

MasterScope Network Manager WebAPI 1.1 Reference Manual

Copyrights

The information in this document is the property of NEC Corporation. No part of this document may be reproduced or transmitted in any form by any means, electronic or mechanical, for any purpose, without the express written permission of NEC Corporation.

The information in this manual may not include all the information disclosed by NEC Corporation or may include expressions that differ from information disclosed by other means. Also, this information is subject to change or deletion without prior notice.

Although every effort has been made to ensure accuracy in producing this manual, NEC Corporation does not guarantee the accuracy or applicability of the information contained herein. In addition, NEC Corporation is not liable for any loss or damage incurred as a result of the use or non-use of this information by any party.

Trademarks

- NEC and NEC logo are registered trademarks or trademarks of NEC Corporation in Japan and other countries.
- Apache Tomcat are registered trademarks of Apache Software Foundation.
- Cisco, IOS, and Catalyst are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.
- This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).
- This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).
- Other company names and product names are trademarks or registered trademarks of their respective companies.
- Trademark symbols such as [™] and [®] are not indicated in the main text.

Introduction

Thank you for choosing MasterScope Network Manager (hereafter referred to as Network Manager for short).


This manual describes WebAPI 1.1 for remotely operating the management information of Network Manager.

Please read this manual carefully before using WebAPI.

Notation and Text Conventions


In this manual, the following notation is used to indicate items that require special attention and supplementary information.

Notation of items requiring attention and supplementary information

Mark	Description
 Caution	Indicating important points that the user should observe to configure and use the product properly.
Tip	Indicating useful information.

In this manual, the following text conventions are used.

Text conventions

Notation	Description	Example
	Indicating graphical user interfaces such as dialog boxes, tabs, menus, items, and buttons.	Network View window, OK button
<code><userinput></code>	Indicating items that change depending on the user environment or items that the user must specify.	<code><filepath></code>
<code>configuration file</code>	Indicating the contents of the configuration file.	Set the following value <code>port = 27120</code>
<code>command line</code>	Indicating command line operations.	Run the following command: <code>> NvPROReloadDefFileMgr</code>

Contents

Chapter 1. WebAPI Overview	1
1.1 Operations that WebAPI Supports	2
1.2 System Architecture for using WebAPI	3
1.3 Pre-configuration for using WebAPI	4
Chapter 2. Common Specification of WebAPI.....	6
2.1 Request Format	7
2.1.1 End point.....	7
2.1.2 Authentication Method.....	7
2.1.3 HTTP Version	8
2.1.4 HTTP Methods.....	8
2.1.5 HTTP Request Header	8
2.1.6 Specifying Query Parameter.....	9
2.2 Request and Response Data Format	10
2.3 Response Format.....	11
2.3.1 HTTP Response Header	11
2.3.2 Common HTTP Status Codes.....	12
2.3.3 Error Response Format.....	13
2.4 WebAPI Version.....	13
Chapter 3. Resource Format.....	15
3.1 Map Resource (MapObject)	16
3.2 Node Resource (NodeObject).....	17
3.3 Map Member Resource (MapMemberObject).....	26
3.4 Node Status Resource (NodeStatusObject).....	27
3.5 Node Member Resource (NodeMemberObject)	27
3.6 Status Monitoring Rule Entry Resource (StsMonObject)	28
3.7 Status Monitoring Rule Resource (StsMonRuleObject).....	30
3.8 Data Collection Entry Resource (DataColObject)	31
3.9 Alert Resource (AlertObject)	35
3.10 Notes of Caution in Operating Resource	38
Chapter 4. WebAPI Reference.....	39
4.1 Configuration Information Management API	40
4.1.1 Obtaining Map List	40
4.1.2 Obtaining Map Details	41
4.1.3 Adding Map	41
4.1.4 Updating Map	42
4.1.5 Deleting Map	44
4.1.6 Obtaining Node List.....	44
4.1.7 Obtaining Node Details.....	46

4.1.8 Adding Node.....	47
4.1.9 Updating Node.....	51
4.1.10 Deleting Node.....	55
4.1.11 Updating Device Information of Node.....	56
4.1.12 Obtaining Severity Status List of Node	58
4.1.13 Obtaining Severity Status of Node	60
4.1.14 Obtaining Map Configuration.....	60
4.1.15 Expanding Standard Component Name Specification Format	62
4.2 Status Monitoring Setting API	63
4.2.1 Obtaining Status Monitoring Rule Entry List	63
4.2.2 Obtaining Status Monitoring Rule Entry	63
4.2.3 Adding Status Monitoring Rule Entry	64
4.2.4 Updating Status Monitoring Rule Entry	65
4.2.5 Deleting Status Monitoring Rule Entry	67
4.2.6 Obtaining Status Monitoring Rule List.....	67
4.3 Data Collection Setting API	68
4.3.1 Obtaining Data Collection Entry List	68
4.3.2 Obtaining Data Collection Entry	69
4.3.3 Adding Data Collection Entry	70
4.3.4 Updating Data Collection Entry	74
4.3.5 Deleting Data Collection Entry	79
4.4 Alert Management API	79
4.4.1 Obtaining Alert List	79
4.4.2 Obtaining Alert Details	83
4.4.3 Updating Alert	84
4.4.4 Updating Alert Properties in a Batch	85
4.4.5 Deleting Alert	88
4.4.6 Deleting Alerts in a Batch	88
Chapter 5. Tutorial.....	91
5.1 Operating Configuration Information	92
5.1.1 Adding New Map.....	92
5.1.2 Deleting Information of Map and its Subordinates in a Batch	93
5.1.3 Adding New Node	96
5.1.4 Updating Group Information that Node belonging	116
5.1.5 Suspending Node Monitoring belonging to Specified Map	118
5.1.6 Deleting Node	124
5.1.7 Confirming Fault Condition of Nodes belonging to Specified Map.....	126
5.2 Operating Status Monitoring Rule Entry	128
5.2.1 Adding New Status Monitoring Rule Entry.....	128
5.2.2 Updating Monitoring Target of Status Monitoring Rule Entry	132
5.2.3 Deleting Status Monitoring Rule Entry	136
5.3 Operating Data Collection Entry	140
5.3.1 Adding New Data Collection Entry	140
5.3.2 Updating Collection Interval of Data Collection Entry.....	142
5.3.3 Deleting Data Collection Entry	146

5.4 Operating Alert	149
5.4.1 Confirming Alert of Target Node	149
5.4.2 Recovering Target Alert	152
Appendix A. Standard Specification Format	155
A.1 Standard Matching Specification Format.....	155
A.2 Standard Component Name Specification Format	155
Appendix B. Specifying wildcards	157

Chapter 1.

WebAPI Overview

Network Manager provides WebAPI for remotely operating the management information of it. This chapter describes the overview of WebAPI, the system architecture and the pre-configuration for using WebAPI.

Contents

1.1 Operations that WebAPI Supports	2
1.2 System Architecture for using WebAPI	3
1.3 Pre-configuration for using WebAPI	4

1.1 Operations that WebAPI Supports

This WebAPI supports the following operations.

Table 1-1 Operations WebAPI supports.

Function name	Operation	Description
Configuration Information Management	Obtaining map list	Obtaining, adding, updating and deleting the property information of the map. These are the same functions as the map view operations of the monitoring device window.
	Obtaining map details	
	Adding map	
	Updating map	
	Deleting map	
	Obtaining node list	Obtaining, adding, updating and deleting the property information of the node. In addition, obtaining the interface property information. These are the same functions as the map view operations of the monitoring terminal window. However, the operations of the login information are not supported.
	Obtaining node details	
	Adding node	
	Updating node	
	Deleting node	
	Updating device information of node	Updating the property information of the registered node by obtaining the information using SNMP. Providing the same functions as Update Property menu of the monitoring terminal window and execution of <code>nvpnnodeup</code> command.
	Obtaining severity state list of node	Obtaining the severity status (alert occurrence state) of the node. These are the same functions as the icon color confirmation functions of the map view of the monitoring terminal window.
	Obtaining severity state of node	
	Obtaining map configuration	Obtaining the configuration information of the map and node belonging to the specified map. This is the function to confirm the configuration information, that is same as the tree view displaying function of the monitoring terminal window.
	Expanding standard component name specification format	Obtaining the component name corresponding to the specified standard component name specification format. In the case of specifying the target component of the status monitoring rule entry or the data collection entry, this operation is used to confirm the target specific component when the standard component name specification format is used.
State Monitoring	Obtaining state monitoring rule entry list	Obtaining, adding, updating and deleting the information of the status monitoring rule entry. In addition, obtaining the built-in state information of the status monitoring rule. These are the same operation functions as those of State Monitoring view of the monitoring terminal window. However, the embedding and removing operations of the status monitoring rule are not supported.
	Obtaining state monitoring rule entry	
	Adding state monitoring rule entry	
	Updating state monitoring rule entry	
	Deleting state monitoring rule entry	

Function name	Operation	Description
	Obtaining state monitoring rule list	
Data Collection	Obtaining data collection entry list	Obtaining, adding, updating and deleting the information of the data collection entry.
	Obtaining data collection entry	These are the same operation functions as those of Data Collecting Setting view of the monitoring terminal window. However, the following operations are not supported. <ul style="list-style-type: none"> Operation of the cumulative schedule in the threshold monitoring and setting. Operations of adding a new MIB calculation formula and confirming the registered definition (i.e. operations corresponding to those of MIB Expression Creating window).
	Adding data collection entry	
	Updating data collection entry	
	Deleting data collection entry	
Alert Management	Obtaining alert list	Obtaining, updating and deleting the alert information.
	Obtaining alert	These are the same operation functions as those of Alert Management window of the monitoring terminal window.
	Confirming alert	
	Confirming alert in a batch	
	Deleting alert	
	Deleting alert in a batch	

Refer to "[Chapter 4. WebAPI Reference \(page 39\)](#)" for the details of each API.

1.2 System Architecture for using WebAPI

This section describes the system architecture for using WebAPI.

To use WebAPI, it needs to install the following two components of the Service Governor on the server that the manager function is installed.

Service Governor:

- MasterScope Service Governor
- Application Server (Tomcat)

The client system using WebAPI accesses to the Service Governor and operates MasterScope Network Manager as described in "[Figure 1-1 Standard architecture for using WebAPI \(page 3\)](#)".



Figure 1-1 Standard architecture for using WebAPI

In the environment that multiple MasterScope Network Manager are located, install the Service Governor on one of the servers that the manager function is installed as described in "[Figure 1-2 System architecture under multiple managers environment \(pattern 1\) \(page 4\)](#)".

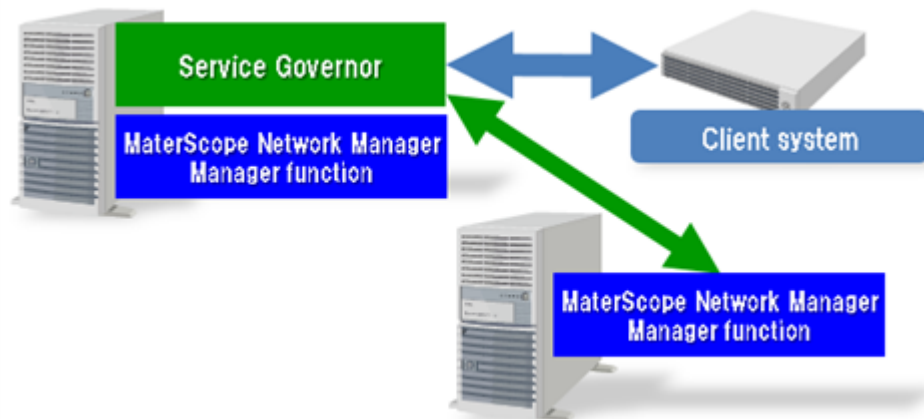


Figure 1-2 System architecture under multiple managers environment (pattern 1)

As described in "Figure 1-3 System architecture under multiple managers environment (pattern 2) (page 4)", it is possible to configure the system by installing the Service Governor on a different server with those of the manager function is installed.

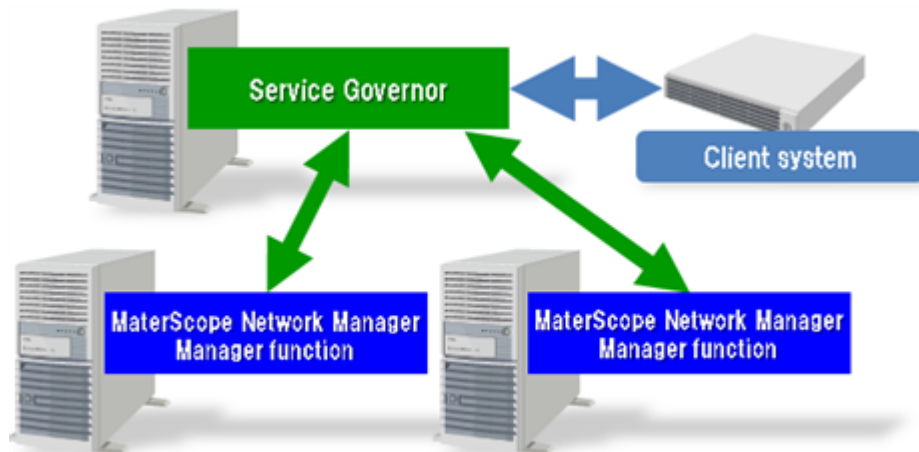


Figure 1-3 System architecture under multiple managers environment (pattern 2)

In any case, the client system using WebAPI can operate all MasterScope Network Manager by accessing the Service Governor.

1.3 Pre-configuration for using WebAPI

It needs to set configurations before using WebAPI.

Configure in the following steps before using WebAPI.

Table 1-2 Steps of configuration for using WebAPI

No.	Operation	Outline
1	Installing service governors	Install service governors in the following steps. <ol style="list-style-type: none"> 1. Determine a server to install service governors by referring to "1.2 System Architecture for using WebAPI (page 3)". 2. Install service governors (i.e. MasterScope Service Governor and Application Server) using installation media.
2	Changing firewall settings	Set the firewall rules to allow the communication between the client system and the service governors, and between the service governors and the manager functions.

No.	Operation	Outline
3	Pre-configuring of SSL/TLS communication between service governors and manager functions (optional)	<p>Pre-configure the system in the following steps when using HTTPS in the communications between the service governors and the manager functions.</p> <p>Tip</p> <p>It is recommended to perform this configuration in the environment that the service governors and the manager functions are installed on different servers.</p> <ol style="list-style-type: none"> 1. Write the file names of the server certificate and the secret key to the definition file, i.e. HttpServerMgr.ini, of the manager functions. 2. Store the files of the server certificate and the secret key on the fixed location of the manager functions.
4	Setting connection between service governors and manager functions	<p>Set the connection between the service governors and the manager functions of MasterScope Network Manager in the following steps.</p> <ol style="list-style-type: none"> 1. Set the connection of the manager functions side by using the monitoring device functions. 2. Set the connection of the service governors side by updating the definition file, i.e. fwapi.properties, of the service governors.
5	Configuring SSL/TLS communication between client system and service governors (optional)	<p>Change the configuration of the service governors in the following steps only when the client system using WebAPI connects to WebAPI using HTTPS.</p> <ol style="list-style-type: none"> 1. Import the server certificate to the service governors. 2. Edit the Tomcat definition file, i.e. server.xml, depending on the contents of the imported server certificate. 3. Open the env script in a text editor and update the contents.

Refer to “*MasterScope Service Governor Installation Guide*” recorded on the installation media for the details of configuration instructions.

Chapter 2.

Common Specification of WebAPI

This chapter describes the common specification of a request and a response format for using WebAPI.

Contents

2.1 Request Format	7
2.2 Request and Response Data Format	10
2.3 Response Format.....	11
2.4 WebAPI Version.....	13

2.1 Request Format

This section describes the request format of WebAPI.

2.1.1 End point

The following URLs are used as end points of API.

- `http://<Domain Name>/umf/fw/nvp/`
- `https://<Domain Name>/umf/fw/nvp/`

2.1.2 Authentication Method

In this product, all APIs require the authentication for each request.

Authenticating the request by appending the following authentication information to the HTTP request header.

```
Date: Date
Authorization: SharedKeyLite AccessKeyId:Signature
```

Users create the authentication information in the following steps.

1. Running the following WebApiTool.bat commands to create AccessKeyId and SecretAccessKey.

```
AccessKeyId = "Ws/jdOKW1VBwbjwgNq49BS+sc5cLKs9Qh5LeafXDVSg="
SecretAccessKey = "2slMDrTqYwlixmzUX/CDXTEFHi9WIMjXhIR6quKmFrE="
```

Refer to “API Authentication Key” part of “MasterScope Service Governor Installation Guide” for the details of the way to run WebApiTool.bat commands.

2. Determining the request date(Date).

```
Date: Fri, 21 Aug 2015 13:15:45 GMT
```

3. Determining the WebAPI path to execute without query characters (i.e., "?" and its following characters).

```
CanonicalizedResource = "/umf/fw/nvp/nodes"
```

4. Concatenating the request date(Date) and the WebAPI path by using a line feed character(CRLF) and creating the character string to be authenticated.

```
CRLF = "\x0D\x0A"
StringToSign = Date + CRLF + CanonicalizedResource
```

5. Creating HMAC(SHA256) based on the character string to be authenticated and SecretAccessKey, and then, encoding it in Base64.

```
Signature = Base64(HmacSHA256(SecretAccessKey, StringToSign))
#=> "rpFv195+j7TUV+C1W4LpPb7sBskmjVyUV2jNTz69UfU="
```

6. Creating the HTTP request header.

Date: Fri, 21 Aug 2015 13:15:45 GMT

Authorization: SharedKeyLite Ws/jdOKW1VBwbjwgNq49BS+sc5cLKs9Qh5LeafXDV

Sg=:rpFv195+j7TUV+C1W4LpPb7sBskmjVyUV2jNTz69UfU=

2.1.3 HTTP Version

This subsection describes the details of supported HTTP versions.

Table 2-1 Supported HTTP versions

Protocol	HTTP/1.0	HTTP/1.1	HTTP/2
HTTP	-	o	-
HTTPS	-	o	-

2.1.4 HTTP Methods

Four types of HTTP methods are used in WebAPI of this product.

Table 2-2 List of used HTTP methods

Method	Usage
GET	Obtaining the list and the details of the existing resource.
POST	Creating a new resource. A part of property information of the new created resource can be omitted depending on the used WebAPI. This method is also used to execute the operations other than GET(obtaining list and details), PUT(updating) and DELETE(deleting) of the resource.
PUT	Updating the existing resource. A part of property information of the updated resource can be omitted depending on the used WebAPI.
DELETE	Deleting the existing resource.

2.1.5 HTTP Request Header

The HTTP request header is needed to send the HTTP request to WebAPI.

o: Mandatory, *: optional, -: not used

Table 2-3 List of required HTTP request header

Header	Method				Description
	GET	POST	PUT	DELETE	
Content-Type	-	*	*	-	Specifying the data format in the requesting. Set "application/json". Processed in the same way as "application/json" is set when setting the value of this header is omitted or either "application/*" or "*/*" is set.
Content-Length	-	o	o	-	Specifying the request data size with a byte count.
Accept	*	*	*	*	Specifying the receivable data format in the responding. Set "application/json". Processed in the same way as "application/json" is set when setting the value of this header is omitted or either "application/*" or "*/*" is set.
Accept-Charset	*	*	*	*	Specifying the receivable character set in the responding.

Header	Method				Description
	GET	POST	PUT	DELETE	
					Set "utf-8". Processed in the same way as "utf-8" is set when setting the value of this header is omitted.
Connection	*	*	*	*	Specifying whether continuing to use TCP connection or not. Setting "close" only when expressly stopping to use TCP connection. Otherwise, including the case that setting the value of this header is omitted, this header is processed in the same way as the continuous TCP connection is requested.
Host	o	o	o	o	Specifying the host name of the request destination server.
X-NVP-API-Version	o	o	o	o	Specifying the API version with the format of "major.minor". The value of this header except for the most significant one digit (major) can be omitted. When the second or after digit (minor) is omitted, this header is processed in the same way as the newest version is set in the omitted part. Set "1" or "1.1" in this version. Refer to "2.4 WebAPI Version (page 13)" for the details.
Date	o	o	o	o	Specifying the issue date of the request. e.g.) "Sun, 04 Mar 2012 08:12:31 GMT"
Authorization	o	o	o	o	Specifying the information used for the authentication. Specifying with the format of "SharedKeyLite AccessKeyId:Signature". Refer to "2.1.2 Authentication Method (page 7)" for the details.

2.1.6 Specifying Query Parameter

Specifying the parameters used for controlling the detailed operations of WebAPI with the format of the URL query character string.

URL query indicates the parameters that are appended to the URL in the requesting in the format of "key=value".

