target hosts before running the command, and asks users whether or not to run the command.

When this option is specified, "y" is automatically entered.

When you omit this option, if there is no problem, enter "y". Entering "n" cancel running the command.

--all
-a

Power-on is performed on all the hosts listed in the host information file `[ vh_host_w.conf ]`. When this option is specified, you do not need to specify *target*.

--help
-h

Help message is displayed.

--file=*log_path*
-f *log_path*

Specify the file where execution results are written. When this option is omitted, results are displayed on the console window as standard output.

The `powerctrl_w` displays target hostnames and operation results as execution results.

Specifying host(*target*)

Specify the names of the hosts targeted for power control. The hostnames specified here need to be registered in the host information file `[ vh_host_w.conf ]`.

When hostnames end with an integer greater than zero, you can enter a range of integers to specify hosts to be operated.

See below to specify hostnames.

(1)	Specifying a single host	host0
(2)	Specifying multiple hosts	host0 host1
(3)	Specifying a range of hosts	host[0-100]
(4)	Combining (2) and (3)	host[0-100] host200 host300

Example

```
# /opt/nec/sysmng-soft/bin/powerctrl_w host00[1-2]
Total host count is 2.
Do you want to execute "powerctrl_w host00[1-2]" ? [y/n] y
send power control command to host001
send power control command to host002
```

FILES

[/opt/nec/sysmng-soft/etc/vh_host_w.conf]	Host information file that defines the relation between host name and MAC address of VHs.

SEE ALSO

" *powerctrl* ", " *vh_host_w.conf* "

5.4. setup-hostlist

NAME

setup-hostlist -- Tool for converting VH information files at once

SYNOPSIS

setup-hostlist

DESCRIPTION

`setup-hostlist` executes `create-hostlist.py` to creates all host list files using the VH information file `[ /opt/nec/sysmng-soft/etc/host-list.xlsx ]`.

File types and paths of created files are as follows:

`[ /opt/nec/sysmng-soft/etc/ansible/vh-hosts ]`

Host list file in the inventory file format defining hosts to be managed when Ansible is used to manage the VH configuration

`[ /opt/nec/sysmng-soft/etc/cobimport.txt ]`

Host list file for the cobimport.sh tool, which registers system records at once when Cobbler is used to install the OS on VHs (Not used in this guide.)

`[ /opt/nec/sysmng-soft/etc/vh_host.conf ]`

Host list file defining those controlled by the powerctrl tool when BMC is used to control power

`[ /opt/nec/sysmng-soft/etc/vh_host_w.conf ]`

Host list file defining those controlled by the powerctrl_w tool when WOL is used to control power

For the information of data to register in VH information file, see " *5.1. create-hostlist.py* on page 67 ".

RETURN VALUE

Zero is returned for a successful host list file creation and one is returned for a failed creation.

FILES

/opt/nec/sysmng-soft/etc/host-list.xlsx VH information file

SEE ALSO

" *create-hostlist.py* ", " *powerctrl* ", " *powerctrl_w* "

5.5. vh_host.conf

NAME

vh_host.conf -- Host information file for powerctrl tool

SYNOPSIS

`[ /opt/nec/sysmng-soft/etc/vh_host.conf ]`

DESCRIPTION

`[ vh_host.conf ]` is the file where VHs controlled by the power control tool, powerctrl, using BMC are registered. This file is created by `create-hostlist.py` tool.

Register one host in one line in the following format.

```
    Hostname  BMC IP address
```

Separate entry fields with space(s) or tab(s) and write VH hostnames in the first field and BMC IP addresses of VHs in the second field.

(Example)

```
host001  192.168.0.1
host002  192.168.0.2
host003  192.168.0.3
host004  192.168.0.4
```

In the above format, strings between the character "#" and the end of a row are treated as comments and are ignored.

FILES

`[ /opt/nec/sysmng-soft/etc/vh_host.conf ]`	Host information file where VH hostnames and their BMC IP addresses are listed

SEE ALSO

" *create-hostlist.py* ", " *powerctrl* "

5.6. vh_host_w.conf

NAME

vh_host_w.conf -- Host information file for powerctrl_w tool

SYNOPSIS

`[ /opt/nec/sysmng-soft/etc/vh_host_w.conf ]`

DESCRIPTION

`[ vh_host_w.conf ]` is the file where VHs controlled by the power control tool `powerctrl_w` using WOL are registered. This file is created by `create-hostlist.py` tool.

Register one host in one line in the following format.

```
    Hostname        MAC address
```

Separate entry fields with space(s) or tab(s) and write VH hostnames in the first field and MAC addresses of VHs' network card in the second field.

(Example)

```
host001  00:13:D3:4B:83:72
host002  00:13:D3:4B:83:96
host003  00:13:D3:4B:84:94
host004  00:13:D3:4B:84:62
```

FILES

`[ /opt/nec/sysmng-soft/etc/vh_host_w.conf ]`	Host information file where VH hostnames and their MAC addresses are listed

SEE ALSO

" *create-hostlist.py* ", " *powerctrl_w* "

Appendix A Monitoring Items of Zabbix

A.1. The Item Keys Provided by the Loadable Modules

The following tables show the item keys provided by the Zabbix loadable modules. An item key can take parameters which are passed to the loadable modules. The loadable modules obtain information about VEs or cores specified with the parameters.