Appending "?" character to the end of the URL, and then, appending the parameters after it in the format of "key=value". When using multiple URL queries, concatenating each parameter with "&" character.

Example:

```
GET /umf/fw/nvp/nodes?ID=12345&Detail=full
```

General caution of specifying API parameters

- Specifying "Detail=" is equivalent to setting an empty character string in "Detail" WebAPI parameter. In many WebAPIs, operations in the case that nothing is specified and that an empty

character string is set are the same, on the other hand, some WebAPIs operates differently in the respective case. Confirm the specification of each WebAPI before using.

- Not only URLs but also URL query strings become the targets of URL encoding.

Standard parameter

The standard parameter commonly used in all WebAPIs is described in the following.

Table 2-4 Standard parameter list

Parameter	Omission	Description
ManagerName	Y	Specifying the name of the target manager in the environment that multiple managers exist. Setting the value of this parameter can be omitted in the environment that the manager is only one. "409 Conflict" HTTP state code is returned when this parameter is not specified in the environment that multiple managers exist. "404 NotFound" is returned when the specified manager is not found.

2.2 Request and Response Data Format

The data format of the message body of requests and responses is described as follows.

- Character encoding

Encoding of character strings supports UTF-8.

- Format of message body

Supporting JSON format defined in RFC 4627.

Caution

In JSON format, "number" type is defined as the data type to be processed as integers, and then, integer notation, decimal notation and exponent notation can be used to present numeric data.

However, WebAPI of MasterScope Network Manager supports only integer notation for representing numeric data. For clarifying that WebAPI supports only integer notation, this manual uses not "number" type but the following originally defined data type in the explanation of each resource property.

Table 2-5 Original data type for representing numeric data

Data type	Description
integer	The data type that supports only integer notation of all types defined in JSON format. An error is returned when the numeric data including a decimal (e.g. "10.0") or an numeric data with exponent notation (e.g. "1e+1") is specified as the input for the resource property of this data type.

- Format of date

Supporting the subset of the following date format (time-secfrac is not supported) defined in RFC3339 to specify the string representing the date.

Table 2-6 Dateformat

	Date format	Description
1	YYYY-mm-ddTHH:MM:SS	Year, month, date plus hours, minutes and seconds in UTC time zone.
2	YYYY-mm-ddTHH:MM:SSZ	
3	YYYY-mm-ddTHH:MM:SS-HH:MM	Year, month, date plus hours, minutes and seconds in the time zone facing west for HH hours and MM minutes from UTC time zone.
4	YYYY-mm-ddTHH:MM:SS+HH:MM	Year, month, date plus hours, minutes and seconds in the time zone facing east for HH hours and MM minutes from UTC time zone.

2.3 Response Format

This section describes the response format of WebAPI.

2.3.1 HTTP Response Header

The HTTP header is appended to the HTTP response of WebAPI.

o: appended, *: optionally appended, -: not appended

Table 2-7 List of appended HTTP response header

Header	Method				Description
	GET	POST	PUT	DELETE	
Content-Type	o	o	o	* 1)	Notifying the data format in the responding. Content-Type: application/json; charset=utf-8
Content-Length	*	*	*	* 1)	Notifying the size of the response data with a byte count. This header is not appended when the response uses chunked transfer encoding.
Transfer-Encoding	*	*	*	* 1)	Notifying the encoding format used in the transferring. This header is appended when the response uses chunked transfer encoding. Content-Length header is not appended when this header is set to "Transfer-Encoding: chunked".
Connection	*	*	*	*	Notifying the state of the connection (e.g. Keep-Alive or close). Processed in the same way as "Keep-Alive" is set when this header is not appended. The client received the response whose Connection header is set to "close" needs to reestablish a connection because it cannot use the previous TCP connection again.
X-NVP-API-Version	o	o	o	o	Notifying the version of the called WebAPI in the format of "major.minor.revision". Note that this format is different from that of the request header.

Note

1. These headers are appended only when DELETE(Delete process) failed. These are not appended when the delete process succeeded because the response has no message body in the case of success.

Caution

A part of the HTTP headers may not be appended regardless of the contents described in "[Table 2-7 List of appended HTTP response header \(page 11\)](#)" when the API process failed.

2.3.2 Common HTTP Status Codes

The result of success or failure of the API process is reported by HTTP status codes.

Common HTTP status codes of each API are described in "[Table 2-8 List of HTTP status codes API returns \(page 12\)](#)". Refer to "[Chapter 4. WebAPI Reference \(page 39\)](#)" for specific HTTP status codes of each API. Refer to "[2.3.3 Error Response Format \(page 13\)](#)" for the format of the error response.

Table 2-8 List of HTTP status codes API returns

code	Description
400 Bad Request	<p>The structure of the request is wrong. Examples of the common causes of this error in each API are as follows.</p> <ul style="list-style-type: none"> • HTTP request header "X-NVP-API-VERSION" is not specified. • The specified API version in HTTP request header "X-NVP-API-VERSION" is invalid. • The format of the message body of the HTTP request is not JSON format. (This is applied only when the message body of the request is specified.)
401 Unauthorized	<p>The user authentication failed. Examples of the common causes in each API are as follows.</p> <ul style="list-style-type: none"> • The authentication key is not specified. • The authentication key is invalid.
404 Not Found	<p>The specified resource does not exist. Examples of the common causes in each API are as follows.</p> <ul style="list-style-type: none"> • The specified URL is invalid. • The manager specified in ManagerName parameter does not exist.
405 Method Not Allowed	<p>A HTTP method that is not allowed to use is specified. An example of the common cause in each API is as follows.</p> <ul style="list-style-type: none"> • Specifying a HTTP method that is not supported in the API request format.
406 Not Acceptable	<p>The information that is not allowed to accept is included in the Accept-related header of the request. Examples of the common causes in each API are as follows.</p> <ul style="list-style-type: none"> • The specification in Accept header does not match the response format that the server supports. • The specification in Accept-Charset header does not match the character set that the server supports.
411 Length Required	<p>The server denied the access because the Content-Length header in the request is inadequate. An example of the common cause in each API is as follows.</p> <ul style="list-style-type: none"> • Content-Length header is not specified.
413 Request Entity Too Large	<p>The request entity is beyond the acceptable range of the server. An example of the common cause in each API is as follows.</p> <ul style="list-style-type: none"> • The size of the message body in the request is too large.

code	Description
414 Request-URI Too Long	The length of the request URI is beyond the acceptable range of the server. An example of the common cause in each API is as follows. <ul style="list-style-type: none"> The specified URL is too long.
415 Unsupported Media Type	The specified media type is not supported in the server. An example of the common cause in each API is as follows. <ul style="list-style-type: none"> The specification in Content-Type header in the request does not match the media type that the server supports.
500 Internal Server Error	An error occurs in the server during the process of API. Examples of the common causes in each API are as follows. <ul style="list-style-type: none"> There is no manager. An unexpected error occurs.
503 Service Unavailable	The service is temporarily overloaded and unavailable. Examples of the common causes in each API are as follows. <ul style="list-style-type: none"> The server can not accept the request because it is in initialization process of a service. The server can not temporarily accept a new request because it is processing a lot of requests.

2.3.3 Error Response Format

The status codes of 400 series (i.e. Client Error) or 500 series (i.e. Server Error) that are described in "2.3.2 Common HTTP Status Codes (page 12)" is returned in the status line of the HTTP response when the API process failed. In addition, each API may return the details of error contents in the following JSON format¹⁾.

Resource format

```
{
  "Error" : string,
  "Exception" : string
}
```

Table 2-9 Properties

Property Name	JSON Type	Description
Error	string	The error message for describing the contents of the error.
Exception	string	The Exception information that is corresponding to the HTTP status code.

Note

- The format other than JSON format (e.g. HTML format) may be used in the message body depending on the error contents.

2.4 WebAPI Version

This product controls the compatibility among different product versions using API version.

When an API is called by specifying an older version of API, the functions of the specified API version are supported.

List of supported APIs

- Version 1.1 or later
 - ["4.4 Alert Management API \(page 79\)"](#)
- 1.0
 - ["4.1 Configuration Information Management API \(page 40\)"](#)
 - ["4.2 Status Monitoring Setting API \(page 63\)"](#)
 - ["4.3 Data Collection Setting API \(page 68\)"](#)

Correspondence list with product version

Product version	API version
9.0	1.1 (1.1.1)
8.0	1.1

Chapter 3.

Resource Format

This chapter describes the resource format of the request and response message body of each API.

Contents

3.1 Map Resource (MapObject)	16
3.2 Node Resource (NodeObject).....	17
3.3 Map Member Resource (MapMemberObject)	26
3.4 Node Status Resource (NodeStatusObject).....	27
3.5 Node Member Resource (NodeMemberObject)	27
3.6 Status Monitoring Rule Entry Resource (StsMonObject)	28
3.7 Status Monitoring Rule Resource (StsMonRuleObject).....	30
3.8 Data Collection Entry Resource (DataColObject)	31
3.9 Alert Resource (AlertObject)	35
3.10 Notes of Caution in Operating Resource	38

3.1 Map Resource (MapObject)

The resource format of the map is described in the following.

Resource format

```
{
  "ID" : integer,
  "Name" : string,
  "IconType" : string,
  "Alias" : string | null,
  "NetworkAddress" : string | null,
  "NetworkMask" : string | null,
  "IPv6NetworkAddress" : string | null,
  "IPv6PrefixLength" : integer | null,
  "Administrator" : string | null,
  "Location" : string | null,
  "ApplicationPath" : string | null,
  "URL" : string | null
}
```

Properties

Property Name	JSON Type	Description
ID	integer	The identification of the map. Specified with a number from 1 to 2147483647. This value is invariable from when the map is created to when the map is deleted, and this is for reference only.
Name	string	The name of the map. Specified with a character string of at least 1 character and up to 63 characters using alphanumeric characters, multi-byte characters, hyphens(-), under bars(_) and dots(.). Parts of functions of Network Manager are not available when JIS2004 characters or "nul"(these are case-insensitive) are set in the process of adding or editing a map. In addition, it is not allowed to register multiple maps that have the same name in the system.
IconType	string	The icon type. Specified in the range of the icon types described in "Appendix B: List of icon type" of user's manual.
Alias	string	The alias of the map. Specified with a character string of up to 255 characters.
NetworkAddress	string	The network address of IPv4. Specified in the format of IPv4 such as 192.168.10.0.
NetworkMask	string	The subnet mask of IPv4. Specified in the format of IPv4 such as 255.255.255.0.
IPv6NetworkAddress	string	The network address of IPv6. Specified with the global unicast address that is composed of characters from 0 to 9, a to e, A to E and colons(:).

Property Name	JSON Type	Description
IPv6PrefixLength	integer	The prefix length of IPv6. Specified with a number from 0 to 128.
Administrator	string	The manager of the map. Specified with a character string of up to 255 characters.
Location	string	The location of the map. Specified with a character string of up to 255 characters.
ApplicationPath	string	The application path of the application that is run when Start Application menu of the target map is selected. Specified with a character string of up to 4096 characters, and specified with either a absolute path or a relative path.
URL	string	The URL that is accessed when Start Web browser menu of the target map is selected. Specified with a string of up to 2083 characters.

3.2 Node Resource (NodeObject)

The resource format of the node is described in the following.

Resource format

```
{
  "ID" : integer,
  "Name" : string,
  "IconType" : string,
  "Alias" : string | null,
  "IPAddress" : string | null,
  "IPv6Address" : string | null,
  "LocalIPAddress" : string | null,
  "SNMPVersion" : "1" | "2C" | "3" | null,
  "SNMPCommunityGet" : string | null,
  "SNMPCommunitySet" : string | null,
  "SNMPPort" : integer | null,
  "SNMPCharCode" : "Japanese (JIS 0208-1990 and 0212-1990)" |
    "Japanese (Shift-JIS)" | "Unicode (UTF-8)" |
    null,
  "DefaultTargetPort" : string | null,
  "HardwareType" : string | null,
  "OSType" : string | null,
  "Administrator" : string | null,
  "Location" : string | null,
  "Memo" : string | null,
  "RoutingControl" : "forwarding" | "not forwarding" | null,
  "SoftwareVersion" : string | null,
  "SerialNumber" : string | null,
  "Group" : string | null,
  "DeviceFrontPanel" : string | null,
  "IPv4Interfaces" : [
    {
      "Index" : integer,
      "MACAddress" : string | null,
```

```

        "Type" : string,
        "Description" : string | null,
        "FEX_ID" : integer | null,
        "IPv4" : [
            {
                "Address" : string,
                "SubnetMask" : string
            },
            ...
        ]
    },
    ...
],
"IPv6Interfaces" : [
    {
        "Index" : integer,
        "MACAddress" : string | null,
        "Description" : string | null,
        "LowerLayer" : string,
        "IPv6" : [
            {
                "Address" : string,
                "PrefixLength" : integer
            },
            ...
        ]
    },
    ...
],
"SNMPv3UserName" : string | null,
"SNMPv3EngineID" : string | null,
"SNMPv3SecurityLevel" : "NoAuth/NoPriv" | "Auth/NoPriv" | "Auth/Priv",
"SNMPv3AuthenticationProtocol" : "MD5" | "SHA1" | null,
"SNMPv3AuthenticationPassword" : string | null,
"SNMPv3PrivacyProtocol" : "DES" | null,
"SNMPv3PrivacyPassword" : string | null,
"SNMPv3SeverityOfInvalidEngineID" :
    "Warning" | "Minor Fault" | "Major Fault" | "Critical State" |
    null,
"ApplicationPath" : string | null,
"URL" : string | null,
"DiscoveryProtocol" : 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
    100 | 101 | 200 | 201 | null,
"TelnetServer" : "On" | "Off",
"AgentType" : integer | null,
"AdministrationNodeName" : string | null,
"SysName" : string | null,
"MonitoringMode" : "On" | "Off",
"SysObjectID" : string | null,
"SysDescr" : string | null,
"FEX_ID" : integer | null,
"DPID" : string | null,
"FloatingIP" : string | null,
"WebAccessPort" : integer | null
}

```


Properties

Property Name	JSON Type	Description
ID	integer	The identification of the node. Specified with a number from 1 to 2147483647. This value is invariable from when the node is created to when the node is deleted, and this is for reference only.
Name	string	The name of the node. Specified with a character string of at least 1 character and up to 63 characters using alphanumeric characters, multi-byte characters, hyphens(-), under bars(_) and dots(.). Parts of functions of MasterScope Network Manager are not available when JIS2004 characters or "nul" (these are case-insensitive) are set in the process of adding or editing a node.
IconType	string	The icon type. Specified in the range of the icon types described in "Appendix B: List of icon type" of user's manual.
Alias	string	The alias of the node. Specified with a character string of up to 255 characters.
IPAddress	string	The network address of IPv4. Specified in the format of IPv4 such as 192.168.0.1 Some functions that need IPv4 address are not available when this property is set to null.
IPv6Address	string	The network address of IPv6. Specified with the global unicast address that is composed of characters from 0 to 9, a to e, A to E and colons(:). Some functions that need IPv6 address are not available when this property is set to null.
LocalIPAddress	string	The local network address of IPv4 in NAT environment. Specified in the format of IPv4 such as 192.168.0.1.
SNMPVersion	string	SNMP version. <ul style="list-style-type: none"> • 1 : Version 1 • 2C : Version 2C • 3 : Version 3 In the adding and updating processes, C as in "2C" is case-insensitive, on the other hand, this character is converted in upper case in the obtaining process. When the value of this property is null, it is processed as 1 is set in the SNMP communication.
SNMPCommunityGet	string	The SNMP community name for Get method. Specified with a character string of up to 255 characters. A SNMP communication is not established when the SNMPVersion is set to 1 or 2C and this property is set to null in the adding or updating processes.

Property Name	JSON Type	Description
		When an empty character string is set in this property, a SNMP communication is established with using the null character string as its community name.
SNMPCommunitySet	string	<p>The SNMP community name for Set method.</p> <p>Specified with a character of up to 255 characters.</p> <p>When SNMPVersion is set to 1 or 2C and this property is set to null in the adding and updating processes, a SNMP communication is established using the community name specified in SNMPCommunityGet.</p> <p>When SNMPCommunication is set to null too, a SNMP communication is not established.</p> <p>When an empty character string is set in this property, a SNMP communication is established with using the null character string as its community name.</p>
SNMPPort	integer	<p>The port number used for SNMP GET/SET messages.</p> <p>Specified with a number from 1 to 65535.</p> <p>When this property is set to null, port 161 is used in SNMP communications.</p>
SNMPCharCode	string	<p>The string code for decoding the information obtained from network devices using SNMP.</p> <p>Specified in one of the following formats. This property is case-insensitive.</p> <ul style="list-style-type: none"> • "Japanese (JIS 0208-1990 and 0212-1990)" • "Japanese (Shift-JIS)" • "Unicode (UTF-8)" <p>This property is processed in the same way as "Japanese (Shift-JIS)" is set when null is set.</p>
DefaultTargetPort	string	<p>The default target port that is used when the port number is omitted in the setting of the entries of the state monitoring rule and the data collection.</p> <p>Specified with a character string of up to 4090 characters.</p> <p>Multiple port numbers can be set by specifying the value of ifIndex (integer) and separating by commas(,) in the adding and updating processes.</p> <p>In addition, half-width spaces can be used in front and behind the value.</p>
HardwareType	string	<p>The hardware type.</p> <p>Specified with a character string of up to 255 characters.</p>
OSType	string	<p>The type of the operation system.</p> <p>Specified in the range of character strings that can be selected in the icon property.</p> <p>Processed in the same way as null is set when "OhterHost" is set in this property.</p>
Administrator	string	<p>The administrator name of the device.</p> <p>Specified with a character string of up to 255 characters.</p>
Location	string	<p>The location of the device.</p> <p>Specified with a character string of up to 255 characters.</p>

Property Name	JSON Type	Description
Memo	string	<p>The memo information.</p> <p>Specified with a character string of up to 255 characters.</p> <p>A line feed code is not allowed to use in this property in the adding and updating processes.</p>
RoutingControl	string	<p>The device identification information that is used for identifying whether the device controls network paths or not.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "forwarding": Activates IP forwarding. • "not forwarding": Inactivates IP forwarding. <p>Processed in the same way as "not forwarding" is set when null is set in this property.</p>
SoftwareVersion	string	<p>The version of the device software.</p> <p>Specified with a character string of up to 255 characters.</p> <p>A character string including commas is not allowed to use in this property in the adding and updating processes.</p>
SerialNumber	string	<p>The serial number of the device.</p> <p>Specified with a character string of up to 255 characters.</p>
Group	string	<p>The group name that the node belongs to.</p> <p>The maximum length of each group name is up to 63 characters using alphanumeric characters, multi-byte characters, hyphens (-), underscores (_) and dots (.). In addition, the maximum length of the group name including each group name and colons(:) (these are used to separate each group name) is up to 1000 characters.</p> <p>When this property is set to null or an empty character string is set, the node is processed as it does not belong to any group.</p>
DeviceFrontPanel	string	<p>The type of the panel window.</p> <p>Specified with a character string of up to 255 characters.</p> <p>When this property is set to null, this property is processed in the same way as "NDEVICE" is set, that indicates a generic panel.</p>
IPv4Interfaces	object[]	<p>The information of the IPv4 interface.</p> <p>For reference only.</p> <p>Specified with the array of each IPv4 interface resource.</p> <p>Refer to "IPv4 properties (page 25)" and "IPv4 Interfaces properties (page 25)" for the details of properties of IPv4 interface resource.</p> <p>An empty array is set in this property in the case that the interface information is not obtained or that there is no interface information in the result of obtaining the information.</p>
IPv6Interfaces	object[]	<p>The information of the IPv6 interface. For reference only.</p> <p>Specified with the array of each IPv6 interface resource.</p> <p>Refer to "IPv6 properties (page 26)" and "IPv6 Interfaces properties (page 25)" for the details of properties of IPv6 interface resource.</p> <p>An empty array is set in this property in the case that the interface information is not obtained or that there is no interface information in the result of obtaining the information.</p>
SNMPv3UserName	string	<p>The user name of SNMPv3.</p>

Property Name	JSON Type	Description
		Specified with a character string of at least 1 character and up to 32 characters with printable ASCII characters.
SNMPv3EngineID	string	<p>The ID for identifying the SNMPv3 agent.</p> <p>Specified in one of the following three formats in the adding and updating processes.</p> <ol style="list-style-type: none"> Two hex digits separated with colons(:). Insert 0x at the beginning. (e.g. 0x11:22:33:44:55) Two hex digits separated with colons(:). (e.g. 11:22:33:44:55) Two hex digits separated with spaces. (e.g. 11 22 33 44 55) <p>The 1st format described above is used in the obtaining process.</p> <p>SNMPv3 traps are not received when this property is set to null.</p>
SNMPv3SecurityLevel	string	<p>The security level of the SNMPv3 communication.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> "NoAuth/NoPriv": No authentication and no encryption "Auth/NoPriv": Authentication, but no encryption "Auth/Priv": Authentication and encryption
SNMPv3AuthenticationProtocol	string	<p>The authentication protocol for the SNMPv3 communication.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> "MD5" "SHA1"
SNMPv3AuthenticationPassword ¹⁾	string	<p>The authentication password for the SNMPv3 communication.</p> <p>Specified with a character string of at least 8 characters and up to 255 characters with printable ASCII characters.</p>
SNMPv3PrivacyProtocol	string	<p>The encryption protocol to be used for the SNMPv3 communication.</p> <p>Specified with the following string.</p> <ul style="list-style-type: none"> "DES"
SNMPv3PrivacyPassword ¹⁾	string	<p>The privacy password for the SNMPv3 communication.</p> <p>Specified with a character string of at least 8 characters and up to 255 characters with printable ASCII characters.</p>
SNMPv3SeverityOfInvalidEngineID	string	<p>The alert severity to be published when the EngineID in the received SNMPv3 trap does not match that stored in Network Manager.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> "Warning" "Minor Fault" "Major Fault" "Critical State" <p>When this property is set to null, the alert is not published even if an invalid EngineID is detected.</p>
ApplicationPath	string	<p>The application path of the application that is run when Start Application menu of the target node is selected.</p>

Property Name	JSON Type	Description
		Specified with a character string of up to 4096 characters, and specified with either a absolute path or a relative path.
URL	string	The URL that is accessed when Start Web browser menu of the target node is selected. Specified with a character string of up to 2083 characters.
DiscoveryProtocol	integer	<p>The protocol name to be used for detecting neighboring devices. This protocol decides the process of "discovering physical topology" and "Topology information checking tool".</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • 1 : IP8800/700 series, ES8800/1700 series(OADP) • 2 : Cisco router series, Catalyst series(CDP) • 3 : CX-uH24(OADP) • 4 : IP8800/S300, S400 series(OADP) • 5 : IP8800/R400 series(OADP) • 6 : ProCurve2800(CDP) • 7 : Brocade Communications Systems (former Foundry Networks) device(FDP) • 8 : IP8800/S2400, S3600, S6300, S6700 series(OADP) • 9 : LLDP support device • 100: Other switches • 101: Other routers • 200: Other servers • 201: Other terminals <p>Processed in the same way as 100 (Other switches) is set when null is set in this property.</p>
TelnetServer	string	<p>The identification information whether the node has the function as a telnet server or not.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "On": Telnet server function is available. • "Off": Telnet server function is not available.

Property Name	JSON Type	Description
AgentType	integer	<p>The agent type of the device.</p> <p>Used to identify the device type in the SyslogDiagnosis function and so on. This value is not needed to be consciously specified in the adding and updating processes because this value is automatically decided based on the value of IconType.</p> <p>When null is set in this property, this property is processed in the same way as 0 is set, this indicates a generic agent type, .</p>
AdministrationNodeName	string	<p>The node name of the device administrating the device.</p> <p>Specified with a character string of up to 63 characters.</p> <p>For example, in the case that the target device is a virtual machine, this property specifies the node name of the host device administrating the target virtual machine.</p> <p>In the environment of multi context structure using Cisco ASA5500 series, the value of this property is used when using the configuration administration function.</p>
SysName	string	<p>The device name assigned on the management.</p> <p>Specified with a character string of up to 255 characters.</p> <p>In the adding and updating processes, setting the the same value as that of ".iso.org.dod.internet.mgmt.system.sysName.0" of MIB of the target device when specifying this property manually.</p>
MonitoringMode	string	<p>The state of the monitoring mode.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "On": Monitoring-mode ON • "Off": Monitoring-mode OFF
SysObjectID	string	<p>The value of iso.org.dod.internet.mgmt.mib-2.system.sysObjectID of the device.</p> <p>For reference only.</p>
SysDescr	string	<p>The value of iso.org.dod.internet.mgmt.mib-2.system.sysDescr of the device.</p> <p>For reference only.</p>
FEX_ID	integer	<p>The ID for identifying Nexus2000.</p> <p>Specified with a number from 0 to 4294967295.</p> <p>Managing the fault of Nexus2000 (e.g. receiving SNMP traps and monitoring the status of the interface) by setting the value properly.</p> <p>Each Nexus2000 node is not monitored when this property is set to null.</p>
DPID	string	<p>The Datapath ID of the Openflow-enabled network device.</p> <p>Specified in the range from "0000-0000-0000-0000" to "ffff-ffff-ffff-ffff".</p>
FloatingIP	string	<p>The virtual IPv4 address of the device.</p> <p>In the environment that devices have redundant configuration, the IP address shared with multiple devices is specified with IPv4 address format such as "192.168.0.1".</p>

Property Name	JSON Type	Description
		The process of topology discovery of ProgrammableFlow uses the value of this property. When null is set in this property, the value of IPAddress property is used.
WebAccessPort	integer	<p>The port number to access the web interface of the device.</p> <p>Specified with a number from 0 to 65535 in the adding and updating processes. When specified with the value of outside this range, an error occurs during the operation using this property.</p> <p>The process of topology discovery of ProgrammableFlow uses the value of this property. When null is set in this property, 8080 port is used.</p>

Note

1. Although this API returns node resource, these properties do not return the values by default because these are security related properties. For including security related information in the return values, it needs to explicitly specify the parameter for including it when calling API.

IPv4 Interfaces properties

Property	JSON Type	Description
Index	integer	The index of the interface
MACAddress	string	The MAC address of the interface
Type	string	The type of the interface
Description	string	The description of the interface
FEX_ID	integer	FEX ID
IPv4	object[]	The array of IPv4 property objects associated with the interface.

Tip

All references are for reference only.

IPv4 properties

Property	JSON Type	Description
Address	string	IPv4 address
SubnetMask	string	Subnet mask

Tip

All references are for reference only.

IPv6 Interfaces properties

Property	JSON Type	Description
Index	integer	The index of the interface
MACAddress	string	The MAC address of the interface

Property	JSON Type	Description
Description	string	The description of the interface
LowerLayer	string	The lower layer protocol of the interface
IPv6	object[]	The array of IPv6 property objects associated with the interface.

Tip

All references are for reference only.

IPv6 properties

Property	JSON Type	Description
Address	string	IPv4 address
PrefixLength	integer	Prefix length

Tip

All references are for reference only.

3.3 Map Member Resource (MapMemberObject)

The resource expressing maps and nodes belonging to the map.

Resource format

```
{
  "Type" : "map",
  "ID" : integer,
  "Name" : string,
  "Members" : [ MapMemberObject | NodeMemberObject ] | null
}
```

Properties

Property Name	JSON Type	Description
Type	string	The object type. Specified with "map" that indicates being a map.
ID	integer	The identification of the map. Refer to "3.1 Map Resource (MapObject) (page 16)" for the details.
Name	string	The name of the map. Refer to "3.1 Map Resource (MapObject) (page 16)" for the details.
Members	object[]	The array of maps (MapMemberObject) or nodes (NodeMemberObject) belonging to the target map. An empty array is set when there are no maps and nodes belonging to the map. In addition, this property is set to null when obtaining the information without considering the structure of maps and nodes belonging to the

Property Name	JSON Type	Description
		map (e.g. in the case that "true" is not set in Recursive described in "4.1.14 Obtaining Map Configuration (page 60)").

3.4 Node Status Resource (NodeStatusObject)

The resource expressing the severity(status) of the node.

Resource format

```
{
  "Type" : "node",
  "ID" : integer,
  "Name" : string,
  "Status" : "Normal" | "Unmanaged" | "Unknown" | "Warning" | "Minor" | "Major" | "Fatal"
}
```

Properties

Property Name	JSON Type	Description
Type	string	The object type. Specified with "node" that indicates being a node.
ID	integer	The identification of the node. Refer to "3.2 Node Resource (NodeObject) (page 17)" for the details.
Name	string	The name of the node. Refer to "3.2 Node Resource (NodeObject) (page 17)" for the details.
Status	string	The character string indicating the current severity of the node. Specified with one of the followings. <ul style="list-style-type: none"> "Normal" "Unmanaged" "Unknown" "Warning" "Minor" "Major" "Fatal"

3.5 Node Member Resource (NodeMemberObject)

The resource format to express nodes belonging to the map is described in the following.

Resource format

```
{
  "Type" : "node",
  "ID" : integer,
```

```
"Name" : string
}
```

Properties

Property Name	JSON Type	Description
Type	string	The object type. Specified with "node" that indicates being a map.
ID	integer	The identification of the node. Refer to "3.2 Node Resource (NodeObject) (page 17)" for the details.
Name	string	The name of the node. Refer to "3.2 Node Resource (NodeObject) (page 17)" for the details.

3.6 Status Monitoring Rule Entry Resource (StsMonObject)

The resource format to express the status monitoring rule entry is described in the following.

Resource format

```
{
  "ID" : integer,
  "Rule" : string,
  "Name" : string,
  "Status" : "Running" | "Stopped",
  "Nodes" : string,
  "Interval" : string,
  "AlertSeverity" : "Warning" | "Minor" | "Major" | "Fatal",
  "Arguments" : [ string ]
}
```

Properties

Property Name	JSON Type	Description
ID	integer	The identification of the status monitoring entry. Managed as the invariant value from when this entry is created to when this is deleted, and this is for reference only.
Rule	string	The name of the status monitoring rule. The head part of the listed rules delimited by a colon(:) on State Monitoring window is processed as the name of the status monitoring rule. For example, updown:UpDownCheck rule is described as "updown". Refer to <i>"7.1 Status monitoring Rules"</i> of <i>"MasterScope Network Manager user's manual"</i> for the kinds of status monitoring rules and their details that Network Manager provides by default.
Name	string	The title of the status monitoring rule entry.

Property Name	JSON Type	Description
		The usable characters and their number are not limited in the adding and updating processes, however, it is recommended to specify with an unique name of up to 259 characters for distinguishing from other status monitoring rule entries.
Status	string	<p>The monitoring status of the status monitoring rule entry. Specified with one of the followings.</p> <ul style="list-style-type: none"> • "Running": Monitored. • "Stopped": Not monitored.
Nodes	string	<p>The name of the monitored node (component name). Can be specified in the format described in "A.2 Standard Component Name Specification Format (page 155)" in the adding and updating processes.</p>
Interval	string	<p>The monitoring interval. Specified in the format of number + unit character. The meaning of each unit character is as follows.</p> <ul style="list-style-type: none"> • "s": seconds, "m": minutes, "h": hours <p>Can be specified in the range of the following values in the adding and updating processes.</p> <ul style="list-style-type: none"> • Seconds: 1 to 32767 seconds • Minutes: 1 to 32767 minutes • Hours: 1 to 32767 hours <p>When obtaining the value of this property, the value is converted to other integer using the largest unit regardless of the specified unit in the adding and updating processes. For example, "120s" is converted to "2m" when obtaining the value.</p>
AlertSeverity	string	<p>The severity of the generated alert when a failure occurs. Specified with one of the followings.</p> <ul style="list-style-type: none"> • "Warning": Warning • "Minor": Minor failure • "Major": Major failure • "Fatal": Fatal failure
Arguments	string[]	<p>The array of the variable value. Specified in the following format.</p> <pre>"Arguments" : ["variable1", "variable2", ...]</pre> <p>Example:</p>

Property Name	JSON Type	Description
		<p>When each parameter is set as follows in updown:UpDownCheck rule,</p> <ul style="list-style-type: none"> Maximum response time[msec]: 300 Number of times: 5 Number of times connection failed: 10 <p>this property is set as follows.</p> <pre>"Arguments" : ["300", "5", "10"]</pre> <p>Refer to “7.1 Status Monitoring Rules” of “<i>MasterScope Network Manager user's manual</i>” for the details of variable values of status monitoring rules that Network Manager provides by default.</p>

3.7 Status Monitoring Rule Resource (StsMonRuleObject)

The resource format to express the status monitoring rule is described in the following.

Resource format

```
{
  "Name" : string,
  "Status" : "Attached" | "Detached"
}
```

Properties

Property Name	JSON Type	Description
Name	string	<p>The name of the status monitoring rule.</p> <p>The head part of the listed rules delimited by a colon(:) on State Monitoring window is processed as the name of the status monitoring rule.</p> <p>For example, updown:UpDownCheck rule is described as "updown".</p> <p>Refer to “7.1 Status Monitoring Rules” of “<i>MasterScope Network Manager user's manual</i>” for the kinds of status monitoring rules and their details that MasterScope Network Manager provides by default.</p>
Status	string	<p>The built-in status.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> "Attached": Attached. "Detached": Detached.

3.8 Data Collection Entry Resource (DataColObject)

The resource format to express the data collection entry is described in the following.

Resource format

```
{
  "ID" : integer,
  "Name" : string,
  "ReportType" : "TrafficOfHost" | "TrafficOfHub64bit" | "TrafficOfHub" |
    "TrafficOfWAN64bit" | "TrafficOfWAN" | "ServerLoad" |
    "CPUBusyOfCatalyst" | "MIBExpression" | "General",
  "Status" : "Running" | "Stopped",
  "Nodes" : string,
  "Interval" : string,
  "OperationMode" : "Store" | "Monitor" | "Both",
  "ReportDataOutput" : "On" | "Off" | null,
  "Format" : "Bps" | "Pps" | "Percent" | "Diff" | "Abs" | null,
  "MibExprName" : string | null,
  "Instances" : string | null,
  "MIB" : string | null,
  "RiseOrFall" : "Rise" | "Fall" | null,
  "Threshold" : string | null,
  "AlertSeverity" : "Normal" | "Warning" | "Minor" | "Major" | "Fatal" |
    null,
  "AlertOutputCondition" : "Continuous" | "Total" | null,
  "ContinuousTimes" : integer | null,
  "RecoveryAlert" : "On" | "Off" | null,
  "RecoveryThreshold" : string | null
}
```

Properties

Property Name	JSON Type	Description
ID	integer	The identification of the data collection entry. Managed as the invariant value from when this entry is created to when this is deleted, and this is for reference only.
Name	string	The title of the collection entry. Specified with a character string of at least 1 character and up to 127 characters. Allowed to specify some value only in the adding process, whereas updating process is not allowed. Not allowed to specify the followings in the adding process. <ul style="list-style-type: none"> Specifying a overlapped title. (Half-width alphanumeric characters are case-insensitive.) Specifying surrogate pairs. Specifying symbol characters (i.e. “\ / : ? “ < > and . ”).
ReportType	string	The report type. Specified with one of the followings. <ul style="list-style-type: none"> "TrafficOfHost" :

Property Name	JSON Type	Description
		<p>Traffic of a specific node</p> <ul style="list-style-type: none"> • "TrafficOfHub64bit" : Traffic of a specific port of a hub(64bit) • "TrafficOfHub" : Traffic of a specific port of a hub • "TrafficOfWAN64bit" : Traffic of WAN(64bit) • "TrafficOfWAN" : Traffic of WAN • "ServerLoad" : Load of server • "CPUBusyOfCatalyst" : CPU Busy Rate of Catalyst device • "MIBExpression" : MIB expression • "General" : General purpose <p>Refer to “7.2 Data Collection Rules ”of “<i>MasterScope Network Manager user's manual</i>” for the detailed behavior of each report type.</p> <p>Allowed to specify some value only in the adding process, whereas updating process is not allowed.</p>
Status	string	<p>The collection status of the entry.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "Running": Collecting • "Stopped": Not collecting
Nodes	string	<p>The name of the collection target node(component name).</p> <p>Specified with a character string of at least 1 character and up to 1023 characters.</p> <p>Can be specified in the format described in "A.2 Standard Component Name Specification Format (page 155)" in the adding and updating processes.</p>
Interval	string	<p>The collection interval.</p> <p>Specified in the format of number + unit character.</p> <p>The meaning of each unit character is as follows.</p> <ul style="list-style-type: none"> • "s" : seconds, "m" : minutes, "h" : hours <p>Can be specified in the range of the following values in the adding and updating processes.</p> <ul style="list-style-type: none"> • Seconds: 1 to 999 seconds • Minutes: 1 to 999 minutes • Hours: 1 to 23 hours

Property Name	JSON Type	Description
OperationMode	string	<p>The collection mode.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "Store": Storing data only • "Monitor": Monitoring threshold only • "Both": Both storing data and monitoring threshold
ReportDataOutput	string	<p>The CSV output mode.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "On": Storing both data for graphs and reports • "Off": Storing data for graphs only <p>This property is validated when the value of OperationMode property is set to "Store" or "Both", otherwise, this is set to null.</p>
Format	string	<p>The data format of the output.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "Bps": bps (bits per second) • "Pps": pps (packet per second) • "Percent": % • "Diff": Difference • "Abs": Without data processing <p>This property is validated when the value of ReportType property is set to "General"(General purpose), otherwise, this is set to null.</p>
MibExprName	string	<p>The rule name of the MIB expression.</p> <p>Specified with a character string of at least 1 character and up to 127 characters.</p> <p>This property is validated when the value of ReportType property is set to "MIBExpression"(MIB Expression), otherwise, this is set to null.</p>
Instances	string	<p>The MIB instance.</p> <p>Specified with a character string of up to 1023 characters.</p> <p>Concatenating each instance with a comma when specifying multiple instances in the adding and updating processes.</p> <p>This property is validated when the value of ReportType property is set to one of the followings, otherwise, this is set to null.</p> <ul style="list-style-type: none"> • "TrafficOfHub64bit" : Traffic of a specific port of a hub(64bit) • "TrafficOfHub" : Traffic of a specific port

Property Name	JSON Type	Description
		<ul style="list-style-type: none"> • "TrafficOfWAN64bit" : Traffic of WAN(64bit) • "TrafficOfWAN" : Traffic of WAN • "MIBExpression" : MIB expression
MIB	string	<p>The collection target MIB.</p> <p>Concatenating each MIB with a comma when specifying multiple MIBs in the adding and updating processes. The maximum number of MIB is three.</p> <p>This property is validated when the value of ReportType property is set to "General"(General purpose), otherwise, this is set to null.</p>
RiseOrFall	string	<p>The comparative method of threshold.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "Rise": Detecting a value exceeding threshold • "Fall": Detecting a value falling below threshold <p>The relation between the value of Threshold and RecoveryThreshold needs to meet the following restriction depending on the value of this property.</p> <ul style="list-style-type: none"> • "Rise": Threshold >= RecoveryThreshold • "Fall": Threshold <= RecoveryThreshold <p>This property is validated when the value of OperationMode property is set to "Monitor" or "Both", otherwise, this is set to null.</p>
Threshold	string	<p>The threshold used for threshold monitoring.</p> <p>Specified with a number from 0 to $(2^{64}-1)$.</p> <p>This property is validated when the value of OperationMode property is set to "Monitor" or "Both", otherwise, this is set to null.</p>
AlertSeverity	string	<p>The severity of the alert exceeding threshold.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "Normal" • "Warning" • "Minor" • "Major" • "Fatal" <p>This property is validated when the value of OperationMode property is set to "Monitor" or "Both", otherwise, this is set to null.</p>
AlertOutputCondition	string	<p>The monitoring mode of threshold.</p> <p>Specified with one of the followings.</p> <ul style="list-style-type: none"> • "Continuous": Continuous monitoring • "Total": Cumulative monitoring

Property Name	JSON Type	Description
		This property is validated when the value of OperationMode property is set to "Monitor" or "Both", otherwise, this is set to null.
ContinuousTimes	integer	The upper limit number of the continuous monitoring times. Specified with a number from 1 to 10,000. This property is validated when the value of AlertOutputCondition property is set to "Continuous", otherwise, this is set to null.
RecoveryAlert	string	The presence or absence of the recovery alert output. Specified with one of the followings. <ul style="list-style-type: none"> "On": Outputting recovery alert. "Off": Not outputting recovery alert. This property is validate when the value of RecoveryAlert property is set to "Continuous", otherwise, this is set to null.
RecoveryThreshold	string	The recovery threshold. Specified with a number from 0 to $(2^{64}-1)$. This property is validated when the value of RecoveryAlert property is set to "On", otherwise, this is set to null.

3.9 Alert Resource (AlertObject)

The resource format of alerts are described in the following.