Table 7: The Item Keys Provided by ve_hw_item.so

Item Key Name				
	Meaning	**Attribute**	**Parameter**	**Remarks**
vehw.attached_ves				
	The number of VEs	Numerical Value (Integer)	None	Return an integer in decimal
vehw.mmm_version				
	vehw.mmm_version	String	None	An example of the return value: 01.000
vehw.ve_state				
	VE status	String	ve - VE number	The return value is "0,"1","2","3", or "4". The meanings of the values are as follows: • 0:UNINITIALIZED • 1:ONLINE • 2:OFFLINE • 3:MAINTENANCE • 4:UNAVAILABLE
vehw.ve_model[ve]				
	VE model	String	ve - VE number	An example of the return value: A1
vehw.product_type[ve]				
	Product type	String	ve - VE number	An example of the return value: 00
vehw.cores[ve]				
	The number of available cores on a VE	Numerical value (integer)	ve - VE number	Return the number of cores in decimal.
vehw.ve_chip_sn[ve]				

Item Key Name				
	Meaning	**Attribute**	**Parameter**	**Remarks**
	The serial number of a VE chip	String	ve - VE number	An example of the return value: 0x0123456789ABCDEF
vehw.ve_board_sn[ve]				
	The serial number of a VE card	String	ve - VE number	An example of the return value: 0x0123456789ABCDEF
vehw.ve_driver_version[ve]				
	VE driver version	String	ve - VE number	An example of the return value: 01.00
vehw.vmcfw_version[ve]				
	VMCFW version	String	ve - VE number	An example of the return value: 01.00
vehw.core_clock[ve]				
	Core clock frequency	Numerical value (integer)	ve - VE number	The unit of the return value is Hz. An example of the return value:1600000000
vehw.base_clock[ve]				
	Base clock frequency	Numerical value (integer)	ve - VE number	The unit of the return value is Hz. An example of the return value:1000000000
vehw.memory_clock[ve]				
	Memory clock frequency	Numerical value (integer)	ve - VE number	The unit of the return value is Hz. An example of the return value:1000000000
vehw.memory_size[ve]				
	Total memory size	Numerical value (integer)	ve - VE number	The unit of the return value is GByte. An example of the return value:48
vehw.l1_icache_size[ve]				
	L1 I cache size per core	Numerical value (floating point number)	ve - VE number	The unit of the return value is Byte. An example of the return value:32000
vehw.l1_dcache_size[ve]				
	L1 D cache size per core	Numerical value (integer)	ve - VE number	The unit of the return value is Byte. An example of the return value:32000
vehw.l2_cache_size[ve]				

Item Key Name				
	Meaning	**Attribute**	**Parameter**	**Remarks**
	L2 cache size per core	Numerical value (integer)	ve - VE number	The unit of the return value is Byte. An example of the return value:256000
vehw.l3_cache_size[ve]				
	LLC size of a VE chip	Numerical value (integer)	ve - VE number	The unit of the return value is Byte. An example of the return value:16000000
vehw.partitioning_mode[ve]				
	Partitioning mode	String	ve - VE number	The return value is "0" or "1". The meanings of the values are as follows: • 0:Partitioning mode is disabled. • 1:Partitioning mode is enabled.
vehw.throttling_level[ve]				
	Throttling level	Numerical value (integer)	ve - VE number	An example of the return value:2
vehw.pci.bus_id[ve]				
	PCI bus ID	String	ve - VE number	An example of the return value: 0000:22:00.0
vehw.pci.vendor_id[ve]				
	PCI vendor ID	String	ve - VE number	An example of the return value: 0x001C
vehw.pci.device_id[ve]				
	PCI device ID	String	ve - VE number	An example of the return value:0x1BCF
vehw.pci.class_code[ve]				
	PCI class code	String	ve - VE number	An example of the return value:0x0B4000
vehw.pci.current_link_speed[ve]				
	The current link speed	Numerical value (floating point number)	ve - VE number	The unit of the return value is GT/s.An example of the return value:8.0
vehw.pci.max_link_speed[ve]				
	The maximum link speed	Numerical value (floating point number)	ve - VE number	The unit of the return value is GT/s.An example of the return value:8.0

Item Key Name				
	Meaning	**Attribute**	**Parameter**	**Remarks**
vehw.pci.negotiated_link_width[ve]				
	The current link width	String	ve - VE number	An example of the return value:x16
vehw.pci.maximum_link_width[ve]				
	The maximum link width	String	ve - VE number	An example of the return value:x16
vehw.temperature.chip_core[ve,core]				
	VE core temperature	Numerical value (floating point number)	ve - VE number core - Core number (*1)	The unit of the return value is degrees Celsius. An example of the return value: 30.125
vehw.temperature.diode0[ve]				
	VE CPU temperature(core 8,9 side)	Numerical value (floating point number)	ve - VE number	The unit of the return value is degrees Celsius. An example of the return value: 33.1
vehw.temperature.diode1[ve]				
	VE CPU temperature (core 0,1 side)	Numerical value (floating point number)	ve - VE number	The unit of the return value is degrees Celsius. An example of the return value:34.25
vehw.temperature.aux_power_side[ve]				
	Power connector side temperature (intake)	Numerical value (floating point number)	ve - VE number	The unit of the return value is degrees Celsius. An example of the return value:30.25
vehw.temperature.bracket_side[ve]				
	LED bracket side temperature(exhaust)	Numerical value (floating point number)	ve - VE number	The unit of the return value is degrees Celsius. An example of the return value: 31.5
vehw.temperature.adt7462[ve]				
	Board temperature	Numerical value (floating point number)	ve - VE number	The unit of the return value is degrees Celsius. An example of the return value:32.75
vehw.fan_speed[ve]				
	Fan rotation number	Numerical value (floating point number)	ve - VE number	The unit of the return value is rpm. An example of the return value:7700
vehw.voltage.core_vdd_0[ve]				