Resource format

```
{
  "ID" : integer,
  "Type" : "System" | "SNMPTrap" | "SystemLog" |
           "EventLog" | "Warning" | "Minor" |
           "Major" | "Critical" | "Recovered" | "Other" | "AllClear" ,
  "Sender" : string | null,
  "Severity" : "Normal" | "Warning" | "Minor" | "Major" |
              "Fatal" | "Unknown",
  "RecoveryStatus" : "NeedNotRecover" | "NeedRecover" |
                    "Recovered" | "AutoRecover",
  "ConfirmStatus" : "Unconfirmed" | "Confirmed",
  "ComponentType" : string,
  "ComponentName" : string,
  "IPAddress" : string | null,
  "OccurredTime" : string,
  "RecoveryTime" : string | null,
  "SummaryMessage" : string | null,
  "DetailMessage" : string | null,
  "ActionMessage" : string | null,
  "SNMPVersion" : string | null,
  "SNMPCommunity" : "1/2c" | "3" | null,
  "AgentAddress" : string | null,
  "EnterpriseOID" : string | null,
  "GenericCode" : integer | null,
  "SpecificCode" : integer | null,
}
```

```
"TimeStamp" : integer | null
}
```

Properties

Only RecoveryStatus property and ConfirmStatus property can be updated.

Property	JSON Type	Description	Searching with value
ID	integer	The ID to specify the alert uniquely. Specified with a number from 1 to 4294967295.	Y
Type	string	The alert type. One of the followings is set. <ul style="list-style-type: none"> "System": System "SNMPTrap": SNMP trap "SystemLog": System log "EventLog": Event log "Warning": Warning "Minor": Minor fault "Major": Major fault "Critical": Critical fault "Recovered": Recovered "AllClear": All clear "Other": Others 	Y
Sender	string	The alert sender. Specified with a character string of up to 63 characters. Null is set when Type property is set to "SNMPTrap".	Y ²⁾
Severity	string	The alert severity. One of the followings is set. <ul style="list-style-type: none"> "Fatal ": Fatal "Major" : Major "Minor" : Minor "Warning" : Warning "Unknown" : Unknown "Normal" : Normal 	Y
RecoveryStatus	string	The recovery status of the alert. One of the followings is set. <ul style="list-style-type: none"> "NeedRecover": Need recovery "Recovered" : Already recovered "AutoRecover" : Auto recovery "NeedNotRecover": Need not recovery 	Y
ConfirmStatus	string	The confirmation status of the alert. Either of the followings is set. <ul style="list-style-type: none"> "Confirmed": Confirmation status is ON. "Unconfirmed": Confirmation status is OFF. 	Y
ComponentType	string	The component type of the alert origin.	Y

Property	JSON Type	Description	Searching with value
		One of the followings is set. <ul style="list-style-type: none"> • Node : "node" • Interface : "if" • System event : "system" 	
ComponentName	string	The component name of the alert origin. Specified with a character string of at least 1 character and up to 63 characters.	Y ²⁾
IPAddress	string	The IP address of the component of the alert origin. Specified in IPv4 format or IPv6 format.	Y ²⁾
OccurredTime	string	The occurrence time of the alert. Specified with the 3rd or 4th format described in " Table 2-6 Dateformat (page 11) " (including timezone information).	Y
RecoveryTime ¹⁾	string	The recovery time of the alert. Specified with the 3rd or 4th format described in " Table 2-6 Dateformat (page 11) " (including timezone information).	N
SummaryMessage	string	The summary message of the alert. Specified with a character string of up to 128 characters. Null is set when this property is empty (none is set).	Y ²⁾
DetailMessage ¹⁾	string	The detail message of the alert. Specified with a character string of up to 2000 characters. Null is set when this property is empty (none is set).	N
ActionMessage ¹⁾	string	The action message for the alert. Specified with a character string up to 1280 characters. Null is set when this property is empty (none is set).	N
SNMPVersion ¹⁾	string	The SNMP version of the SNMP trap. <ul style="list-style-type: none"> • 1/2c: Version 1 or Version 2C • 3: Version 3 Null is set when some value other than "SNMP" is set in Type property.	N
SNMPCommunity ¹⁾	string	The SNMP community name of the SNMP trap. Specified with a character string of up to 255 characters. Null is set when some value other than "SNMP Trap" is set in Type property.	Y ²⁾
AgentAddress ¹⁾	string	The agent address of the SNMP trap(IPv4 address format). Null is set when some value other than "SNMP Trap" is set in Type property.	Y
EnterpriseOID ¹⁾	string	The vendor identification of the SNMP trap. Specified with a character string of up to 255 characters. Specified with the combination of numbers and dots(.). Null is set when some value other than "SNMP Trap" is set in Type property.	Y ²⁾
GenericCode ¹⁾	integer	The generic trap code of the SNMP trap. Specified with a number from 0 to 6.	Y

Property	JSON Type	Description	Searching with value
		Null is set when some value other than "SNMP Trap" is set in Type property.	
SpecificCode ¹⁾	integer	The specific trap code of the SNMP trap. Specified with a number from -2147483648 to 2147483647. Null is set when some value other than "SNMP Trap" is set.	Y
TimeStamp ¹⁾	integer	The sysUptime of the SNMP trap. Specified with a number from 0 to 4294967295. Null is set when some value other than "SNMP Trap" is set in Type property.	N

Note

1. The API returning multiple alert resources as its return value does not return above property information by default. Refer to "[4.4.1 Obtaining Alert List \(page 79\)](#)" for the details.
2. It is possible to search using "[Appendix B. Specifying wildcards \(page 157\)](#)".

3.10 Notes of Caution in Operating Resource

- Null is basically returned when an unset property is referred. An empty character string is returned when the string type property that an empty character string is set is referred. Explicitly distinguish between null and an empty character string in using WebAPI on the client machine.
- When the property name of the resource is wrong in the requesting, MasterScope Network Manager receiving it does not judge it as an error and continues to process it as an invalid property name. Be careful not to miss the property name of the resource in the adding and updating processes because an unintended process may be executed.
- There is possibility that the kinds of values of properties and resource properties are added without prior notice in the future. In addition, API's major version and minor version are not changed in the following cases.
 - In the case that an optional property name is added.
 - In the case that the kind of value of property is added.

Considering the above points, it is recommended to use WebAPI on the client machine on the premise that an unknown property and value of property may be returned.

Chapter 4.

WebAPI Reference

This chapter describes the detailed specification of each API.

Contents

4.1 Configuration Information Management API	40
4.2 Status Monitoring Setting API	63
4.3 Data Collection Setting API	68
4.4 Alert Management API	79

4.1 Configuration Information Management API

This section describes the API specification for operating configuration information of maps and nodes.

4.1.1 Obtaining Map List

Obtaining the list of map information.

It is possible to narrow down the results of obtaining maps by specifying query parameters using "[A.1 Standard Matching Specification Format \(page 155\)](#)".

Request format

```
GET /umf/fw/nvp/maps
```

- API specific query parameter

Parameter	Omission	Description
NameFilter	Y	<p>The parameter to narrow down the results of obtaining the map list by the map name.</p> <p>Tip</p> <p>"A.1 Standard Matching Specification Format (page 155)" can be used as specification format.</p>

- Body
- None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the list succeeded.
400	Failure	The format of " A.1 Standard Matching Specification Format (page 155) " that is specified in NameFilter query parameter is invalid.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

```
[
  MapObject
]
```

Returning the values of only ID and Name.

Refer to "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" for the details of each property.

Caution

It is not allowed to obtain the property information of **NetworkView** and **NetworkManagement**.

4.1.2 Obtaining Map Details

Obtaining the detailed property information of the specified map.

Request format

```
GET /umf/fw/nvp/maps/{map_id}
```

Specifying the value of ID of "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" in `{map_id}`.

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the property information of the map succeeded.
404	Failure	The specified map is not found.
405	Failure	0 is set to <code>{map_id}</code> in the URL.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

```
MapObject
```

Refer to "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" for the details of each property.

Caution

It is not allowed to obtain the property information of **NetworkView** and **NetworkManagement**.

4.1.3 Adding Map

Adding a new map.

Request format

```
POST /umf/fw/nvp/maps
```

- API specific query parameter

Parameter	Omission	Description
UpperMap	Y	The parameter for specifying the adding destination map of the new map.

Parameter	Omission	Description
		Specifying the ID of the adding destination map. A new map is created on NetworkManagement when setting the value of this parameter is omitted or 0 is set in this parameter.

- Body

MapObject

Refer to "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" for the details of each property.

- The details of specified map resource

Property	Omission	Description
ID	Y	Ignored even when some value is set in this property.
Name	N	Not allowed to set to null.
IconType	Y	"map" is set when setting the value of this property is omitted. Not allowed to set to null.
Others	Y	Null is set when setting the value of this property is omitted.

Response format

- Status code

code	Meaning	Description
200	Success	The map is added successfully.
400	Failure	Either of the following errors occurs. <ul style="list-style-type: none"> • The format of the query parameter is invalid. • The format of the map resource specified in the requesting is invalid.
404	Failure	The specified map in UpperMap query parameter is not found.
409	Failure	The name of the new map is duplicated with that of the existing map.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

MapObject

Returning values of all properties including ID and omitted properties.

Refer to "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" for the details of each property.

4.1.4 Updating Map

Updating the property information of the specified map.

Request format

```
PUT /umf/fw/nvp/maps/{map_id}
```


Specifying the value of ID of "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" in `{map_id}`.

- API specific query parameter

None

- Body

MapObject

Refer to "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" for the details of each property.

- Details of specified map resource

Property	Omission	Description
ID	Y	When some value is set in this property, it needs to match the value with <code>{map_id}</code> in the URL.
Name	Y	Not allowed to set to null.
IconType	Y	
Others	Y	-

Omitted properties are not updated. Specified properties are only updated.

Response format

- Status code

code	Meaning	Description
200	Success	The map is updated successfully.
400	Failure	Either of the following errors occurs. <ul style="list-style-type: none"> • The format of the map resource in the requesting is invalid. • The <code>{map_id}</code> in the URL and the ID specified in the message body are not matched.
404	Failure	The specified map is not found.
405	Failure	The <code>{map_id}</code> in the URL is set to 0.
409	Failure	The updated map name is duplicated with that of the existing map.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

MapObject

Returning values of all properties including not updated properties.

Refer to "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" for the details of each property.

Caution

It is not allowed to update **NetworkView** and **NetworkManagement**.

4.1.5 Deleting Map

Deleting the specified map. All icons of maps and nodes belonging to the specified map are recursively deleted.

Request format

```
DELETE /umf/fw/nvp/maps/{map_id}
```

Specifying the value of ID of "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" in `{map_id}`.

- API specific query parameter
none
- Body
none

Response format

- Status code

code	Meaning	Description
200	Success	The map is deleted successfully.
404	Failure	The delete target map is not found.
405	Failure	The <code>{map_id}</code> in the URL is set to 0.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body
none

Caution

It is not allowed to delete **NetworkView** and **NetworkManagement**.

4.1.6 Obtaining Node List

Obtaining the list of node information.

It is possible to narrow down the results of obtaining nodes by specifying query parameters using "[A.1 Standard Matching Specification Format \(page 155\)](#)".

Request format

```
GET /umf/fw/nvp/nodes
```

- API specific query parameters

Parameter	Omission	Description
NameFilter	Y	The parameter to narrow down the results of obtaining the node list by the node name.

Parameter	Omission	Description
		Tip <ul style="list-style-type: none"> "A.1 Standard Matching Specification Format (page 155)" can be used as specification format. It is not allowed to specify this parameter together with MapFilter query parameter.
MapFilter	Y	<p>The parameter to narrow down the results of obtaining the node list by the map ID.</p> <p>Targeting nodes belonging to the specified map for obtaining. Maps belonging to the specified map are recursively searched.</p> <p>All nodes belonging to NetworkManagement become targets for obtaining when setting the value of this parameter is omitted or 0 is set in this parameter.</p> Tip <p>It is not allowed to specify this parameter together with NameFilter query parameter.</p>

- Body

None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the list succeeded.
400	Failure	<p>Either of the following errors occurs.</p> <ul style="list-style-type: none"> The format of "A.1 Standard Matching Specification Format (page 155)" that is specified in NameFilter query parameter is invalid. NameFilter query parameter and MapFilter query parameter are specified together.
404	Failure	The specified map in MapFilter query parameter is not found.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

```
[
  NodeObject
]
```

Returning the values of only ID and Name.

Refer to "[3.2 Node Resource \(NodeObject\) \(page 17\)](#)" for the details of each property.

Caution

When multiple icons are assigned for one node, each property is returned together as one node's property. In this case, when the value of **Icon Type** of each icon's property is different, the value of one of these **Icon Type** is set in IconType of the returned node resource.

4.1.7 Obtaining Node Details

Obtaining the property of the node.

Request format

```
GET /umf/fw/nvp/nodes/{node_id}
```

Specifying the value of ID of "[3.2 Node Resource \(NodeObject\) \(page 17\)](#)" in `{node_id}`.

- API specific query parameter

Parameter	Omission	Description
Detail	Y	<p>The parameter to specify the range of the returned properties.</p> <ul style="list-style-type: none"> • None is specified Returning properties except for security related information. • full Returning all properties including security related information. • Others Processed in the same way as none is specified. <p>Refer to "1. (page 25)" of "3.2 Node Resource (NodeObject) (page 17)" for the security related information.</p> <p>Tip</p> <p>Specifying values in ShowInterfaces query parameter to return interface related properties.</p>
ShowInterfaces	Y	<p>The parameter to specify whether returning interface related properties or not.</p> <ul style="list-style-type: none"> • true Returning interface related properties. • false Not returning interface related properties. • None is specified Processed in the same way as true is set. • Others Processed in the same way as true is set.

- Body

None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the property succeeded.
404	Failure	The node with the specified ID is not found.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

NodeObject

The contents of the returned property are differed depending on the specified values in query parameters. Refer to the explanation of each query parameter for the details.

Refer to "[3.2 Node Resource \(NodeObject\) \(page 17\)](#)" for the details of each property.

Caution

When multiple icons are assigned for one node and different values are set in their **Icon Type**, the value of one of these **Icon Type** is set in IconType of the returned node resource.

4.1.8 Adding Node

Adding a new node.

Request format

POST /umf/fw/nvp/nodes

- API specific query parameters

Parameter	Omission	Description
UpperMap	Y	The parameter to specify the adding destination map of the new map. Specifying the ID of the adding destination map. The icon of the new node is created on NetworkManagement when setting the value of this parameter is omitted or 0 is set in this parameter.
MultipleIcon	Y	<p>The parameter to specify whether handling the following case as an error or not. The case is that the node name of the new node is same with that of other node existing on some map other than the specified map in UpperMap query parameter.</p> <ul style="list-style-type: none"> • true Creating an icon of the node on the specified map without handling the adding process as an error. In this case, the property of the new node is updated by the contents of property of the existing node whose node name is same as that of the new node. The way of updating the property information is the same when "4.1.9 Updating Node (page 51)" is executed. • false Handling the adding node process as an error. • None is specified Processed in the same way as false is set. • Others Processed in the same way as false is set. <p>Tip</p> <p>Processed as an error even when true is set in this query parameter in the case that the node with the same node name is</p>

Parameter	Omission	Description
		already located on the map specified in UpperMap query parameter.
Detail	Y	<p>The parameter to specify the range of properties returned after adding the node.</p> <ul style="list-style-type: none"> • None is specified Returning properties except for security related information. • full Returning all properties including security related information. • Others Processed in the same way as none is specified. <p>Refer to "1. (page 25)" of "3.2 Node Resource (NodeObject) (page 17)" for the security related information.</p> <p>Tip</p> <p>Specifying values in ShowInterfaces query parameter to return interface related properties.</p>
ShowInterfaces	Y	<p>The parameter to specify whether returning interface related properties or not after adding the node.</p> <ul style="list-style-type: none"> • true Returning interface related properties. • false Not returning interface related properties. • None is specified Processed in the same way as true is set. • Others Processed in the same way as true is set.

- Body

NodeObject

Refer to "3.2 Node Resource (NodeObject) (page 17)" for the details of each property.

- Details of specified node resource

Property	Omission	Description
ID	Y	Ignored even when some value is set in this property.
Name	N	Not allowed to set null.
IconType	Y	<p>"host" is set when setting the value of this property is omitted. Not allowed to set null.</p> <p>Tip</p> <p>When setting the values of the following properties is omitted or null is set in these properties, proper values are automatically decided based on the value of this property.</p> <ul style="list-style-type: none"> • HardwareType • OSType

Property	Omission	Description
		<ul style="list-style-type: none"> • AgentType • DeviceFrontPanel • DiscoveryProtocol • TelnetServer
SNMPVersion	Y	<p>Processed in the same way as version 1 is specified in SNMP communications when null is set in this property.</p> <p>Tip</p> <p>It may be required to specify the following properties when 3 is set in this property. Refer to the explanation of each property for the details.</p> <ul style="list-style-type: none"> • SNMPv3UserName • SNMPv3SecurityLevel • SNMPv3AuthenticationProtocol • SNMPv3AuthenticationPassword • SNMPv3PrivacyProtocol • SNMPv3PrivacyPassword
HardwareType	Y	<p>The values of these properties are automatically decided based on the value of Icon Type specified in IconType when setting the value of this property is omitted or null is set in this parameter.</p> <p>Tip</p> <p>Null may be set in this property depending on the value of Icon Type.</p>
OSType	Y	
AgentType	Y	
RoutingControl	Y	Processed in the same way as "not forwarding" is set when setting the value of this property is omitted or null is set in this parameter.
Group	Y	<p>Null is set when setting the value of this parameter is omitted.</p> <p>When the following symbolic characters are included in the specified character string, the symbolic characters are removed.</p> <ul style="list-style-type: none"> • Half-width space • Comma(,) used as delimiter at the top or the end of the character string <p>When commas(,) used as delimiter are consecutively set, they are replaced to one comma.</p>
DeviceFrontPanel	Y	<p>The value of this property is automatically decided based on the value of Icon Type specified in IconType when setting the value of this property is omitted or null is set in this property.</p> <p>Tip</p> <p>Null may be set in this property depending on the value of Icon Type.</p>
IPv4Interfaces	Unnecessary	Ignored even when some value is set in these properties.
IPv6Interfaces	Unnecessary	

Property	Omission	Description
SysObjectID	Unnecessary	
SysDescr	Unnecessary	
SNMPv3UserName	Y/N	Required to specify when "3" is set in SNMPVersion3. Not allowed to set to null.
SNMPv3SecurityLevel	Y	Not allowed to set to null. "NoAuth/NoPriv"(without authentication and encryption) is set when setting the value of this property is omitted.
SNMPv3AuthenticationProtocol	Y/N	Required to specify when "3" is set in SNMPVersion and "Auth/NoPriv"(with authentication and without encryption) or "Auth/Priv"(with authentication and encryption) is set in SNMPv3SecurityLevel. Not allowed to set to null.
SNMPv3AuthenticationPassword	Y/N	
SNMPv3PrivacyProtocol	Y/N	Required to specify when "3" is set in SNMPVersion and "Auth/Priv"(with authentication and encryption) is set in SNMPv3SecurityLevel. Not allowed to set to null.
SNMPv3PrivacyPassword	Y/N	
DiscoveryProtocol	Y	<p>The value of this property is automatically decided based on the value of Icon Type specified in IconType when setting the value of this property is omitted or null is set in this property.</p> <p>Tip</p> <p>Null may be set in this property depending on the value of Icon Type.</p>
TelnetServer	Y	<p>The value of this property is automatically decided based on the value of Icon Type specified in IconType when setting the value of this property is omitted.</p> <p>Not allowed to set to null.</p> <p>Tip</p> <p>"Off" may be set in this property depending on the value of Icon Type.</p>
MonitoringMode	Y	"Off" is set when setting the value of this property is omitted. Not allowed to set to null.
Others	Y	Null is set when setting the value of this property is omitted.

Response format

- Status code

code	Meaning	Description
200	Success	The node is added successfully.
400	Failure	<p>One of the following errors occurs.</p> <ul style="list-style-type: none"> • The format of the query parameter is invalid. • The format of the node resource specified in the requesting is invalid.

code	Meaning	Description
		<ul style="list-style-type: none"> Required properties are not specified or null is set in the properties when "3" is set in SNMPVersion.
404	Failure	The specified map in UpperMap query parameter is not found.
409	Failure	One of the following errors occurs. <ul style="list-style-type: none"> The node with the specified node name already exists when some value other than "true" is set in MultipleIcon query parameter. The node with the specified node name already exists on the map specified in UpperMap query parameter. The number of the device license is in short.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

NodeObject

Returning values of all properties including ID and omitted properties.

Note that a part of properties can not be returned depending on the specified contents in Detail and ShowInterfaces query parameters. Refer to the explanation of each query parameter for the details.

Refer to ["3.2 Node Resource \(NodeObject\) \(page 17\)"](#) for the details of each property.

4.1.9 Updating Node

Updating the property information of existing nodes.

Request format

```
PUT /umf/fw/nvp/nodes/{node_id}
```

Specifying the value of ID of ["3.2 Node Resource \(NodeObject\) \(page 17\)"](#) in {node_id}.

- API specific query parameters

Parameter	Omission	Description
Detail	Y	The parameter to specify the range of the returned properties after updating. <ul style="list-style-type: none"> None is specified Returning properties except for security related information. full Returning all properties including security related information. Others Processed in the same way as none is specified. Refer to "1. (page 25)" of "3.2 Node Resource (NodeObject) (page 17)" for the security related information.

Parameter	Omission	Description
		Tip Specifying values in ShowInterfaces query parameter to return interface related properties.
ShowInterfaces	Y	The parameter to specify whether returning interface related properties or not after updating. <ul style="list-style-type: none"> • true Returning interface related properties. • false Not returning interface related properties. • None is specified Processed in the same way as true is set. • Others Processed in the same way as true is set.

- Body

NodeObject

Refer to "[3.2 Node Resource \(NodeObject\) \(page 17\)](#)" for the details of each property.

- The details of specified node resource

Property	Omission	Description
ID	Y	When some value is set in this property, it needs to match the value with {node_id} in the URL .
Name	Y	Not allowed to set to null.
IconType	Y	Not allowed to set to null. Tip The values of the following properties are automatically decided based on the specified value in this property when setting the values of these properties are omitted or null is set in these properties. <ul style="list-style-type: none"> • HardwareType • OStype • AgentType • DeviceFrontPanel • DiscoveryProtocol • TelnetServer
SNMPVersion	Y	Processed in the same way as version 1 is specified in SNMP communications when null is set in this property. Tip It may be required to specify the following properties when 3 is set in this property. Refer to the explanation of each property for the details. <ul style="list-style-type: none"> • SNMPv3UserName

Property	Omission	Description
		<ul style="list-style-type: none"> • SNMPv3SecurityLevel • SNMPv3AuthenticationProtocol • SNMPv3AuthenticationPassword • SNMPv3PrivacyProtocol • SNMPv3PrivacyPassword
HardwareType	Y	<p>The values of these properties are automatically decided based on the value of Icon Type specified in IconType when some value is set in IconType and setting the values of these properties are omitted or null is set in these properties.</p> <p>The values of these properties are automatically decided based on the value of already set Icon Type when setting the value of IconType is omitted and null is set in this property.</p> <p>Tip</p> <p>Null may be set in this property depending on the value of Icon Type.</p>
OSType	Y	
AgentType	Y	
Group	Y	<p>When the following symbolic characters are included in the specified character string, the symbolic characters are removed.</p> <ul style="list-style-type: none"> • Half-width space • Comma(,) used as delimiter at the top or the end of the character string <p>When commas(,) used as delimiter are consecutively set, they are replaced to one comma.</p>
DeviceFrontPanel	Y	<p>The value of this property is automatically decided based on the value of Icon Type specified in IconType when some value is set in IconType and setting the value of this property is omitted or null is set in this property.</p> <p>The value of this property is automatically decided based on the value of already set Icon Type when setting the value of IconType is omitted and null is set in this property.</p> <p>Tip</p> <p>Null may be set in this property depending on the value of Icon Type.</p>
IPv4Interfaces	Y	Ignored even when some value is set in these properties.
IPv6Interfaces	Y	
SysObjectID	Y	
SysDescr	Y	
SNMPv3UserName	Y/N	Required to specify when "3" is set in SNMPVersion.
SNMPv3SecurityLevel	Y/N	Not allowed to set to null.
SNMPv3AuthenticationProtocol	Y/N	Required to specify when "3" is set in SNMPVersion and "Auth/NoPriv"(with authentication and without encryption) or "Auth/Priv"(with authentication and encryption) is set in SNMPv3SecurityLevel.
SNMPv3AuthenticationPassword	Y/N	Not allowed to set to null.

Property	Omission	Description
SNMPv3PrivacyProtocol	Y/N	Required to specify when "3" is set in SNMPVersion and "Auth/Priv"(with authentication and encryption) is set in SNMPv3SecurityLevel.
SNMPv3PrivacyPassword	Y/N	Not allowed to set to null.
DiscoveryProtocol	Y	<p>The value of this property is automatically decided based on the value of Icon Type specified in IconType when some value is set in IconType and setting the value of this property is omitted or null is set in this property.</p> <p>The value of this property is automatically decided based on the value of already set Icon Type when setting the value of IconType is omitted and null is set in this property.</p> <p>Tip</p> <p>Null may be set in this property depending on the value of Icon Type.</p>
TelnetServer	Y	<p>The value of this property is automatically decided based on the value of Icon Type specified in IconType when some value is set in IconType and setting the value of this property is omitted.</p> <p>Not allowed to set to null.</p> <p>Tip</p> <p>"Off" may be set in this property depending on the value of Icon Type.</p>
MonitoringMode	Y	Not allowed to set to null.
Others	Y	-

Omitted properties are not updated. Specified properties are only updated.

Note that the values of HardwareType, OSType, AgentType, DeviceFrontPanel, DiscoveryProtocol and TelnetServer can be updated depending on the value of **Icon Type**. Refer to the explanation of each property for the details.

Response format

- Status code

code	Meaning	Description
200	Success	The property of the node is updated successfully.
400	Failure	<p>One of the following errors occurs.</p> <ul style="list-style-type: none"> • The format of the node resource specified in the requesting is invalid. • The <code>{node_id}</code> in the URL and the ID specified in the message body are not matched. • Required properties are not specified or null is set in the properties when "3" is set in SNMPVersion3.
404	Failure	The node with the specified ID is not found.
409	Failure	The node with the specified Name already exists in the name updating.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

NodeObject

The values of all properties including ID and omitted properties are returned.

Note that a part of properties may not be returned depending on the specified contents of Detail and ShowInterfaces query parameters. Refer to the explanation of each query parameter for the details.

Refer to ["3.2 Node Resource \(NodeObject\) \(page 17\)"](#) for the details of each property.

Caution

In the environment that icons of the same node are located in multiple maps, when updating the value of the property of one of the nodes using this API, all properties of the node are updated.

4.1.10 Deleting Node

Deleting the information of existing nodes.

The following two patterns of delete operation can be executed in the environment that icons of the same type of node are located in multiple maps.

1. Deleting the icon of the specified node on the specified map.
2. Deleting all icons of the specified node.

Tip

When only one icon is registered in one node, the above processes of ["1. \(page 55\)"](#) and ["2. \(page 55\)"](#) become the same.

Request format

```
DELETE /umf/fw/nvp/nodes/{node_id}
```

Specifying the value of ID of ["3.2 Node Resource \(NodeObject\) \(page 17\)"](#) in {node_id}.

- API specific query parameters

Parameter	Omission	Description
All	Y	<p>The parameter to specify the way to delete in the environment that icons of the same type of node are located in multiple maps.</p> <ul style="list-style-type: none"> • None is specified Deleting only the icon existing the specified map. (Process of "1. (page 55)") • true Deleting all icons of the specified node. (Process of "2. (page 55)") • Others Processed in the same way as none is specified.
UpperMap	Y	<p>The parameter to specify the map registered the deleted node.</p> <p>Specifying the ID of the map that the icon of the deleted node exists. 0 is set when specifying NetworkManagement.</p>

Parameter	Omission	Description
		Required to specify this query parameter when none is set or some value other than "true" is set in All query parameter.

- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	The node is deleted successfully.
400	Failure	None is set in All query parameter, or some value other than "true" is set in All query parameter on the other hand none is set in UpperMap query parameter.
404	Failure	One of the following errors occurs. <ul style="list-style-type: none"> • The node with the specified ID does not exist. • The map specified in UpperMap query parameter does not exist. • The specified node does not exist on the map specified in UpperMap query parameter.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body
None

4.1.11 Updating Device Information of Node

Updating properties of the registered node by obtaining device information using SNMP.

Properties to be updated are as follows.

Categories	Device information	Interface information
Required properties	<ul style="list-style-type: none"> • Routing control information (RoutingControl) • Software version (SoftwareVersion) • SNMP engine ID (SNMPv3EngineID) • Agent Type (AgentType) • sysName (SysName) • sysObjectID (SysObjectID) • sysDescr (SysDescr) 	<ul style="list-style-type: none"> • IPv4 interface information (IPv4Interfaces, IPv4) • IPv6 interface information (IPv6Interfaces, IPv6) • FexID (FEX_ID)
Others	<ul style="list-style-type: none"> • OS type 	<ul style="list-style-type: none"> • Default target port

Categories	Device information	Interface information
	(OSType) • Administrator (Administrator) • Location (Location)	(DefaultTargetPort)

Request format

POST /umf/fw/nvp/nodes/{node_id}/update

Specifying the value of node resource ID in {node_id}.

- API specific query parameters

Parameter	Omission	Description
Action	Y	<p>The parameter to specify the range of the updated properties.</p> <ul style="list-style-type: none"> • None is specified Updating only the required properties of device information and interface information. • all Updating all properties of device information and interface information. • interface Updating only the required properties of interface information. • interface-all Updating all properties of interface information. • Others Processed in the same way as none is specified.
Detail	Y	<p>The parameter to specify the range of the returned properties after updating.</p> <ul style="list-style-type: none"> • None is specified Returning properties except for security related information. • full Returning all properties including security related information. • Others Processed in the same way as none is specified. <p>Refer to "1. (page 25)" of "3.2 Node Resource (NodeObject) (page 17)" for the security related information.</p> <p>Tip</p> <p>Specifying values in ShowInterfaces query parameter to return interface related properties.</p>
ShowInterfaces	Y	<p>The parameter to specify whether returning interface related properties or not after updating.</p> <ul style="list-style-type: none"> • true Returning interface related properties. • false Not returning interface related properties.

Parameter	Omission	Description
		<ul style="list-style-type: none"> None is specified Processed in the same way as true is set. Others Processed in the same way as true is set.

- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	The properties of the node is updated successfully.
404	Failure	The node with the specified ID is not found.
409	Failure	The required properties (e.g. IP address and SNMP information) for updating device information are not set in the node.
500	Failure	The SNMP communication related error occurs.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

NodeObject

Returning values of all properties.

Note that a part of properties may not be returned depending on the specified contents in Detail and ShowInterfaces query parameters. Refer to the explanation of each query parameter for the details.

Refer to "[3.2 Node Resource \(NodeObject\) \(page 17\)](#)" for the details of each property.

4.1.12 Obtaining Severity Status List of Node

Obtaining the list of the severity status information (status of alert generation) of nodes.

It is possible to obtain the list of information of nodes generating alerts that whose current severity are higher than those of the specified severity by specifying the severity in the query parameter.

Request format

```
GET /umf/fw/nvp/nodes/status
```

- API specific query parameters

Parameter	Omission	Description
Severity	Y	The parameter to specify the severity.

Parameter	Omission	Description
		<p>Obtaining the list of nodes whose current severities are higher than the specified severity.</p> <p>Specified with one of the following severity character strings.</p> <ul style="list-style-type: none"> • Normal • Unmanaged • Unknown • Warning • Minor • Major • Fatal <p>All severity levels become target for obtaining when setting the value of this parameter is omitted.</p>
MapFilter	Y	<p>The parameter to narrow down the target nodes.</p> <p>All nodes belonging to the specified map become the target for obtaining by setting the ID of the map resource. In addition, when there are some maps belonging to the specified map, these maps are recursively searched.</p> <p>All nodes belonging to NetworkManagement become the target for obtaining when setting the value of this parameter is omitted or 0 is set in this parameter.</p>

- Body
- None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the list of the severity status of nodes succeeded.
400	Failure	The format of the query parameter specified in the requesting is invalid.
404	Failure	The map specified in MapFilter query parameter is not found.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

```
[
  NodeStatusObject
]
```

Refer to "[3.4 Node Status Resource \(NodeStatusObject\) \(page 27\)](#)" for the details of each property.

Caution

The returned severity information is depended on the priority of the severity setting when some value is set in Severity query parameter.

4.1.13 Obtaining Severity Status of Node

Obtaining the severity status information (status of alert generation) of the specified node.

Request format

```
GET /umf/fw/nvp/nodes/{node_id}/status
```

Specifying the value of ID of "[3.2 Node Resource \(NodeObject\) \(page 17\)](#)" in `{node_id}`.

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the severity status of the node succeeded.
404	Failure	The node with the specified ID is not found.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that can be used.

- Body

```
NodeStatusObject
```

Refer to "[3.4 Node Status Resource \(NodeStatusObject\) \(page 27\)](#)" for the details of each property.

4.1.14 Obtaining Map Configuration

Obtaining the list of the information of maps and nodes belonging to the specified map.

Request format

```
GET /umf/fw/nvp/maps/{map_id}/members
```

Specifying the value of ID of "[3.1 Map Resource \(MapObject\) \(page 16\)](#)" in `{map_id}`.

Setting 0 in `{map_id}` to specify **NetworkManagement** that is the top level map.

- API specific query parameters

Parameter	Omission	Description
Type	Y	<p>The parameter to reduce the kinds of obtaining components.</p> <ul style="list-style-type: none"> • None is specified Obtaining both information of maps and nodes. • map Obtaining only map information.

Parameter	Omission	Description
		<ul style="list-style-type: none"> Others Processed in the same way as none is specified.
Recursive	Y	The parameter to specify whether recursively searching maps belonging to the specified map or not. <ul style="list-style-type: none"> None is specified Searching only the specified map. <ul style="list-style-type: none"> true Searching all maps recursively including maps belonging to the specified map. <ul style="list-style-type: none"> Others Processed in the same way as none is specified. Tip <hr/> When some value other than true is set in this parameter, the value of Members of " 3.3 Map Member Resource (MapMemberObject) (page 26) " in the response message body always becomes null regardless of the existence of nodes and maps belonging to the specified map. <hr/>

- Body
- None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining of the list succeeded.
404	Failure	The specified map is not found.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

```
[
  MapMemberObject or NodeMemberObject
]
```

Returning the JSON data expressing tree structure.

Refer to "[3.3 Map Member Resource \(MapMemberObject\) \(page 26\)](#)" and "[3.5 Node Member Resource \(NodeMemberObject\) \(page 27\)](#)" for the details of each property.

Caution

- When there are multiple icons for one node on the map, their information is returned into one object.
- The information of the specified map itself is not returned.

4.1.15 Expanding Standard Component Name Specification Format

Obtaining the list of components or group members specified in the standard component name format.

This API is used to confirm the concrete node name of the monitoring target when the monitoring target of the status monitoring rule and the data collection entry is specified in "[A.2 Standard Component Name Specification Format \(page 155\)](#)" or with a group.

The component is returned with excluding overlapped information.

Request format

```
GET /umf/fw/nvp/nodes/std-expand
```

- API specific query parameter

Parameter	Omission	Description
Format	N	<p>The parameter to specify the expanding "A.2 Standard Component Name Specification Format (page 155)" or the grp component.</p> <p>The component types to be specified are only node or grp. Ignored when some value other than node and grp is set in this parameter.</p> <p>Expanded on the list of node components belonging to the specified group when grp component is specified in this parameter.</p>

- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the list succeeded.
400	Failure	<p>Either of the following errors occurs.</p> <ul style="list-style-type: none"> • The value of Format query parameter is not set. • The format of "A.2 Standard Component Name Specification Format (page 155)" specified in Format query parameter is invalid.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that can be used.

- Body

```
[
  NodeMemberObject
]
```

Refer to "[3.5 Node Member Resource \(NodeMemberObject\) \(page 27\)](#)" for the details of each property.

4.2 Status Monitoring Setting API

This section describes the API specification for operating status monitoring functions.

4.2.1 Obtaining Status Monitoring Rule Entry List

Obtaining the setting information of all status monitoring rule entries.

Request format

```
GET /umf/fw/nvp/stsmon/entries
```

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the information of status monitoring rule entries succeeded.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

```
[
  StsMonObject
]
```

Refer to "[3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)](#)" for the details of each property.

4.2.2 Obtaining Status Monitoring Rule Entry

Obtaining the setting information of the target status monitoring rule entry.

Request format

```
GET /umf/fw/nvp/stsmon/entries/{stsmon_id}
```

Specifying the value of ID of "[3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)](#)" in {stsmon_id}.

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the information of the status monitoring rule entry succeeded.
404	Failure	The specified status monitoring rule entry is not found.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

`StsMonObject`

Refer to "[3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)](#)" for the details of each property.

4.2.3 Adding Status Monitoring Rule Entry

Adding a new status monitoring rule entry.

Request format

`POST /umf/fw/nvp/stsmon/entries`

- API specific query parameter

None

- Body

`StsMonObject`

Refer to "[3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)](#)" for the details of each property.

- The details of specified status monitoring rule entry resource

Property	Omission	Description
ID	Y	Ignored even when some value is set in this property.
Rule	N	Specifying the status monitoring rule built-in Network Manager. Not allowed to set to null.
Name	Y	Not allowed to set to null. ""(an empty character) is set when setting the value of this property is omitted.
Status	Y	Not allowed to set to null. "Stopped"(not monitored) is set when setting the value of this property is omitted.
Nodes	N	Not allowed to set to null. Can be specified in " A.2 Standard Component Name Specification Format (page 155) ".

Property	Omission	Description
Interval	Y	Not allowed to set to null. "5m"(5 minutes) is set when setting the value of this property is omitted.
AlertSeverity	Y	Not allowed to set to null. "Warning"(warning) is set when setting the value of this property is omitted.
Arguments	Y	Not allowed to set to null. When setting the value of this property is omitted or the number of arguments is not enough, ""(an empty character) is set instead of the missing arguments. Handled as an error when the number of arguments is over the required number.

Response format

- Status code

code	Meaning	Description
200	Success	The status monitoring rule entry is added successfully.
400	Failure	Either of the following errors occurs. <ul style="list-style-type: none"> • The format of the status monitoring rule entry resource specified in the requesting is invalid. • The number of arguments specified in Arguments exceeds the required number in the status monitoring rule.
404	Failure	The specified rule is not found.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

```
StsMonObject
```

Returning the values of all properties including ID and the omitted properties.

Refer to ["3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)"](#) for the details of each property.

4.2.4 Updating Status Monitoring Rule Entry

Updating the setting of the added status monitoring rule entry. This API is also used to update the monitoring status of the status monitoring rule entry.

Request format

```
PUT /umf/fw/nvp/stsmon/entries/{stsmon_id}
```

Specifying the value of ID of ["3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)"](#) in {stsmon_id}.

- API specific query parameter

None

- Body

StsMonObject

Refer to "[3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)](#)" for the details of each property.

- The details of specified status monitoring rule entry resource.

Property	Omission	Description
ID	Y	When some value is set in this property, it needs to match the value with <code>{node_id}</code> in the URL.
Rule	Y	Ignored even when some value is set in this property.
Nodes	Y	Not allowed to set to null. Can be specified in " A.2 Standard Component Name Specification Format (page 155) ".
Arguments	Y	Not allowed to set to null for each argument. When the number of arguments is not enough, ""(an empty character) is set instead of the missing arguments. Handled as an error when the number of arguments is over the required number.
Others	Y	Not allowed to set to null.

Tip

- * Omitted properties are not updated. Specified properties are only updated.
- * To update properties other than the property of Status, it needs that the value of Status of the current status monitoring rule is "Stopped"(not monitored) or "Stopped"(not monitored) is set in Status in the update requesting. Otherwise, an error (409) is returned.

Response format

- Status code

code	Meaning	Description
200	Success	The status monitoring rule entry is updated successfully.
400	Failure	One of the following errors occurs. <ul style="list-style-type: none"> • The format of the status monitoring rule entry resource specified in the requesting is invalid. • The number of arguments specified in Arguments exceeds the required number in the status monitoring rule. • The <code>{stsmon_id}</code> in the URL and the ID specified in the message body are not matched.
404	Failure	The specified status monitoring rule entry is not found.
409	Failure	The status monitoring rule entry can not be updated because it is under monitoring.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

StsMonObject

Returning the values of all properties including not updated properties.

Refer to "[3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)](#)" for the details of each property.

Caution

It is not allowed to update the status monitoring rule entry whose value of Status is "Running"(monitored).

4.2.5 Deleting Status Monitoring Rule Entry

Deleting the registered status monitoring rule entry.

Request format

```
DELETE /umf/fw/nvp/stsmon/entries/{stsmon_id}
```

Specifying the value of ID of "[3.6 Status Monitoring Rule Entry Resource \(StsMonObject\) \(page 28\)](#)" in {stsmon_id}.

- API specific query parameter

None

- Body

None

Response format

- Status code

code	Meaning	Description
200	Success	The status monitoring rule entry is deleted successfully.
404	Failure	The specified status monitoring rule entry is not found.
409	Failure	The specified status monitoring rule entry can not be deleted because it is under monitoring.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that can be used.