Item Key Name				
	Meaning	**Attribute**	**Parameter**	**Remarks**
	VE internal voltage 0	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:0.903
vehw.voltage.core_vdd_1[ve]				
	VE internal voltage 1	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:0.904
vehw.voltage.ve_vdd[ve]				
	0.89V_VDD voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:0.891
vehw.voltage.ve_avdd[ve]				
	0.90V_AVDD voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:0.900
vehw.voltage.hbm_e_vddc_1_2v[ve]				
	1.2V_HBM_ E_VDDC voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:1.199
vehw.voltage.hbm_e_vddq_1_2v[ve]				
	1.2V_HBM_ E_VDDQ voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:1.201
vehw.voltage.hbm_w_vddc_1_2v[ve]				
	1.2V_HBM_ W_VDDC voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:1.2
vehw.voltage.hbm_w_vddq_1_2v[ve]				
	1.2V_HBM_ W_VDDQ voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:1.206
vehw.voltage.aux_12v[ve]				
	The auxiliary power unit 12V voltage 0	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:11.9375
vehw.voltage.edge_12v[ve]				
	Edge 12V voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:12.0625
vehw.voltage.edge_3_3v[ve]				

Item Key Name				
	Meaning	**Attribute**	**Parameter**	**Remarks**
	Edge 3.3V voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:3.3024
vehw.voltage.vpp_2_5v[ve]				
	2.5V_VPP voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:2.509
vehw.current.aux_12v[ve]				
	The auxiliary power unit 12V current 0	Numerical value (floating point number)	ve - VE number	The unit of the return value is A. An example of the return value:1.211
vehw.current.edge_12v[ve]				
	Edge 12V current	Numerical value (floating point number)	ve - VE number	The unit of the return value is A. An example of the return value:1.213
vehw.voltage.pll_0_89v[ve]				
	0.89V_PLL voltage	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:0.891
vehw.voltage.vddh_1_8v[ve]				
	1.8V_VDDH	Numerical value (floating point number)	ve - VE number	The unit of the return value is V.An example of the return value:1.8
vehw.eccs.mcu_correctable_error[ve]				
	ECC information MCU 1Bit error	String	ve - VE number	An example of the return value:2017/05/30 18:04:30
vehw.eccs.llc_correctable_error[ve]				
	ECC information LLC 1Bit error	String	ve - VE number	An example of the return value:2017/05/30 18:04:30
vehw.eccs.core_dp_correctable_error[ve]				
	ECC information COREDP 1Bit error	String	ve - VE number	An example of the return value:2017/05/30 18:04:30
vehw.eccs.comn_dp_correctable_error[ve]				
	ECC information COMN DP 1Bit error	String	ve - VE number	An example of the return value:2017/05/30 18:04:30
vehw.eccs.vr_correctable_error[ve]				
	ECC information VR 1Bit error	String	ve - VE number	An example of the return value:2017/05/30 18:04:30
vehw.errs.fault_occur_time[ve]				

Item Key Name				
	Meaning	**Attribute**	**Parameter**	**Remarks**
	ERR information The occurrence time of a severe failure	String	ve - VE number	An example of the return value:2017/05/30 18:04:30
vehw.errs.fault_errors[ve]				
	ERR Information The kind of a severe failure	String	ve - VE number	An example of the return value:Linkdown Error
vehw.errs.non_fault_occur_time[ve]				
	ERR Information The occurrence time of a slight failure	String	ve - VE number	An example of the return value:2017/05/30 18:04:30
vehw.errs.non_fault_errors[ve]				
	ERR Information The kind of a slight failure	String	ve - VE number	An example of the return value:NONE

Table 8: The List of Keys Provided by ve_os_item.so

Key				
	Meaning	**Attribute**	**Parameter**	**Remarks**
ve_os.state[ve]				
	The status of OS	String	ve - VE number	The return value is "0","1","2", or "3". The meanings of the values are as follows: • 0:ONLINE • 1:OFFLINE • 2:INITIALIZING • 3:TERMINATING
ve_vm.memory.size[ve,mode]				
	Memory usage	Numerical value (integer)	ve - VE number mode - total, free, buffers, cached, available, or pavailable	The unit of the return value is Byte
ve_system.ve.load[ve,cpu,mode]				
	VE load	Numerical value (floating point number)	ve - VE number cpu - all, percore mode - avg1, avg5, or avg15	The return value is the number of processes.
ve_system.ve.util[ve,core,type]				

Key				
	Meaning	**Attribute**	**Parameter**	**Remarks**
	VE usage	Numerical value (floating point number)	ve - VE number core - Core number type - idle, nice, user, system, interrupt, or softirq	The unit of the return value is %.

A.2. The Items and Triggers Provided by the Templates

The Zabbix plugin package file includes the templates which use item keys provided by the Zabbix loadable modules. The format of an item name is an item key name followed by a VE number in square brackets as follows. Please refer to " *A.1. The Item Keys Provided by the Loadable Modules* on page 75 " for the item names.

VE Number	Item Key Name	Item Name
0	vehw.attached_ves	vehw.attached_ves[0]
1	vehw.attached_ves	vehw.attached_ves[1]

The following Tables show the item names of VE number 0 defined in template_ve0.xml. Some items include VEOS or VEHW defined as Application. The other template of VE is different in the VE number of the item name. Some items of VEHW have triggers.