- Body

None

Caution

It is not allowed to delete the status monitoring rule entry whose value of Status is "Running"(monitored). Update the value of Status to "Stopped"(not monitored) before executing this API.

4.2.6 Obtaining Status Monitoring Rule List

Obtaining the information (rule name and loaded state) of all status monitoring rules.

Request format

```
GET /umf/fw/nvp/stsmon/rules
```

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the information of status monitoring rules succeeded.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that can be used.

- Body

```
StsMonRuleObject
```

Refer to ["3.7 Status Monitoring Rule Resource \(StsMonRuleObject\) \(page 30\)"](#) for the details of each property.

4.3 Data Collection Setting API

This chapter describes the API specification for operating data collection functions.

4.3.1 Obtaining Data Collection Entry List

Obtaining the setting information of all data collection entries.

Request format

```
GET /umf/fw/nvp/datacol/entries
```

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the information of the data collection entry succeeded.

code	Meaning	Description
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

```
[
  DataColObject
]
```

Refer to ["3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)"](#) for the details of each property.

Caution

When the data collection entry whose alert condition is set to cumulative scheduling is already registered using the monitoring device function, it is possible to obtain the information of the data collection entry itself, on the other hand, it is impossible to obtain the schedule information that is set in the data collection entry.

4.3.2 Obtaining Data Collection Entry

Obtaining the setting information of the target data collection entry.

Request format

```
GET /umf/fw/nvp/datacol/entries/{datacol_id}
```

Specifying the value of ID of ["3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)"](#) in {datacol_id}.

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the information of the data collection entry succeeded.
404	Failure	The specified data collection entry is not found.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

```
DataColObject
```

Refer to "[3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)](#)" for the details of each property.

Caution

When the data collection entry whose alert condition is set to cumulative scheduling is already registered using the monitoring device function, it is possible to obtain the information of the data collection entry itself, on the other hand, it is impossible to obtain the schedule information that is set in the data collection entry.

4.3.3 Adding Data Collection Entry

Adding a new data collection entry.

Request format

POST /umf/fw/nvp/datacol/entries

- API specific query parameter
None
- Body

DataColObject

Refer to "[3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)](#)" for the details of each property.

- The details of specified data collection entry resource

Property	Omission	Description
ID	Y	Ignored even when some value is set in this property.
Name	N	Not allowed to set to null. In addition, not allowed to set to the followings as the title. <ul style="list-style-type: none"> • The title duplicated with other data collecting entries. (Half-width alphanumeric characters are case-insensitive.) • The title including surrogate pair characters. • The title including symbolic characters, i.e. "\ / : ? " < > . , " .
ReportType	N	Not allowed to set to null.
Status	Y	Not allowed to set to null. "Stopped"(not collecting) is set when setting the value of this property is omitted.
Nodes	N	Not allowed to set to null. Can be specified with " A.2 Standard Component Name Specification Format (page 155) ".
Interval	Y	Not allowed to set to null. "10m"(10 minutes) is set when setting the value of this property is omitted.
OperationMode	Y	Not allowed to set to null. "Store"(data storing only) is set when setting the value of this property is omitted.

Property	Omission	Description
		Tip Not allowed to specify some value other than "Store"(data storing only) when ReportType is set to "ServerLoad"(load of server).
ReportDataOutput	Y	Not allowed to set to null. "On" is set when setting the value of this property is omitted under the condition that OperationMode is set to "Store"(data storing only) or "Both"(both data storing and threshold monitoring). Tip Null is always set when OperationMode is set to some value other than "Store"(data storing only) or "Both"(both data storing and threshold monitoring).
Format	Y/N	Required to specify with the followings when ReportType is set to "General"(general purpose). In this case, not allowed to set to null. <ul style="list-style-type: none"> • Bps bps(bits per second) • Pps pps(packet per second) • Percent % • Diff Difference • Abs Without data processing Tip Null is always set when ReportType is set to some value other than "General"(general purpose).
MibExprName	Y/N	Required to specify the already registered MIB expression name when ReportType is set to "MIBExpression"(MIB expression). In this case, not allowed to set to null. Tip Null is always set when ReportType is set to some value other than "MIBExpression"(MIB expression).
Instances	Y/N	Required to specify the collection target MIB instance when ReportType is set to one of the followings. In this case, not allowed to set to null. <ul style="list-style-type: none"> • TrafficOfHub64bit Traffic of a specific port of a hub(64bit) • TrafficOfHub Traffic of a specific port • TrafficOfWAN64bit Traffic of WAN(64bit) • TrafficOfWAN

Property	Omission	Description
		<p>Traffic of WAN</p> <ul style="list-style-type: none"> MIBExpression <p>MIB expression</p> <p>When the following symbols are included in the specified character string, they are removed.</p> <ul style="list-style-type: none"> Half-width space Comma(,) used as delimiter at the top or the end of the character string <p>When commas(,) used as delimiter are consecutively set, they are replaced to one comma.</p> <p>Tip</p> <ul style="list-style-type: none"> Null is always set when ReportType is set to some value other the values described above. ""(an empty character) can be set in the meaning of no-specification only when ReportType is set to "MIBExpression"(MIB expression).
MIB	Y/N	<p>Required to specify the collection target MIB when ReportType is set to "General"(general purpose). In this case, not allowed to set to null.</p> <p>When the following symbols are included in the specified character string, they are removed.</p> <ul style="list-style-type: none"> Half-width space Comma(,) used as delimiter at the top or the end of the character string <p>When commas(,) used as delimiter are consecutively set, they are replaced to one comma.</p> <p>Tip</p> <p>Null is always set when ReportType is set to some value other than "General"(general purpose).</p>
RiseOrFall	Y	<p>Some value is set when OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring). In this case, not allowed to set to null.</p> <p>"Rise"(detecting excess) is set when setting the value of this property is omitted.</p> <p>Tip</p> <p>Null is always set when OperationMode is set to "Store"(data storing only).</p>
Threshold	Y/N	<p>Required to specify when OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring). In this case, not allowed to set to null.</p> <p>Tip</p> <p>Null is always set when OperationMode is set to "Store"(data storing only).</p>

Property	Omission	Description
AlertSeverity	Y	<p>Some value is set when OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring). In this case, not allowed to set to null. "Warning" is set when setting the value of this property is omitted.</p> <p>Tip</p> <p>Null is always set when OperationMode is set to "Store"(data storing only).</p>
AlertOutputCondition	Y	<p>Some value is set when OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring). In this case, not allowed to set to null. Set "Continuous"(continuous) in this property. When setting the value of this property is omitted, "Continuous"(continuous) is set too.</p> <p>Tip</p> <p>Null is always set when OperationMode is set to "Store"(data storing only).</p>
ContinuousTimes	Y	<p>Some value is set when AlertOutputCondition is set to "Continuous"(continuous). In this case, not allowed to set to null. 1 is set when setting the value of this property is omitted.</p> <p>Tip</p> <p>Null is always set when AlertOutputCondition is set to some value other than "Continuous"(continuous).</p>
RecoveryAlert	Y	<p>Some value is set when AlertOutputCondition is set to "Continuous"(continuous). In this case, not allowed to set to null. "On" is set when setting the value of this property is omitted.</p> <p>Tip</p> <p>Null is always set when AlertOutputCondition is set to some value other than "Continuous"(continuous).</p>
RecoveryThreshold	Y/N	<p>Required to specify when RecoveryAlert is set to "On". In this case, not allowed to set to null.</p> <p>Tip</p> <p>Null is always set when RecoveryAlert is set to some value other than "On".</p>

Response format

- Status code

code	Meaning	Description
200	Success	The data collection entry is added successfully.
400	Failure	<p>One of the following errors occurs.</p> <ul style="list-style-type: none"> • The format of the data collection entry resource specified in the requesting is invalid.

code	Meaning	Description
		<ul style="list-style-type: none"> The properties required depending on the value of ReportType are not specified. ""(an empty character) or more than 3 MIB names is specified in specifying MIB. The required properties in the case that OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring) are not specified. Some value other than "Store"(data storing only) is set in OperationMode when ReportType is set to "ServerLoad"(load of server). The magnitude relation between the value of the specified Threshold and that of RecoveryThreshold is invalid.
404	Failure	The MIB expression rule specified in MIBExprName is not found.
409	Failure	The specified entry name is duplicated with the entry name of the existing collection entry.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

```
DataColObject
```

Returning the values of all properties including ID and the omitted properties.

Refer to ["3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)"](#) for the details of each property.

4.3.4 Updating Data Collection Entry

Updating the settings of the registered data collection entry. This API is also used to update the monitoring state of the data collection entry.

Request format

```
PUT /umf/fw/nvp/datacol/entries/{datacol_id}
```

Specifying the value of ID of ["3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)"](#) in {datacol_id}.

- API specific query parameter
None
- Body

```
DataColObject
```

Refer to ["3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)"](#) for the details of each property.

- The details of specified data collection entry resource

Property	Omission	Description
ID	Y	When some value is set in this property, it needs to match the value with <code>{datacol_id}</code> in the URL.
Name	Y	Not allowed to update the registered settings. Ignored even when some value is set in these properties.
ReportType	Y	
Nodes	Y	Not allowed to set to null. Can be specified with " A.2 Standard Component Name Specification Format (page 155) ".
OperationMode	Y	Not allowed to set to null. Not allowed to set to some value other than "Store"(data storing only) when ReportType is set to "ServerLoad"(load of server).
Format	Y	<p>Setting one of the followings when ReportType is set to "General"(general purpose). In this case, not allowed to set to null.</p> <ul style="list-style-type: none"> • Bps bps(bits per second) • Pps pps(packet per second) • Percent % • Diff Difference • Abs Without data processing <p>Tip</p> <hr/> <p>Null is always set when ReportType is set to "General"(general purpose).</p> <hr/>
MibExprName	Y	<p>Specifying the registered MIB expression name when ReportType is set to "MIBExpression"(MIB expression). In this case, not allowed to set to null.</p> <p>Tip</p> <hr/> <p>Null is always set when ReportType is set to some value other than "MIBExpression"(MIB expression).</p> <hr/>
Instances	Y	<p>Specifying the collection target MIB instance when ReportType is set to one of the followings. In this case, not allowed to set to null.</p> <ul style="list-style-type: none"> • TrafficOfHub64bit Traffic of a specific port of a hub(64bit) • TrafficOfHub Traffic of a specific port of a hub • TrafficOfWAN64bit Traffic of WAN(64bit) • TrafficOfWAN Traffic of WAN • MIBExpression MIB expression

Property	Omission	Description
		<p>When the following symbols are included in the specified character string, they are removed.</p> <ul style="list-style-type: none"> • Half-width space • Comma(,) used as delimiter at the top or the end of the character string <p>When commas(,) used as delimiter are consecutively set, they are replaced to one comma.</p> <p>Tip</p> <ul style="list-style-type: none"> • Null is always set when ReportType is set to some value other than the values described above. • ""(an empty character) can be set in the meaning of no-specification only when ReportType is set to "MIBExpression"(MIB expression).
MIB	Y	<p>Specifying the collection target MIB when ReportType is set to "General"(general purpose). In this case, not allowed to set to null.</p> <p>When the following symbols are included in the specified character string, they are removed.</p> <ul style="list-style-type: none"> • Half-width space • Comma(,) used as delimiter at the top or the end of the character string <p>When commas(,) used as delimiter are consecutively set, they are replaced to one comma.</p> <p>Tip</p> <p>Null is always set when ReportType is set to some value other than "General"(general purpose).</p>
RiseOrFall	Y	<p>Some value is set when OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring). In this case, not allowed to set to null.</p> <p>In the case that updating the value of OperationMode from "Store"(data storing only), "Rise"(detecting excess) is set when setting the value of this property is omitted.</p> <p>Tip</p> <p>Null is always set when OperationMode is set to "Store"(data storing only).</p>
Threshold	Y/N	<p>Some value is set when OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring). In this case, not allowed to set to null.</p> <p>Not allowed to omit the value of this property when updating the value of OperationMode from "Store"(data storing only).</p> <p>Tip</p> <p>Null is always set when OperationMode is set to "Store"(data storing only).</p>
AlertSeverity	Y	<p>Some value is set when OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring). In this case, not allowed to set to null.</p>

Property	Omission	Description
		<p>In the case that updating the value of OperationMode from "Store"(data storing only), "Warning" is set when setting the value of this property is omitted.</p> <p>Tip</p> <p>Null is always set when OperationMode is set to "Store"(data storing only).</p>
AlertOutputCondition	Y	<p>Some value is set when OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring). In this case, not allowed to set to null.</p> <p>Not allowed to set some value other than "Continuous"(continuous) when updating the value of OperationMode from "Store"(data storing only). In this case, "Continuous"(continuous) is set when the value of this property is omitted.</p> <p>Tip</p> <ul style="list-style-type: none"> • Null is always set when OperationMode is set to "Store"(data storing only). • Setting "Total"(total) only when keeping this value "Total"(total) without updating.
ContinuousTimes	Y	<p>In updating the value of AlertOutputCondition from some value to "Continuous"(continuous), 1 is set when setting the value of this property is omitted. Not allowed to set to null.</p> <p>Tip</p> <p>Null is always set when AlertOutputCondition is set to some value other than "Continuous"(continuous).</p>
RecoveryAlert	Y	<p>In updating the value of AlertOutputCondition from some value to "Continuous"(continuous), "On" is set when setting the value of this property is omitted. Not allowed to set to null.</p> <p>Tip</p> <p>Null is always set when AlertOutputCondition is set to some value other than "Continuous"(continuous).</p>
RecoveryThreshold	Y/N	<p>Required to specify in the following cases. In this case, not allowed to set to null.</p> <ul style="list-style-type: none"> • The case when updating RecoveryAlert from "Off" to "On". • The case when updating the value of OperationMode from "Store"(data storing only) to "Monitor"(threshold monitoring only) or "Both"(data storing only), and moreover, setting the value of RecoveryAlert is omitted or "On" is set in this property. <p>Tip</p> <p>Null is always set when RecoveryAlert is set to "Off" or null.</p>
Others	Y	Not allowed to set to null.

Tip

* Omitted properties are not updated. Specified properties are only updated.

- * To update properties other than that of Status, it needs that the value of the current data collection entry is "Stopped"(not collecting), or "Stopped"(not collecting) is set in Status in the update requesting. Otherwise, an error(409) is returned.

Response format

- Status code

code	Meaning	Description
200	Success	The data collection entry is updated successfully.
400	Failure	One of the following errors occurs. <ul style="list-style-type: none"> • The format of the data collection entry resource specified in the requesting is invalid. • The properties required depending on the value of ReportType are not specified. • ""(an empty character) or more than 3 MIB names is specified in specifying MIB. • The required properties in the case that OperationMode is set to "Monitor"(threshold monitoring only) or "Both"(both data storing and threshold monitoring) are not specified. • The magnitude relation between the value of the specified Threshold and that of RecoveryThreshold is invalid. • The {datacol_id} in the URL and the ID specified in the message body are not matched.
404	Failure	Either of the following errors occurs. <ul style="list-style-type: none"> • The specified data collection entry is not found. • The specified MIB expression rule is not found.
409	Failure	One of the following errors occurs. <ul style="list-style-type: none"> • The magnitude relation between the value of the specified Threshold and that of the existing RecoveryThreshold is invalid. • The magnitude relation between the value of the existing Threshold and that of the specified RecoveryThreshold is invalid. • Some value other than "Store"(data storing only) is set in OperationMode when ReportType is set to "ServerLoad"(load of server). • The data collection entry is under collection process.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

DataColObject

Returning the values of all properties including not updated properties.

Refer to ["3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)"](#) for the details of each property.

Caution

It is not allowed to update the data collection entry whose value of Status is "Running"(collecting).

4.3.5 Deleting Data Collection Entry

Deleting the registered data collection entry.

Request format

```
DELETE /umf/fw/nvp/datacol/entries/{datacol_id}
```

Specifying the value of ID of "[3.8 Data Collection Entry Resource \(DataColObject\) \(page 31\)](#)" in {datacol_id}.

- API specific query parameter

None

- Body

None

Response format

- Status code

code	Meaning	Description
200	Success	The data collection entry is deleted successfully.
404	Failure	The specified data collection entry is not found.
409	Failure	The specified data collection entry can not be deleted because it is under monitoring.
503	Failure	The process is suspended due to timeout of an exclusive lock.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

None

Caution

It is not allowed to delete the data collection entry whose value of Status is "Running"(collecting). Update the value of Status to "Stopped"(not collecting) before using this API.

4.4 Alert Management API

This section describes the specification of API to obtain the alert list.

4.4.1 Obtaining Alert List

Obtaining the list of all alerts.

Request format

```
GET /umf/fw/nvp/alerts
```

Narrowing down the alerts to obtain by specifying query parameters.

- API specific query parameters

Parameter	Omission	Description
Detail	Y	<p>Specifying the property granularity of the alert.</p> <ul style="list-style-type: none"> • None is specified Returning properties excluding a part of property such as details. Refer to the annotation described in section 3.12 for the details of the omitted properties when setting the value of this parameter is omitted. • full Returning all property information. • Others Processed in the same way as none is specified. <p>When this parameter is specified together with Fields parameter, this parameter(Detail) becomes effective, on the other hand, Fields parameter becomes ineffective.</p>
Fields	Y	<p>Specifying alert properties separately.</p> <p>Specifying the properties to include in the response with the format of "Alerts/<property name>". Can be specified multiple properties by concatenating with ","(commas).</p> <p>Example:</p> <pre>Fields=Alerts/ID,Alerts/SummaryMessage</pre> <p>When this parameter is specified together with Detail parameter, Detail parameter becomes effective , on the other hand, this parameter(Fields) becomes ineffective.</p>
SinceOccurTime	Y	<p>Specifying the oldest alert occurrence time.</p> <p>Refer to "Table 2-6 Dateformat (page 11)" for the format that can be specified with.</p> <p>Example:</p> <p>When "Since=2015-03-20T10:00:00+09:00" is set, the alert that occurred at 10:00:00 on 20th May 2015 is included in the target to be obtained.</p>
UntilOccurTime	Y	<p>Specifying the latest alert occurrence time.</p> <p>Refer to "Table 2-6 Dateformat (page 11)" for the format that can be specified with.</p> <p>When "UntilOccurTime=2015-03-21T13:00:00+09:00" is set, the alert that occurred at 13:00:00 on 21th May 2015 is included in the target to be obtained.</p>
ComponentType	Y	<p>Narrowing down the alerts to be obtained based on the component type of origin of the alert.</p> <p>Nodes(node), interfaces(if) and systems(system) can be specified.</p> <p>When ComponentName is specified on the other hand ComponentType is not specified, searching alerts as "node" is set in this parameter. When neither ComponentName and ComponentType is specified, the search results are not narrowed down based on the component type.</p>
ComponentName	Y	<p>Narrowing down the alerts to be obtained based on the component name of origin of the alert.</p> <p>Can be specified with "Appendix B. Specifying wildcards (page 157)".</p>
IPAddress	Y	<p>Narrowing down the alerts to be obtained based on the IP address.</p> <p>Specifies with IPv4 or IPv6 format.</p>

Parameter	Omission	Description
		Can be specified with " Appendix B. Specifying wildcards (page 157) ".
RecoveryStatus	Y	Specifying the status of searching. The search results can be narrowed down based on the multiple status conditions(OR condition) by concatenating them with ","(commas). <ul style="list-style-type: none"> • NeedRecover : Need recovery • AutoRecover : Auto recovery • Recovered : Already recovered • NeedNotRecover : Need not recovery
ConfirmStatus	Y	Narrowing down the alerts to be obtain based on the confirmation status. <ul style="list-style-type: none"> • Confirmed : Confirmed • Unconfirmed : Unconfirmed
Type	Y	Specifying the alert type. The search results can be narrowed down based on the multiple type conditions(OR condition) by concatenating them with ","(commas). <ul style="list-style-type: none"> • System : System • SNMPTrap : SNMP trap • SystemLog : System log • EventLog : Event log • Warning : Warning • Minor : Minor fault • Major : Major fault • Critical : Critical fault • Recovered : Recovered • Other : Others
Severity	Y	Specifying the alert severity. The search results can be narrowed down based on the multiple severity conditions(OR condition) by concatenating them with ","(commas). <ul style="list-style-type: none"> • Fatal : Fatal • Major : Major MAJOR • Minor : Minor MINOR • Warning : Warning • Unknown : Unknown • Normal : Normal
Sender	Y	Narrowing down the alerts based on the alert publisher. Can be specified with " Appendix B. Specifying wildcards (page 157) ". Specified with a character string of up to 63 characters.
SummaryMessage	Y	Narrowing down the alerts based on the character string included in the "Summary". Can be specified with " Appendix B. Specifying wildcards (page 157) ". Specified with a character string of up to 128 characters.
SNMPCommunity	Y	Narrowing down the alerts based on the SNMP community name of SNMPv1 or SNMPv2 trap. Can be specified with " Appendix B. Specifying wildcards (page 157) ". Specified with a character string of up to 255 characters.
AgentAddress	Y	Narrowing down the alerts based on the agent address of SNMPv1 trap.