Table 9: The List of the Items of the Application Name(VEHW)

Item Name				Meaning
	Trigger Name	**Level**	**Trigger Condition**	
vehw.attached_ves				The number of VEs
vehw.base_clock[0]				Base clock frequency
vehw.cores[0]				The number of available cores on a VE LSI
vehw.core_clock[0]				Core clock frequency
vehw.current.aux_12v[0]				Auxiliary power unit 12V current
	ve0_current.aux_12v_warn	Warning	20A or over	
	ve0_current.aux_12v_error	Severe Failure	25A or over	
vehw.current.edge_12v[0]				Edge 12V current
	ve0_current.edge_12v_warn	Warning	5A or over	
	ve0_current.edge_12v_error	Severe Failure	5.5A or over	
vehw.fan_speed[0]				Fan rotation number
vehw.l1_dcache_size[0]				L1 D cache size per core

Item Name				Meaning
	Trigger Name	Level	Trigger Condition	
vehw.l1_icache_size[0]				L1 I cache size per core
vehw.l2_cache_size[0]				L2 cache size per core
vehw.l3_cache_size[0]				LLC cache size per core
vehw.memory_clock[0]				Memory clock frequency
vehw.memory_size[0]				Total memory size
vehw.mmm_version				MMM version
vehw.partitioning_mode[0]				Partitioning mode
vehw.pci.bus_id[0]				PCI bus ID
vehw.pci.class_code[0]				PCI class code
vehw.pci.current_link_speed[0]				The current link speed
vehw.pci.device_id[0]				PCI device ID
vehw.pci.maximum_link_width[0]				The maximum link width
vehw.pci.max_link_speed[0]				The maximum link speed
vehw.pci.negotiated_link_width[0]				The current link width
vehw.pci.vendor_id[0]				PCI vendor ID
vehw.product_type[0]				Product type
vehw.temperature.adt7462[0]				Board temperature
	ve0_adt7462_warn	Warning	85 degree or over	
	ve0_adt7462_error	Severe Failure	93 degree or over	
vehw.temperature.aux_power_side[0]				Power connector side temperature (intake)
	ve0_aux_power_side_warn	Warning	70 degree or over	
	ve0_aux_power_side_error	Severe Failure	77 degree or over	
vehw.temperature.bracket_side[0]				LED bracket side temperature (exhaust)
	ve0_bracket_side_warn	Warning	70 degree or over	
	ve0_bracket_side_error	Severe Failure	77 degree or over	
vehw.temperature.chip_core0[0]				VE Core0 temperature
	ve0_chip_core0_temp_warn	Warning	85 degree or over	
	ve0_chip_core0_temp_error	Severe Failure	93 degree or over	
vehw.temperature.chip_core1[0]				VE Core1 temperature

Item Name			Meaning
	Trigger Name	**Level**	**Trigger Condition**
	ve0_chip_core1_temp_warn	Warning	85 degree or over
	ve0_chip_core1_temp_error	Severe Failure	93 degree or over
vehw.temperature.chip_core2[0]			VE Core2 temperature
	ve0_chip_core2_temp_warn	Warning	85 degree or over
	ve0_chip_core2_temp_error	Severe Failure	93 degree or over
vehw.temperature.chip_core3[0]			VE Core3 temperature
	ve0_chip_core3_temp_warn	Warning	85 degree or over
	ve0_chip_core3_temp_error	Severe Failure	93 degree or over
vehw.temperature.chip_core4[0]			VE Core4 temperature
	ve0_chip_core4_temp_warn	Warning	85 degree or over
	ve0_chip_core4_temp_error	Severe Failure	93 degree or over
vehw.temperature.chip_core5[0]			VE Core5 temperature
	ve0_chip_core5_temp_warn	Warning	85 degree or over
	ve0_chip_core5_temp_error	Severe Failure	93 degree or over
vehw.temperature.chip_core6[0]			VE Core6 temperature
	ve0_chip_core6_temp_warn	Warning	85 degree or over
	ve0_chip_core6_temp_error	Severe Failure	93 degree or over
vehw.temperature.chip_core7[0]			VE Core7 temperature
	ve0_chip_core7_temp_warn	Warning	85 degree or over
	ve0_chip_core7_temp_error	Severe Failure	93 degree or over
vehw.temperature.chip_core8[0]			VE Core8 temperature
	ve0_chip_core8_temp_warn	Warning	85 degree or over
	ve0_chip_core8_temp_error	Severe Failure	93 degree or over
vehw.temperature.chip_core9[0]			VE Core9 temperature
	ve0_chip_core9_temp_warn	Warning	85 degree or over
	ve0_chip_core9_temp_error	Severe Failure	93 degree or over
vehw.temperature.diode0[0]			VE CPU temperature Core8,9 side
	ve0_chip_diode0_warn	Warning	85 degree or over
	ve0_chip_diode0_error	Severe Failure	93 degree or over
vehw.temperature.diode1[0]			VE CPU temperature Core0,1 side
	ve0_chip_diode1_warn	Warning	85 degree or over
	ve0_chip_diode1_error	Severe Failure	93 degree or over

Item Name			Meaning
	Trigger Name	**Level**	**Trigger Condition**
vehw.throttling_level[0]			Throttling level
vehw.ve_board_sn[0]			The serial number of a VE card
vehw.ve_chip_sn[0]			The serial number of a VE chip
vehw.ve_model[0]			VE model
vehw.ve_state[0]			VE status
vehw.vmcfw_version[0]			VMCFW version
vehw.voltage.aux_12v[0]			Auxiliary power unit 12V voltage
	ve0_aux_12v_error	Severe Failure	Below 10.8V or above 13.2V
vehw.voltage.core_vdd_0[0]			VE internal voltage 0
	ve0_core_vdd_0_error	Severe Failure	Below 0.88V or above 1.03V
vehw.voltage.core_vdd_1[0]			VE internal voltage 1
	ve0_core_vdd_1_error	Severe Failure	Below 0.88V or above 1.03V
vehw.voltage.edge_3_3v[0]			Edge 3.3V voltage
	ve0_edge_3_3v_error	Severe Failure	Below 2.97V or above 3.63V
vehw.voltage.edge_12v[0]			Edge 12V voltage
	ve0_edge_12v_error	Severe Failure	Below 10.8V or above 13.2V
vehw.voltage.hbm_e_vddc_1_2v[0]			1.2V_HBM_E_VDDC voltage
	ve0_hbm_e_vddc_1_2v_error	Severe Failure	Below 1.08V or above 1.32V
vehw.voltage.hbm_e_vddq_1_2v[0]			1.2V_HBM_E_VDDQ voltage
	ve0_hbm_e_vddq_1_2v_error	Severe Failure	Below 1.08V or above 1.32V
vehw.voltage.hbm_w_vddc_1_2v[0]			1.2V_HBM_W_VDDC voltage
	ve0_hbm_w_vddc_1_2v_error	Severe Failure	Below 1.08V or above 1.32V
vehw.voltage.hbm_w_vddq_1_2v[0]			1.2V_HBM_W_VDDQ voltage
	ve0_hbm_w_vddq_1_2v_error	Severe Failure	Below 1.08V or above 1.32V
vehw.voltage.pll_0_89v[0]			0.89V PLL voltage
	ve0_pll_0_89v_error	Severe Failure	Below 0.801V or above 0.979V
vehw.voltage.vddh_1_8v[0]			1.8V_VDDH
	ve0_vddh_1_8v_error	Severe Failure	Below 1.62V or above 1.98V

Item Name			Meaning
	Trigger Name	**Level**	**Trigger Condition**
vehw.voltage.ve_avdd[0]			0.90V_AVDD voltage
	ve0_ve_avdd_error	Severe Failure	Below 1.08V or above 1.32V
vehw.voltage.vpp_2_5v[0]			2.5V_VPP voltage
	ve0_vpp_2_5v_error	Severe Failure	Below 2.25V or above 2.75V
vehw.eccs.mcu_correctable_error[0]			ECC information: MCU 1Bit error
vehw.eccs.llc_correctable_error[0]			ECC information: LLC 1Bit error
vehw.eccs.core_dp_correctable_error[0]			ECC information: CORE DP 1Bit error
vehw.eccs.comn_dp_correctable_error[0]			ECC information: COMN DP 1Bit error
vehw.eccs.vr_correctable_error[0]			ECC information: VR 1Bit error
vehw.errs.fault_occur_time[0]			ERR information: The occurrence time of a severe failure
vehw.errs.fault_errors[0]			ERR information: The kind of a severe failure
vehw.errs.non_fault_occur_time[0]			ERR information: The occurrence time of a slight failure
vehw.errs.non_fault_errors[0]			ERR information: The kind of a slight failure

Table 10: The List of the Application Name (VEOS)

Item Name	Meaning
ve_os.state[0]	Monitor the VEOS status
ve_system.ve.load[0,all,avg1]	Monitor the VE load average over the last one minute
ve_system.ve.load[0,all,avg5]	Monitor the VE load average over the last five minutes
ve_system.ve.load[0,all,avg15]	Monitor the VE load average over the last 15 minutes
ve_system.ve.load[0,percore,avg1]	Monitor the core load average over the last one minute
ve_system.ve.load[0,percore,avg5]	Monitor the core load average over the last five minutes

Item Name	Meaning
ve_system.ve.load[0,percore,avg15]	Monitor the core load average over the last 15 minutes
ve_system.ve.util[0,0,idle]	Monitor the core 0 load (idle)
ve_system.ve.util[0,0,interrupt]	Monitor the core 0 load (interrupt)
ve_system.ve.util[0,0,nice]	Monitor the core 0 load (nice)
ve_system.ve.util[0,0,softirq]	Monitor the core 0 load (softirq)
ve_system.ve.util[0,0,system]	Monitor the core 0 load (system)
ve_system.ve.util[0,0,user]	Monitor the core 0 load (user)
ve_vm.memory.size[0,available]	Monitor the VE memory size (available)
ve_vm.memory.size[0,buffers]	Monitor the VE memory size (buffers)
ve_vm.memory.size[0,cached]	Monitor the VE memory size (cached)
ve_vm.memory.size[0,free]	Monitor the VE memory size (free)
ve_vm.memory.size[0,pavailable]	Monitor the VE memory size (utilization)
ve_vm.memory.size[0,total]	Monitor the VE memory size

The following Table is the items defined in template_veos_service.xml. Some items include in The application named VEOS-SERVICE.

Table 11: The List of the Items of the Application Name (VEOS-SERVICE)

Item Name	Meaning
mmm.service	Monitor the MMM services.
mmm.messages	Monitor the logs of MMM. Only the logs with the priority EER are gathered.
ived.service	Monitor the IVED services.
ived.messages	Monitor the logs of IVED. Only the logs with the priority fatal, alert, or crit are gathered.
vemmd.service	Monitor the VEMMD services.
vemmd.messages	Monitor the logs of VEMMD. Only the logs with the priority fatal, alert, or crit are gathered.
veos.messages	Monitor the logs of VEOS. Only the logs with the priority fatal, alert, or crit are gathered.

Appendix B Monitoring Items of Ganglia

B.1. The List of the Monitoring Items (Ganglia)

Following tables are the metrics (monitoring items) provided by the Ganglia plugin modules.The metric names include a VE number and core number as shown in the examples below. Ganglia agent (gmond) configures the metrics according to the recognized number of VEs and cores.

Only variable values of integer or floating point number are displayed in graphs by Ganglia. Fixed values and strings are displayed by pressing "Host Overview" button.

The following is the correspondence between the metrics in the configuration files (*.pyconf) and the metric names automatically configured.