Parameter	Omission	Description
		Specified with IPv4 address format such as 192.168.10.0.
EnterpriseOID	Y	Narrowing down the alerts based on the vendor identification of SNMP trap. Specified with full numeric characters string format, e.g. 1.3.6.1.4.1.119.1.2.69.1.1. Can be specified with " Appendix B. Specifying wildcards (page 157) ". Specified with a character string of up to 255 characters.
GenericCode	Y	Narrowing down the alerts based on the common trap code of SNMP. Specified with integer expression.
SpecificCode	Y	Narrowing down the alerts based on the trap code of SNMP trap. Specified with integer expression.
MaxCount	Y	Specifying the number of the search results to display with integer expression. Specified with a number of from 1 to 1000. 100 is set when setting the value of this parameter is omitted.
NextID	Y	Specifying the ID of the alert to be searched next with integer expression. Setting the value of ResultNextID to that of NextID and the value of ResultNextTime to that of SinceOccurTime, when the response body of ResultNextID is set to some value other than 0. Needed to be used in pairs with ResultNextTime. When the occurrence time of the ID set in NextID and the alert occurrence time set in SinceOccurTime do not match with one another, an unintended alert may be included in the results.

- Body

None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the alert list succeeded.
400	Failure	Either of the following errors occurs. <ul style="list-style-type: none"> • The format of the specified query parameter in the requesting is invalid. • The timezones specified in multiple query parameters do not match with one another.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

```
{
  "ResultNextID" : integer,
  "ResultNextTime" : string,
  "Alerts" : [
    AlertObject
```



```
]
}
```

Refer to "[3.9 Alert Resource \(AlertObject\) \(page 35\)](#)" for the details of each property.

Table 4-3 The details of JSON returned as the body.

Property	JSON Type	Description
ResultNextID	integer	The alert ID to be specified next in the case that the number of alerts meeting the search criteria exceeds MaxCount. "0" is returned when all alerts are already obtained.
ResultNextTime	string	The occurrence time of the alert to be specified next in the case that the number of alerts meeting the search criteria exceeds MaxCount. An empty character("") is returned when all alerts are already obtained.
Alerts	array[]	The array of " 3.9 Alert Resource (AlertObject) (page 35) ".

Caution

1. It may take times for above processes when filtering is not performed.
2. The number of alerts can be processed at one time is the number specified in MaxCount.
When the number of the target alerts exceeds MaxCount, the alert ID to be specified next is set in ResultNextID. Setting the value of ResultNextID included in the response body in NextID and setting the value of ResultNextTime included in the response body in SinceOccurTime, and then, repeating to execute this API until the value of ResultNextID becomes "0".
Refer to "[5.4.1 Confirming Alert of Target Node \(page 149\)](#)" for the calling image.
3. The alert set extracted by the search criteria is obtained in ascending order of alert occurrence time and in ascending order of alert ID.
4. It is not allowed to specify the search criteria that "the value of the property of the target alert is null".

4.4.2 Obtaining Alert Details

Obtaining the property information of the target alert.

Request format

```
GET /umf/fw/nvp/alerts/{alert_id}
```

Specifying the value of ID of "[3.9 Alert Resource \(AlertObject\) \(page 35\)](#)" in {alert_id}.

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	Obtaining the alert succeeded.
404	Failure	The specified alert is not found.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

AlertObject

Refer to ["3.9 Alert Resource \(AlertObject\) \(page 35\)"](#) for the details of each property.

4.4.3 Updating Alert

Updating alert properties. This API is used to confirm or recover an alert.

Request format

PUT /umf/fw/nvp/alerts/{alert_id}

Specifying the value of ID of ["3.9 Alert Resource \(AlertObject\) \(page 35\)"](#) in {alert_id}.

- API specific query parameter

None

- Body

AlertObject

- The details of the specified alert resource

Property	Omission	Description
ID	Y	When some value is set in this property, it needs to match the value with {alert_id} in the URL.
RecoveryStatus	Y	The value that can be set is only "Recovered" against the alert that needs to be recovered. Not allowed to manually recover the auto recovered alert or to return the already recovered alert back to not recovered one. Not allowed to set to null.
ConfirmStatus	Y	Not allowed to set to null.
Others	Y	Ignored even when some value is set in this property.

Tip

The omitted properties are not updated. The specified properties are only updated.

Response format

- Status code

code	Meaning	Description
200	Success	The alert is updated successfully.

code	Meaning	Description
		⚠ Caution Success(status code 200) is returned when the status is not updated such as the case that none is specified in the body or "Confirmed" is set again in ConfirmStatus property that "Confirmed" is already set.
400	Failure	Either of the following errors occurs. <ul style="list-style-type: none"> The format of the alert resource specified in the requesting is invalid. The ID specified in the message body does not match with <code>{alert_id}</code> in the URL.
404	Failure	The specified alert is not found.
409	Failure	Trying to recover some alert other than the manual recovery alert.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

AlertObject

Updating the values of all properties including the not updated properties.

Refer to ["3.9 Alert Resource \(AlertObject\) \(page 35\)"](#) for the details of each property.

⚠ Caution

When confirming(ConfirmStatus) and recovering(RecoveryStatus) are set together, the updating is processed in the order of confirming and recovering.

4.4.4 Updating Alert Properties in a Batch

Updating alerts to same values in a batch. This API is used to confirm or recover multiple alerts.

Request format

POST /umf/fw/nvp/alerts/batch-update

Narrowing down the target alerts by specifying query parameters.

- API specific query parameters

Parameter	Omission	Description
ID	Y	The ID of the target alert. Can be specified as follows. <ul style="list-style-type: none"> Specification of a single ID Example: 12345 Specification of multiple IDs using ',' (commas) Example: 100,121,130,150 Not allowed to specify this parameter together with other search criteria.
SinceOccurTime	Y	The same query parameters as those described in "4.4.2 Obtaining Alert Details (page 83)" . Not allowed to specify these parameters together with "ID" of URL query parameter.
UntilOccurTime	Y	
ComponentType	Y	
ComponentName	Y	

Parameter	Omission	Description
IPAddress	Y	
RecoveryStatus	Y	
ConfirmStatus	Y	
Type	Y	
Severity	Y	
Sender	Y	
SummaryMessage	Y	
SNMPCCommunity	Y	
AgentAddress	Y	
EnterpriseOID	Y	
GenericCode	Y	
SpecificCode	Y	
NextID	Y	
MaxCount	Y	

- Body

AlertObject

- The details of specified alert resource

The omitted properties are not updated. The specified properties are only updated.

Property	Omission	Description
ID	Y	Not needed to be specified. Needed to be matched with the ID of the query parameter when some value is set in this property.
RecoveryStatus	Y	The value to be set is only "Recovered" against the alert that needs to be recovered. It is not allowed to manually recover the auto recovered alert or to return the already recovered alert back to not recovered one. Not allowed to set to null.
ConfirmStatus	Y	Not allowed to set to null.
Others	Y	Ignored even when some value is set in this property.

Response format

- Status code

code	Meaning	Description
200	Success	The alert is updated successfully. The updating process is handled as a success even when the number of target alerts to be updated is 0.
400	Failure	One of the following errors occurs. <ul style="list-style-type: none"> • The format of the specified query parameter in the requesting is invalid. • The format of the specified alert resource in the requesting is invalid.

code	Meaning	Description
		<ul style="list-style-type: none"> The resource ID specified in the query parameter and that of specified in the JSON are not matched. The ID parameter and other search criteria parameters are specified together. The timezones specified in multiple query parameters do not match with one another.
500	Failure	There is an update incomplete alert even when the alert is specified to be recovered. When this error occurs during a batch process, the updates occurring after this error may become incomplete.

Refer to ["2.3.2 Common HTTP Status Codes \(page 12\)"](#) for other status codes that may be used.

- Body

```
{
  "ResultNextID" : integer,
  "ResultNextTime" : string
}
```

Returning the values of all properties including the not updated properties.

Refer to ["3.9 Alert Resource \(AlertObject\) \(page 35\)"](#) for the details of each property.

Table 4-5 The details of JSON returned as the body

Property	JSON Type	Description
ResultNextID	integer	The alert ID to be specified next in the case that the number of alerts meeting the search criteria exceeds MaxCount. "0" is returned when all alerts are already obtained.
ResultNextTime	string	The occurrence time of the alert to be specified next in the case that the number of alerts meeting the search criteria exceeds MaxCount. An empty character("") is returned when all alerts are already obtained.

Caution

The error priority is as follows.

- Stopping the process immediately and returning the 500 series error when the 500 series error occurs during a period from the beginning of the process to the end of the process.
- Confirming values before the process, and returning the 400 error without processing when the values are invalid.
- Returning 200 when the process finished without matching with all the above conditions.

Caution

1. When confirming(ConfirmStatus) and recovering(RecoveryStatus) are set together, the updating is processed in the order of confirming and recovering.
2. The number of alerts can be processed at one time is the number specified in MaxCount.

When the number of the target alerts exceeds MaxCount, the alert ID to be specified next is set in ResultNextID. Setting the value of ResultNextID included in the response body in NextID and setting the value of ResultNextTime included in the response body in SinceOccurTime, and then, repeating to execute this API until the value of ResultNextID becomes "0".

Refer to ["5.4.1 Confirming Alert of Target Node \(page 149\)"](#) for the calling image.

3. It is not allowed to specify the search criteria that "the value of the property of the target alert is null".

4.4.5 Deleting Alert

Deleting the alert.

Request format

```
DELETE /umf/fw/nvp/alerts/{alert_id}
```

Specifying the value of ID of "[3.9 Alert Resource \(AlertObject\) \(page 35\)](#)" in `{alert_id}`.

- API specific query parameter
None
- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	The alert is deleted successfully.
404	Failure	The specified alert is not found.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body
None

4.4.6 Deleting Alerts in a Batch

Deleting alerts in a batch.

Request format

```
POST /umf/fw/nvp/alerts/batch-delete
```

Narrowing down the target alerts by specifying query parameters.

- API specific query parameters

Parameter	Omission	Description
ID	Y	<p>The ID of the target alert. Can be specified as follows.</p> <ul style="list-style-type: none"> • Specification of a single ID Example: 12345 • Specification of multiple IDs using ',' (commas) Example: 100,121,130,150 <p>Not allowed to specify this parameter together with other search criteria.</p>

Parameter	Omission	Description
SinceOccurTime	Y	The same query parameters as those described in " 4.4.2 Obtaining Alert Details (page 83) ". Not allowed to specify these parameters together with "ID" of URL query parameter.
UntilOccurTime	Y	
ComponentType	Y	
ComponentName	Y	
IPAddress	Y	
RecoveryStatus	Y	
ConfirmStatus	Y	
Type	Y	
Severity	Y	
Sender	Y	
SummaryMessage	Y	
SNMPCCommunity	Y	
AgentAddress	Y	
EnterpriseOID	Y	
GenericCode	Y	
SpecificCode	Y	
NextID	Y	
MaxCount	Y	

- Body
None

Response format

- Status code

code	Meaning	Description
200	Success	The target alert is deleted successfully.
400	Failure	One of the following errors occurs. <ul style="list-style-type: none"> • The format of the specified query parameter in the requesting is invalid. • The ID parameter and other search criteria parameters are specified together. • The timezones specified in multiple query parameters do not match with one another.

Refer to "[2.3.2 Common HTTP Status Codes \(page 12\)](#)" for other status codes that may be used.

- Body

```
{
  "ResultNextID" : integer,
  "ResultNextTime" : string
}
```

Refer to "[3.9 Alert Resource \(AlertObject\) \(page 35\)](#)" for the details of each property.

Table 4-5 The details of JSON returned as the body

Property	JSON Type	Description
ResultNextID	integer	The alert ID to be specified next in the case that the number of alerts meeting the search criteria exceeds MaxCount. "0" is returned when all alerts are already obtained.
ResultNextTime	string	The occurrence time of the alert to be specified next in the case that the number of alerts meeting the search criteria exceeds MaxCount. An empty character("") is returned when all alerts are already obtained.

Caution

1. The number of alerts can be processed at one time is the number specified in MaxCount.

When the number of the target alerts exceeds MaxCount, the alert ID to be specified next is set in ResultNextID. Setting the value of ResultNextID included in the response body in NextID and setting the value of ResultNextTime included in the response body in SinceOccurTime, and then, repeating to execute this API until the value of ResultNextID becomes "0".

Refer to "[5.4.1 Confirming Alert of Target Node \(page 149\)](#)" for the calling image.

2. It is not allowed to specify the search criteria that "the value of the property of the target alert is null".

Chapter 5.

Tutorial

This chapter describes the basic way to use WebAPI using specific operation examples.

Contents

5.1 Operating Configuration Information	92
5.2 Operating Status Monitoring Rule Entry	128
5.3 Operating Data Collection Entry	140
5.4 Operating Alert	149

5.1 Operating Configuration Information

This section describes the way to operate the information of maps and nodes using each WebAPI.

5.1.1 Adding New Map

This subsection describes the procedure to add a new map under the existing map.

This operation is executed to manage the management target node in accordance with the physical location of the node and the network structure.

In this subsection, the procedure is described using the following operation example.

Example: Adding map “*First Sales Department*” under map “*Branch A*”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of map “*Branch A*” that is adding destination.

Obtaining map IDs by executing the API for obtaining map list. Refer to ["4.1.1 Obtaining Map List \(page 40\)"](#) for the details.

Tip

It is possible to narrow down target maps by specifying a map name in NameFilter query parameter.

- Request:

```
GET /umf/fw/nvp/maps?NameFilter=%E6%94%AF%E5%BA%97A HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Wed, 11 Nov 2015 07:07:33 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:8ucvyMXZzSNQy5Th9dhwQ6TnMoJ/znxo0SeUVFMkuZc=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 28
Date: Wed, 18 Nov 2015 19:09:39 GMT

[
  {
    "Name": "Branch A",
    "ID": 48
  }
]
```

2. Adding map “*First Sales Department*” under map “*Branch A*”.

Adding the map by executing the API for adding map. Refer to ["4.1.3 Adding Map \(page 41\)"](#) for the details.

Tip

It is possible to specify the adding destination map by specifying a map ID in UpperMap query parameter.

- Request:

```
POST /umf/fw/nvp/maps?UpperMap=106 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 27
Date: Wed, 11 Nov 2015 08:10:35 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:KbpcTMFwcSfykRAbkwlZZ3pzeHub6aLcluVwJwGEJsU=

{
  "Name": "First Sales Department"
}
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 227
Date: Wed, 11 Nov 2015 19:20:29 GMT

{
  "Administrator": null,
  "NetworkAddress": null,
  "Name": "First Sales Department",
  "Alias": null,
  "ApplicationPath": null,
  "IPv6NetworkAddress": null,
  "URL": null,
  "Location": null,
  "NetworkMask": null,
  "IconType": "map",
  "ID": 108,
  "IPv6PrefixLength": null
}
```

5.1.2 Deleting Information of Map and its Subordinates in a Batch

This subsection describes the procedure to delete all information of the specified map, and nodes and maps belonging to the map in a batch.

This operation is executed when it become unnecessary to manage multiple nodes belonging to the target map.

In this subsection, the procedure is described using the following operation example.

Example: Deleting map “*Verification environment*” after confirming the information of nodes belonging to map “*Verification environment*”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of the deleting target map “*Verification environment*”.

Obtaining map IDs by executing the API for obtaining map list. Refer to ["4.1.1 Obtaining Map List \(page 40\)"](#) for the details.

Tip

It is possible to narrow down target maps by specifying a map name in NameFilter query parameter.

- Request:

```
GET /umf/fw/nvp/maps?NameFilter=%E6%A4%9C%E8%A8%BC%E7%92%B0%E5%A2%83 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 00:02:28 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfpPdVj8U=:dgmmkJsSvpcINXYt9vRmdn7Rodzzvfr5J2h6nz2mylo=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 33
Date: Thu, 12 Nov 2015 00:02:30 GMT

[
  {
    "Name": "Verification environment",
    "ID": 48
  }
]
```

2. Checking the information of nodes and maps belonging to map “*Verification environment*”.

Obtaining the information of nodes and maps belonging to the map by executing the API for obtaining map configuration. Refer to ["4.1.14 Obtaining Map Configuration \(page 60\)"](#) for the details.

Tip

It is possible to search maps belonging to the target map recursively by setting "true" in Recursive query parameter.

- Request:

```
GET /umf/fw/nvp/maps/48/members?Recursive=true HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 00:05:29 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:pTsPB0yeZHRykUQbaa0csgb6OlieUn3TCEieKXXiGh4=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 309
Date: Thu, 12 Nov 2015 00:05:30 GMT
```

```
[
  {
    "Type": "map",
    "Name": "test.0",
    "Members": [
      {
        "Type": "node",
        "Name": "subtest.0",
        "ID": 52
      }
    ],
    "ID": 50
  },
  {
    "Type": "map",
    "Name": "test.1",
    "Members": [
      {
        "Type": "node",
        "Name": "subtest.1",
        "ID": 56
      }
    ],
    "ID": 54
  },
  {
    "Type": "node",
    "Name": "test.0",
    "ID": 58
  }
]
```

```

    },
    {
      "Type": "node",
      "Name": "test.1",
      "ID": 60
    },
    {
      "Type": "node",
      "Name": "test.2",
      "ID": 62
    }
  ]

```

3. Deleting map “*Verification environment*”.

Deleting all information of the target map and its subordinate nodes and maps by executing the API for deleting map. Refer to ["4.1.5 Deleting Map \(page 44\)"](#) for the details.

- Request:

```

DELETE /umf/fw/nvp/maps/48 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 00:07:11 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:cBJksEmJcxsvBKmd56R0aZpEkAZVum2tooXtmA58dS4=

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Content-Length: 0
Date: Thu, 12 Nov 2015 00:07:13 GMT

```

5.1.3 Adding New Node

This subsection describes the procedure to add a new node on the specified map.

This operation is executed to add a new node that newly becomes the management target.

In this subsection, the procedure is described using the following operation example.

Example: Adding node “*sys_sw01*” on map “*System Department*”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of map “*System Department*” that is adding destination.

Obtaining map IDs by executing the API for obtaining map list. Refer to ["4.1.1 Obtaining Map List \(page 40\)"](#) for the details.

Tip

It is possible to narrow down target maps by specifying a map name in NameFilter query parameter.

- Request:

```
GET /umf/fw/nvp/maps?NameFilter=%E3%82%B7%E3%82%B9%E3%83%86%E3%83%A0%E9%83%A8 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 00:14:22 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:b/NqmOCF7HQWkF5ycUgNQrACG4mhsmH9yRIO8g2GSx4=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 36
Date: Thu, 12 Nov 2015 00:14:24 GMT

[
  {
    "Name": "System Department",
    "ID": 64
  }
]
```

2. Adding node “sys_sw01” with its monitoring mode OFF under map “System Department”.

Adding the node by executing the API for adding node. Refer to ["4.1.8 Adding Node \(page 47\)"](#) for the details.

Tip

It is possible to specify the adding destination map by specifying a map ID in UpperMap query parameter.

- Request:

```
POST /umf/fw/nvp/nodes?UpperMap=64 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 127
Date: Thu, 12 Nov 2015 00:15:28 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:cGNmBDetBVQZxgokfmqsnJKasvOdPLJbO+88TrULGaM=

{ "Name": "sys_sw01", "SNMPVersion": "2C", "IPAddress": "192.168.1
0.254", "SNMPCommunityGet": "public", "MonitoringMode": "Off" }
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
```

```

X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json;charset=UTF-8
Content-Length: 921
Date: Thu, 12 Nov 2015 00:15:30 GMT

```

```

{
  "WebAccessPort": null,
  "SNMPv3AuthenticationProtocol": null,
  "Group": null,
  "SNMPv3UserName": null,
  "IPv6Interfaces": [],
  "AgentType": null,
  "DPID": null,
  "SNMPv3SeverityOfInvalidEngineID": null,
  "MonitoringMode": "Off",
  "DiscoveryProtocol": null,
  "OSType": null,
  "SNMPPort": null,
  "TelnetServer": "Off",
  "SysDescr": null,
  "AdministrationNodeName": null,
  "Administrator": null,
  "IPv6Address": null,
  "ApplicationPath": null,
  "SNMPCommunitySet": null,
  "Memo": null,
  "HardwareType": null,
  "DeviceFrontPanel": null,
  "SNMPCommunityGet": "public",
  "IconType": "host",
  "IPAddress": "192.168.10.254",
  "SNMPv3EngineID": null,
  "SNMPv3SecurityLevel": "NoAuth/NoPriv",
  "Name": "sys_sw01",
  "FloatingIP": null,
  "URL": null,
  "Alias": null,
  "SNMPv3PrivacyProtocol": null,
  "SNMPCharacterCode": null,
  "SNMPVersion": "2C",
  "ID": 66,
  "RoutingControl": null,
  "IPv4Interfaces": [],
  "SysName": null,
  "SysObjectID": null,
  "SoftwareVersion": null,
  "LocalIPAddress": null,
  "FEX_ID": null,
  "DefaultTargetPort": null,
  "Location": null
}

```

3. Updating the property information of node “sys_sw01” to match with the information of the actual device by obtaining the information using SNMP.