Metric			
	VE number	**Core number**	**Metric Name**
ve_(.+)_cores			
	0	-	ve_0_cores
	1	-	ve_1_cores
ve_(.+)_ve_core(.+)_temp			
	0	1	ve_0_ve_core1_temp
	1	2	ve_1_ve_core2_temp

Table 12: The List of the Metrics Provided by ve.py

Metric			
	Meaning	**Attribute**	**Remarks**
ve_num			
	VE number	Integer (Fixed value)	An integer in the range one to eight in decimal
mmm_ver			
	MMM version	String	
ve_(.+)_model			
	VE model	Integer (Fixed value)	An example of the return value: 0x0001
ve_(.+)_product_type			
	VE product type	Integer (Fixed value)	An example of the return value: 0x0001
ve_(.+)_cores			
	The number of cores of a VE.	Integer (Fixed value)	
ve_(.+)_chip_serial			

Metric			
	Meaning	**Attribute**	**Remarks**
	The serial number of a VE chip	String	
ve_(.+)_board_serial			
	The serial number of a VE card	String	
ve_(.+)_driver_ver			
	VEOS version	String	
ve_(.+)_vmcfw_ver			
	VMCFW version	String	
ve_(.+)_core_clock			
	Core clock frequency of a VE	Integer (Fixed value)	The unit of the return value is MHz.
ve_(.+)_base_clock			
	Base clock frequency of a VE	Integer (Fixed value)	The unit of the return value is MHz.
ve_(.+)_memory_clock			
	Memory clock frequency of a VE	Integer (Fixed value)	The unit of the return value is MHz.
ve_(.+)_memory_size			
	Total memory size	Integer (Fixed value)	The unit of the return value is GByte.
ve_(.+)_l1_icache			
	L1 I cache size per core	Integer (Fixed value)	The unit of the return value is KByte/ core.
ve_(.+)_l1_dcache			
	L1 D cache size per core	Integer (Fixed value)	The unit of the return value is KByte/ core.
ve_(.+)_l2_cache			
	L2 cache size per core	Integer (Fixed value)	The unit of the return value is KByte/ core.
ve_(.+)_llc_cache			
	LLC size of a VE chip	Integer (Fixed value)	The unit of the return value is KByte/VE.
ve_(.+)_partition_mode			
	Partitioning mode	String	ON/OFF
ve_(.+)_throttle_level			
	Throttling level	String	
ve_(.+)_pci_bus			

Metric			
	Meaning	**Attribute**	**Remarks**
	PCI bus ID	String	
ve_(.+)_pci_device			
	PCI device ID	String	
ve_(.+)_ve_state			
	VE status	String	UNINITIALIZED/ OFFLINE/ ONLINE/ UNAVAILABLE/ MAINTENANCE
ve_(.+)_ve_core0_temp			
	VE Core0 temperature	Floating point	The unit of the return value is degree.
ve_(.+)_ve_core1_temp			
	VE Core1 temperature	Floating point	The unit of the return value is degree.
ve_(.+)_temp_ve_diode_1			
	VE CPU temperature Core8,9	Floating point	The unit of the return value is degree.
ve_(.+)_temp_ve_diode_0			
	VE CPU temperature Core0,1	Floating point	The unit of the return value is degree.
ve_(.+)_ve_fan			
	Fan rotation number	Integer	The unit of the return value is RPM.
ve_(.+)_ve_core_vdd0			
	VE internal voltage 0	Floating point	The unit of the return value is mV.
ve_(.+)_ve_core_vdd1			
	VE internal voltage 1	Floating point	The unit of the return value is mV.

Table 13: The List of the Keys Provided by veos.py

Metric			
	Meaning	**Attribute**	**Remarks**
ve_(.+)_os_state			
	OS status	String	The return value is "0","1","2", or "3". The meanings of the values are as follows: • 0:ONLINE • 1:OFFLINE • 2:INITIALIZING • 3:TERMINATING
ve_(.+)_vm_memory_size_free			
	Free memory size	Integer	The unit of the return value is MByte.

Metric			
	Meaning	**Attribute**	**Remarks**
ve_(.+)_vm_memory_size_pfree			
	Percent of free memory size	Floating point	The unit of the return value is %.
ve_(.+)_load1			
	System load average over the last 1 minute	Floating point	The unit of the return value is %.
ve_(.+)_load5			
	System load average over the last 5 minute	Floating point	The unit of the return value is %.
ve_(.+)_load15			
	System load average over the last 15 minutes	Floating point	The unit of the return value is %.
ve_(.+)_vecpu_sys			
	Percent of system time	Floating point	The unit of the return value is %.
ve_(.+)_vecpu_usr			
	Percent of user time	Floating point	The unit of the return value is %.
ve_(.+)_vecpu_idle			
	Percent of idle time	Floating point	The unit of the return value is %.

Appendix C Trouble Shooting

C.1. Operation Management

C.1.1. Entry of the SSH private key's password is required.

Problem

Entry of the SSH private key's password is required for each target machine when running `ansible-playbook` or `ansible` command.

```
$ ansible-playbook vh-update.yml

PLAY [Update SX-Aurora TSUBASA Software on VH machine]
 **********************
Enter passphrase for key '/home/admin/.ssh/id_rsa':
Enter passphrase for key '/home/admin/.ssh/id_rsa':
Enter passphrase for key '/home/admin/.ssh/id_rsa':
 :
```

```
$ ansible all -m ping
Enter passphrase for key '/home/admin/.ssh/id_rsa':
Enter passphrase for key '/home/admin/.ssh/id_rsa':
Enter passphrase for key '/home/admin/.ssh/id_rsa':
 :
```

Cause

The `ssh-agent` is not started or the private key is not registered by the `ssh-add` command after starting the `ssh-agent`.

Solution

Start the `ssh-agent` and use the `ssh-add` to register the private key on the terminal machine where you run `ansible-playbook` or the `ansible` command.

```
$ ssh-agent bash
$ ssh-add  ~/.ssh/id_rsa
Enter passphrase for /home/admin/.ssh/id_rsa: (Input
 passphrase)
Identity added: /home/admin/.ssh/id_rsa (/home/admin/.ssh/
id_rsa)
```

Related information

" *2.4.3. Starting ssh-agent* on page 25 "

C.1.2. Playbook execution fails with "sudo: sorry, you must have a tty to run sudo"

Problem

The error message "sudo: sorry, you must have a tty to run sudo" appears when running the `ansible-playbook` or the `ansible` command, and playbook execution fails.

```
$ ansible all --sudo -a 'command'
vh-name01 | FAILED! => {
    "changed": false,
    "failed": true,
    "module_stderr": "sudo: sorry, you must have a tty to run sudo\n",
    "module_stdout": "",
    "msg": "MODULE FAILURE",
    "rc": 1
}
```

```
$ ansible-playbook vh-update.yml

PLAY [Update SX-Aurora TSUBASA Software on VH machine]
 ****************************
TASK [TSUBASA-local-repo : Install SX-Aurora TSUBASA Software GPG-KEY]
 ***************
fatal: [vh-name01]: FAILED! => {"changed": false, "failed": true,
  "module_stderr": "sudo: sorry, you must have a tty to run sudo\n",
  "module_stdout": "", "msg": "MODULE FAILURE", "rc": 1}
        to retry, use: --limit @/tmp/vh.retry
```

```
$ ansible all --sudo -a 'command'
vh-name01 | FAILED! => {
    "changed": false,
    "failed": true,
    "module_stderr": "sudo: sorry, you must have a tty to run sudo\n",
    "module_stdout": "",
    "msg": "MODULE FAILURE",
    "rc": 1
}
```

Cause

requiretty in /etc/sudoers is enabled on the VH.

Solution

Add the following setting to /etc/sudoers on the VH to disable requiretty.

```
# visudo
 :
Defaults:admin !requiretty
```

Related information

" *2.4.2. Tuning Ansible performance* on page 24 "

" *2.3. Initial Setup of VHs* on page 20 "

C.1.3. Package download fails with "urlopen error [Errno 113] No route to host"

Problem

When performing tasks for software installation or update, the error message "urlopen error [Errno 113] No route to host" appears, and the playbook execution fails.

```
$ ansible-playbook vh-set.yml
 :
TASK [zabbix-agent : Install repository for zabbix-agent]
 ********************
fatal: [vh-name01]: FAILED! => {"changed": false, "failed": true,
  "msg": "Failure downloading http://xxx/xxx.rpm,
  Request failed: <urlopen error [Errno 113] No route to host>"}
            to retry, use: --limit @/tmp/vh.retry
```

Cause

A proxy server is not set up.

Solution

Check the proxy server to be used and add the following setting to vh.yml.

```
  environment:
    http_proxy: http://proxy.example.com:8080
```

Related information

" *2.7. Software Installation on VHs* on page 27 "

" *3.1.3.2. Procedure with Ansible* on page 41 "

" *3.2.3.2. Setup Using Ansible* on page 59 "

C.1.4. Yum repository update fails with "[Errno 14] HTTP Error 502 - Bad Gateway"

Problem

When executing the "Update metadata for yum" task in the TSUBASA-local-repo role to set the SX-Aurora TSUBASA Yum repository access configuration, playbook fails with "http://*x.x.x.x*/repos/TSUBASA/repodata/repomd.xml:[Errno 14] HTTP Error 502 - Bad Gateway".

```
$ ansible-playbook vh.yml
 :
TASK [TSUBASA-local-repo : Update metadata for yum]
 ************************
 :
fatal: [vh-name01]: FAILED! => {"changed": true, "cmd": "yum", "--
setopt=http_caching=packages", "makecache"],
 :
"stderr": "http://x.x.x.x/repos/TSUBASA/repodata/repomd.xml:
[Errno 14] HTTP Error 502 - Bad Gateway\n
 :
```

Cause

The proxy server is not allowed to access to the HTTP server on the management server.

Solution

Check that proxy server is required to access to the HTTP server on the management server.

- When proxy server is required:

  Configure the proxy server to be able to access to the HTTP server on the management server.
- When proxy server is not required:

  Follow the procedure below to access to the HTTP server on the management server without proxy server..

  1. Edit the following files on the management server to add "proxy=_none_" line.

     [/opt/nec/sysmng-soft/etc/ansible/roles/TSUBASA-local-repo/templates/TSUBASA-local.repo.j2]

     ```
     [TSUBASA-local]
     name=SX-Aurora TSUBASA Software Local Repository
     baseurl=http://{{ httpserver }}/{{ reposdir }}
       :
     proxy=_none_
     ```

     [/opt/nec/sysmng-soft/etc/ansible/roles/TSUBASA-InfiniBand/templates/mlnx_ofed.repo.j2]

     ```
     [mlnx_ofed]
     name=MLNX_OFED Repository
     baseurl=http://{{ httpserver }}/{{ tsubasa_ib_ofedpkgpath }}/RPMS
       :
     proxy=_none_
     ```

     Caution
     If the packages for SX-Aurora TSUBASA InfiniBand are not installed on VHs, editing the TSUBASA-InfiniBand role is not necessary.
  2. Run the Playbook again.

     ```
     $ cd /opt/nec/sysmng-soft/etc/ansible
     $ ansible-playbook vh.yml
     ```

Related information

" *2.10. Software Update on VHs* on page 29 "

" *3.1.3.2. Procedure with Ansible* on page 41 "

" *3.2.3.2. Setup Using Ansible* on page 59 "

C.2. Operational Status Monitoring(Zabbix)

C.2.1. Cannot get VE information, and 'Not supported by Zabbix Agent' is shown to item information of monitoring host

Problem

Cannot get VE information, and "Not supported by Zabbix Agent" is shown to item information of monitoring host.

Cause

Configuration of loadable module for Zabbix may be wrong.

Solution

Check the configuration of `[ /etc/zabbix/zabbix_agent.conf ]`. If the configuration is correct, display the following message to the log file `[ /var/log/zabbix/zabbix_agent.log ]` of zabbix agent.

```
loaded modules: ve_hw_item.so, ve_os_item.so
```

Related information

" *3.1.5.3. Information Gathering with Loadable Modules* on page 46 "

C.2.2. Cannot get VE sensor information, and 'File can not access' is shown to item information of monitoring host

Problem

Cannot get VE sensor information, and 'File can not access' is shown to item information of monitoring host.

Cause

VE Status is OFFLINE.

Solution

Set VE Status to ONLINE.

Related information

None.

C.2.3. Cannot get a core temperature of VE, and 'Specified core is not available' is shown to item information of monitoring host.

Problem

Cannot get a core temperature of VE, and 'Specified core is not available' is shown to item information of monitoring host.

Problem

VE core number of the item does not available.

Solution

Turn off the item.

Related information

None

C.3. Operational State Monitoring (Ganglia+Nagios)

C.3.1. Metrics for the SX-Aurora TSUBASA system are not displayed on the Ganglia web interface

Symptoms

Metrics for the Aurora system are not displayed on the Ganglia web interface.

Causes

It can be due to any of the following settings.

- Ganglia agent is not rebooted after the configuration of plugins
- The path to Aurora plugins is not correctly specified in the Ganglia configuration file `[ /etc/ganglia/conf.d/modpython.conf ]`.
- Ganglia server fails to gather metric values because of buffer starvation for communication with Ganglia agent.

Remedies

- Reboot Ganglia agent
- Check the path to the Aurora plugins is correctly specified in the file `[ /etc/ganglia/conf.d/modpython.conf ]`.
- Increase the buffer size for Ganglia server, or reduce the number of metrics of Ganglia agent.

Relevant Information

" *Configuration of Ganglia Agent (gmond)* on page 57 "

" *Configuration of Plugins* on page 57 "

" *Customization of Monitoring Items* on page 58 "

C.3.2. The values in a graph for the SX-Aurora TSUBASA system are displayed as “0” on the Ganglia web interface.

Symptoms

The values in a graph for the SX-Aurora TSUBASA system are displayed as "0" on the Ganglia web interface.

Causes

VE is not online.

Remedies

Set VE online

Relevant Information

None

C.3.3. The status of VH services is displayed as "UNKNOWN" on the Nagios web interface

Symptoms

The status of VH services is displayed as "UNKNOWN" on the Nagios web interface.

Causes

It can be due to any of the following settings.

Causes

It can be due to any of the following settings.

The port for Nagios plugins to access Ganglia server is not opened.

The hostnames of VHs specified in the file `[ /etc/nagios/ve/nodes.cfg ]` are not the same as those specified on the Ganglia web interface.

Remedies

Open the port (8651) for Nagios plugins to access Ganglia server.

Specify the same hostnames of VHs in the file `[ /etc/nagios/ve/nodes.cfg ]` as those on the Ganglia web interface

Relevant Information

" *3.2.2.1. Preparation on the Management Server* on page 52 "

" *3.2.2.3. Configuration of Ganglia Server* on page 53 "

C.4. Power Control

C.4.1. VH power control using the powerctrl commands fails.

Problem

The following operations using the `powerctrl` command fails: powering on/off VHs, information display, or resetting VHs.

Cause

Transmitting packets between the BMCs of the management server and VHs fails. The routing of the BMC network between the management server and VHs is not configured.

Solution

Configure the routing of the BMC network between the management server and VHs.

Run the `chassis status` sub-command of `ipmitool` and check that power status of VH machines is displayed.

```
$ ipmitool -H hostname -U username -P password chassis status
```

If the power status is displayed, packets are sent to the BMC on the VH machine.

Related information

" *1.3. Configuration* on page 11 "

" *4.1. Controlling power with BMC* on page 64 "

C.4.2. VH power control using the powerctrl_w commands fails.

Problem

`powerctrl_w` fails to power on VHs.

Cause

The following things might be the cause.

- WOL is not enabled on the VH machines.
- The management server and VHs reside on different subnets.

Solution

- Enable WOL on VHs.
- Configure your router so that multicast packets should be transferred from the management server to VH machines.

Related information

" *4.2. Powering on/off VHs with WOL* on page 65 "

Appendix D Ansible Playbook

D.1. How to check Ansible playbooks

This appendix explains how to perform checks when you create or edit Ansible playbooks before running playbooks.

- Checking syntax

  Specify --syntax-check option. This option performs only syntax check of a playbook and does not run the playbook.

  ```
  $ ansible-playbook xxx.yml --syntax-check
  ```

- Checking target hosts

  Specify --list-hosts option. This option displays a list of target hosts and does not run the playbook.

  ```
  $ ansible-playbook xxx.yml --list-hosts
  ```

- Checking tasks

  Specify --list-tasks option. This option displays a list of all tasks and does not run the playbook.

  ```
  $ ansible-playbook xxx.yml --list-tasks
  ```

- Checking modifications made by running a playbook

  Specify --check option. This option displays how execution results will be, and does not make any modifications to target hosts. In addition, run this command with the --diff option to check modifications made in target hosts' files.

  ```
  $ ansible-playbook xxx.yml --check --diff
  ```

Bibliography

1. SX-Aurora TSUBASA Installation Guide (with OSS)

SX-Aurora TSUBASA System Software
SX-Aurora TSUBASA
Operation and Management Guide (with OSS)

January, 2019 5th. Edition
NEC Corporation

©NEC Corporation 2018,2019