Updating the property information of the node by executing the API for updating node device information. Refer to ["4.1.11 Updating Device Information of Node \(page 56\)"](#) for the details.

Tip

It is possible to specify the updated range by specifying some value in Action query parameter.

In the following example, "all" is set, this means updating all properties of the information of system and interface.

- Request:

```
POST /umf/fw/nvp/nodes/66/update?Action=all HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 01:00:43 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfpPdVj8U=:b461Ob0HrfFwZk2mkJFqgXh7aVC2pOtGlEB9yxe+pR4=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 7944
Date: Thu, 12 Nov 2015 01:00:50 GMT

{
  "WebAccessPort": null,
  "SNMPv3AuthenticationProtocol": null,
  "Group": null,
  "SNMPv3UserName": null,
  "IPv6Interfaces": [],
  "AgentType": 9,
  "DPID": null,
  "SNMPv3SeverityOfInvalidEngineID": null,
  "MonitoringMode": "Off",
  "DiscoveryProtocol": null,
  "OSType": "IOS",
  "SNMPPort": null,
  "TelnetServer": "Off",
  "SysDescr": "Cisco IOS Software, Catalyst L3 Switch Software (CAT3K_CAA-UNIVERSALK9-M), Version 15.0(1)EX2, RELEASE SOFTWARE (fc2)",
  "Technical Support": "http://www.cisco.com/techsupport",
  "Copyright": "(c) 1986-2013 by Cisco Systems, Inc.",
  "Compiled": "Fri 14-Jun-13 19:24 by pro",
  "AdministrationNodeName": null,
  "Administrator": null,
  "IPv6Address": null,
  "ApplicationPath": null,
  "SNMPCommunitySet": null,
```

```

"Memo": null,
"HardwareType": null,
"DeviceFrontPanel": null,
"SNMPCommunityGet": "public",
"IconType": "host",
"IPAddress": "192.168.10.254",
"SNMPv3EngineID": "0x80:00:00:09:03:00:dc:a5:f4:b5:8b:80",
"SNMPv3SecurityLevel": "NoAuth/NoPriv",
"Name": "sys_sw01",
"FloatingIP": null,
"URL": null,
"Alias": null,
"SNMPv3PrivacyProtocol": null,
"SNMPCharacterCode": null,
"SNMPVersion": "2C",
"ID": 66,
"RoutingControl": "forwarding",
"IPv4Interfaces": [
  {
    "Index": 1,
    "Description": "GigabitEthernet0/0",
    "MACAddress": "dc:a5:f4:b5:8b:80",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 2,
    "Description": "Null0",
    "MACAddress": null,
    "FEX_ID": null,
    "IPv4": [],
    "Type": "other(1) "
  },
  {
    "Index": 3,
    "Description": "GigabitEthernet1/0/1",
    "MACAddress": "dc:a5:f4:b5:8b:81",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 4,
    "Description": "GigabitEthernet1/0/2",
    "MACAddress": "dc:a5:f4:b5:8b:82",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 5,
    "Description": "GigabitEthernet1/0/3",
    "MACAddress": "dc:a5:f4:b5:8b:83",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
],

```

```

{
  "Index": 6,
  "Description": "GigabitEthernet1/0/4",
  "MACAddress": "dc:a5:f4:b5:8b:84",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 7,
  "Description": "GigabitEthernet1/0/5",
  "MACAddress": "dc:a5:f4:b5:8b:85",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 8,
  "Description": "GigabitEthernet1/0/6",
  "MACAddress": "dc:a5:f4:b5:8b:86",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 9,
  "Description": "GigabitEthernet1/0/7",
  "MACAddress": "dc:a5:f4:b5:8b:87",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 10,
  "Description": "GigabitEthernet1/0/8",
  "MACAddress": "dc:a5:f4:b5:8b:88",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 11,
  "Description": "GigabitEthernet1/0/9",
  "MACAddress": "dc:a5:f4:b5:8b:89",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 12,
  "Description": "GigabitEthernet1/0/10",
  "MACAddress": "dc:a5:f4:b5:8b:8a",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 13,

```

```

        "Description": "GigabitEthernet1/0/11",
        "MACAddress": "dc:a5:f4:b5:8b:8b",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 14,
        "Description": "GigabitEthernet1/0/12",
        "MACAddress": "dc:a5:f4:b5:8b:8c",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 15,
        "Description": "GigabitEthernet1/0/13",
        "MACAddress": "dc:a5:f4:b5:8b:8d",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 16,
        "Description": "GigabitEthernet1/0/14",
        "MACAddress": "dc:a5:f4:b5:8b:8e",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 17,
        "Description": "GigabitEthernet1/0/15",
        "MACAddress": "dc:a5:f4:b5:8b:8f",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 18,
        "Description": "GigabitEthernet1/0/16",
        "MACAddress": "dc:a5:f4:b5:8b:90",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 19,
        "Description": "GigabitEthernet1/0/17",
        "MACAddress": "dc:a5:f4:b5:8b:91",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 20,
        "Description": "GigabitEthernet1/0/18",
        "MACAddress": "dc:a5:f4:b5:8b:92",

```

```

        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 21,
        "Description": "GigabitEthernet1/0/19",
        "MACAddress": "dc:a5:f4:b5:8b:93",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 22,
        "Description": "GigabitEthernet1/0/20",
        "MACAddress": "dc:a5:f4:b5:8b:94",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 23,
        "Description": "GigabitEthernet1/0/21",
        "MACAddress": "dc:a5:f4:b5:8b:95",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 24,
        "Description": "GigabitEthernet1/0/22",
        "MACAddress": "dc:a5:f4:b5:8b:96",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 25,
        "Description": "GigabitEthernet1/0/23",
        "MACAddress": "dc:a5:f4:b5:8b:97",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 26,
        "Description": "GigabitEthernet1/0/24",
        "MACAddress": "dc:a5:f4:b5:8b:98",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 27,
        "Description": "GigabitEthernet1/1/1",
        "MACAddress": "dc:a5:f4:b5:8b:99",
        "FEX_ID": null,
        "IPv4": [],

```

```

        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 28,
        "Description": "GigabitEthernet1/1/2",
        "MACAddress": "dc:a5:f4:b5:8b:9a",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 29,
        "Description": "GigabitEthernet1/1/3",
        "MACAddress": "dc:a5:f4:b5:8b:9b",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 30,
        "Description": "GigabitEthernet1/1/4",
        "MACAddress": "dc:a5:f4:b5:8b:9c",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 31,
        "Description": "TenGigabitEthernet1/1/1",
        "MACAddress": "dc:a5:f4:b5:8b:9d",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 32,
        "Description": "TenGigabitEthernet1/1/2",
        "MACAddress": "dc:a5:f4:b5:8b:9e",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 33,
        "Description": "TenGigabitEthernet1/1/3",
        "MACAddress": "dc:a5:f4:b5:8b:9f",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 34,
        "Description": "TenGigabitEthernet1/1/4",
        "MACAddress": "dc:a5:f4:b5:8b:a0",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    },

```

```

{
  "Index": 35,
  "Description": "StackPort1",
  "MACAddress": null,
  "FEX_ID": null,
  "IPv4": [],
  "Type": "propVirtual(53) "
},
{
  "Index": 36,
  "Description": "StackSub-St1-1",
  "MACAddress": null,
  "FEX_ID": null,
  "IPv4": [],
  "Type": "propVirtual(53) "
},
{
  "Index": 37,
  "Description": "StackSub-St1-2",
  "MACAddress": null,
  "FEX_ID": null,
  "IPv4": [],
  "Type": "propVirtual(53) "
},
{
  "Index": 38,
  "Description": "Vlan1",
  "MACAddress": "dc:a5:f4:b5:8b:c7",
  "FEX_ID": null,
  "IPv4": [
    {
      "SubnetMask": "255.255.255.0",
      "Address": "192.168.1.254"
    }
  ],
  "Type": "propVirtual(53) "
},
{
  "Index": 39,
  "Description": "Loopback0",
  "MACAddress": null,
  "FEX_ID": null,
  "IPv4": [],
  "Type": "softwareLoopback(24) "
},
{
  "Index": 40,
  "Description": "Port-channel10",
  "MACAddress": "dc:a5:f4:b5:8b:81",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "propVirtual(53) "
},
{
  "Index": 41,
  "Description": "Port-channel15",
  "MACAddress": "dc:a5:f4:b5:8b:84",
  "FEX_ID": null,

```

```

        "IPv4": [],
        "Type": "propVirtual(53)"
    },
    {
        "Index": 42,
        "Description": "Vlan2",
        "MACAddress": "dc:a5:f4:b5:8b:f7",
        "FEX_ID": null,
        "IPv4": [
            {
                "SubnetMask": "255.255.255.0",
                "Address": "192.168.0.254"
            }
        ],
        "Type": "propVirtual(53)"
    },
    {
        "Index": 43,
        "Description": "Vlan10",
        "MACAddress": "dc:a5:f4:b5:8b:c6",
        "FEX_ID": null,
        "IPv4": [
            {
                "SubnetMask": "255.255.255.0",
                "Address": "192.168.10.254"
            }
        ],
        "Type": "propVirtual(53)"
    },
    {
        "Index": 44,
        "Description": "Vlan20",
        "MACAddress": "dc:a5:f4:b5:8b:d6",
        "FEX_ID": null,
        "IPv4": [
            {
                "SubnetMask": "255.255.255.0",
                "Address": "192.168.20.254"
            }
        ],
        "Type": "propVirtual(53)"
    },
    {
        "Index": 45,
        "Description": "Vlan30",
        "MACAddress": "dc:a5:f4:b5:8b:e5",
        "FEX_ID": null,
        "IPv4": [
            {
                "SubnetMask": "255.255.255.0",
                "Address": "172.17.0.254"
            }
        ],
        "Type": "propVirtual(53)"
    },
    {
        "Index": 46,
        "Description": "Vlan70",

```



```

        "MACAddress": "dc:a5:f4:b5:8b:f3",
        "FEX_ID": null,
        "IPv4": [
            {
                "SubnetMask": "255.255.255.0",
                "Address": "10.0.1.254"
            }
        ],
        "Type": "propVirtual(53) "
    },
    {
        "Index": 47,
        "Description": "Vlan80",
        "MACAddress": "dc:a5:f4:b5:8b:ca",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "propVirtual(53) "
    },
    {
        "Index": 48,
        "Description": "Vlan200",
        "MACAddress": "dc:a5:f4:b5:8b:e2",
        "FEX_ID": null,
        "IPv4": [
            {
                "SubnetMask": "255.255.255.0",
                "Address": "192.168.50.254"
            }
        ],
        "Type": "propVirtual(53) "
    }
],
"SysName": "C3850X_1.st.local",
"SysObjectID": "1.3.6.1.4.1.9.1.1644",
"SoftwareVersion": "15.0(1)EX2",
"LocalIPAddress": null,
"FEX_ID": null,
"DefaultTargetPort": "1,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34",
"Location": "%f\\t"
}

```

4. Updating the monitoring mode of node “sys_sw01” to "ON", and starting the monitoring of node “sys_sw01”.

Updating the property information of the node by executing the API for updating node. Refer to ["4.1.9 Updating Node \(page 51\)"](#) for the details.

Tip

To specify the default target port, specifying some port together with updating the monitoring mode in this operation.

- Request:

```

PUT /umf/fw/nvp/nodes/66 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 50

```

```
Date: Thu, 12 Nov 2015 01:04:37 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:RmA0KcT3Fq/3E5yrFqYM9aFHg2hDROsLDlH93yOhypw=

{
  "DefaultTargetPort": "1",
  "MonitoringMode": "On"
}
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 7854
Date: Thu, 12 Nov 2015 01:04:39 GMT

{
  "WebAccessPort": null,
  "SNMPv3AuthenticationProtocol": null,
  "Group": null,
  "SNMPv3UserName": null,
  "IPv6Interfaces": [],
  "AgentType": 9,
  "DPID": null,
  "SNMPv3SeverityOfInvalidEngineID": null,
  "MonitoringMode": "On",
  "DiscoveryProtocol": null,
  "OSType": "IOS",
  "SNMPPort": null,
  "TelnetServer": "Off",
  "SysDescr": "Cisco IOS Software, Catalyst L3 Switch Software (
CAT3K_CAA-UNIVERSALK9-M), Version 15.0(1)EX2, RELEASE SOFTWARE (fc
2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2013 by Cisco Systems, Inc.
Compiled Fri 14-Jun-13 19:24 by pro",
  "AdministrationNodeName": null,
  "Administrator": null,
  "IPv6Address": null,
  "ApplicationPath": null,
  "SNMPCommunitySet": null,
  "Memo": null,
  "HardwareType": null,
  "DeviceFrontPanel": null,
  "SNMPCommunityGet": "public",
  "IconType": "host",
  "IPAddress": "192.168.10.254",
  "SNMPv3EngineID": "0x80:00:00:09:03:00:dc:a5:f4:b5:8b:80",
  "SNMPv3SecurityLevel": "NoAuth/NoPriv",
  "Name": "sys_sw01",
```

```

"FloatingIP": null,
"URL": null,
"Alias": null,
"SNMPv3PrivacyProtocol": null,
"SNMPCharacterCode": null,
"SNMPVersion": "2C",
"ID": 66,
"RoutingControl": "forwarding",
"IPv4Interfaces": [
  {
    "Index": 1,
    "Description": "GigabitEthernet0/0",
    "MACAddress": "dc:a5:f4:b5:8b:80",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 2,
    "Description": "Null0",
    "MACAddress": null,
    "FEX_ID": null,
    "IPv4": [],
    "Type": "other(1) "
  },
  {
    "Index": 3,
    "Description": "GigabitEthernet1/0/1",
    "MACAddress": "dc:a5:f4:b5:8b:81",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 4,
    "Description": "GigabitEthernet1/0/2",
    "MACAddress": "dc:a5:f4:b5:8b:82",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 5,
    "Description": "GigabitEthernet1/0/3",
    "MACAddress": "dc:a5:f4:b5:8b:83",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 6,
    "Description": "GigabitEthernet1/0/4",
    "MACAddress": "dc:a5:f4:b5:8b:84",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {

```

```

        "Index": 7,
        "Description": "GigabitEthernet1/0/5",
        "MACAddress": "dc:a5:f4:b5:8b:85",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 8,
        "Description": "GigabitEthernet1/0/6",
        "MACAddress": "dc:a5:f4:b5:8b:86",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 9,
        "Description": "GigabitEthernet1/0/7",
        "MACAddress": "dc:a5:f4:b5:8b:87",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 10,
        "Description": "GigabitEthernet1/0/8",
        "MACAddress": "dc:a5:f4:b5:8b:88",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 11,
        "Description": "GigabitEthernet1/0/9",
        "MACAddress": "dc:a5:f4:b5:8b:89",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 12,
        "Description": "GigabitEthernet1/0/10",
        "MACAddress": "dc:a5:f4:b5:8b:8a",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 13,
        "Description": "GigabitEthernet1/0/11",
        "MACAddress": "dc:a5:f4:b5:8b:8b",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 14,
        "Description": "GigabitEthernet1/0/12",

```

```

    "MACAddress": "dc:a5:f4:b5:8b:8c",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 15,
    "Description": "GigabitEthernet1/0/13",
    "MACAddress": "dc:a5:f4:b5:8b:8d",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 16,
    "Description": "GigabitEthernet1/0/14",
    "MACAddress": "dc:a5:f4:b5:8b:8e",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 17,
    "Description": "GigabitEthernet1/0/15",
    "MACAddress": "dc:a5:f4:b5:8b:8f",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 18,
    "Description": "GigabitEthernet1/0/16",
    "MACAddress": "dc:a5:f4:b5:8b:90",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 19,
    "Description": "GigabitEthernet1/0/17",
    "MACAddress": "dc:a5:f4:b5:8b:91",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 20,
    "Description": "GigabitEthernet1/0/18",
    "MACAddress": "dc:a5:f4:b5:8b:92",
    "FEX_ID": null,
    "IPv4": [],
    "Type": "ethernet-csmacd(6) "
  },
  {
    "Index": 21,
    "Description": "GigabitEthernet1/0/19",
    "MACAddress": "dc:a5:f4:b5:8b:93",
    "FEX_ID": null,

```

```

        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 22,
        "Description": "GigabitEthernet1/0/20",
        "MACAddress": "dc:a5:f4:b5:8b:94",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 23,
        "Description": "GigabitEthernet1/0/21",
        "MACAddress": "dc:a5:f4:b5:8b:95",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 24,
        "Description": "GigabitEthernet1/0/22",
        "MACAddress": "dc:a5:f4:b5:8b:96",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 25,
        "Description": "GigabitEthernet1/0/23",
        "MACAddress": "dc:a5:f4:b5:8b:97",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 26,
        "Description": "GigabitEthernet1/0/24",
        "MACAddress": "dc:a5:f4:b5:8b:98",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 27,
        "Description": "GigabitEthernet1/1/1",
        "MACAddress": "dc:a5:f4:b5:8b:99",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    },
    {
        "Index": 28,
        "Description": "GigabitEthernet1/1/2",
        "MACAddress": "dc:a5:f4:b5:8b:9a",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "ethernet-csmacd(6) "
    }

```

```

},
{
  "Index": 29,
  "Description": "GigabitEthernet1/1/3",
  "MACAddress": "dc:a5:f4:b5:8b:9b",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 30,
  "Description": "GigabitEthernet1/1/4",
  "MACAddress": "dc:a5:f4:b5:8b:9c",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 31,
  "Description": "TenGigabitEthernet1/1/1",
  "MACAddress": "dc:a5:f4:b5:8b:9d",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 32,
  "Description": "TenGigabitEthernet1/1/2",
  "MACAddress": "dc:a5:f4:b5:8b:9e",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 33,
  "Description": "TenGigabitEthernet1/1/3",
  "MACAddress": "dc:a5:f4:b5:8b:9f",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 34,
  "Description": "TenGigabitEthernet1/1/4",
  "MACAddress": "dc:a5:f4:b5:8b:a0",
  "FEX_ID": null,
  "IPv4": [],
  "Type": "ethernet-csmacd(6) "
},
{
  "Index": 35,
  "Description": "StackPort1",
  "MACAddress": null,
  "FEX_ID": null,
  "IPv4": [],
  "Type": "propVirtual(53) "
},
{

```

```

        "Index": 36,
        "Description": "StackSub-St1-1",
        "MACAddress": null,
        "FEX_ID": null,
        "IPv4": [],
        "Type": "propVirtual(53) "
    },
    {
        "Index": 37,
        "Description": "StackSub-St1-2",
        "MACAddress": null,
        "FEX_ID": null,
        "IPv4": [],
        "Type": "propVirtual(53) "
    },
    {
        "Index": 38,
        "Description": "Vlan1",
        "MACAddress": "dc:a5:f4:b5:8b:c7",
        "FEX_ID": null,
        "IPv4": [
            {
                "SubnetMask": "255.255.255.0",
                "Address": "192.168.1.254"
            }
        ],
        "Type": "propVirtual(53) "
    },
    {
        "Index": 39,
        "Description": "Loopback0",
        "MACAddress": null,
        "FEX_ID": null,
        "IPv4": [],
        "Type": "softwareLoopback(24) "
    },
    {
        "Index": 40,
        "Description": "Port-channel10",
        "MACAddress": "dc:a5:f4:b5:8b:81",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "propVirtual(53) "
    },
    {
        "Index": 41,
        "Description": "Port-channel15",
        "MACAddress": "dc:a5:f4:b5:8b:84",
        "FEX_ID": null,
        "IPv4": [],
        "Type": "propVirtual(53) "
    },
    {
        "Index": 42,
        "Description": "Vlan2",
        "MACAddress": "dc:a5:f4:b5:8b:f7",
        "FEX_ID": null,
        "IPv4": [

```



```

        {
            "SubnetMask": "255.255.255.0",
            "Address": "192.168.0.254"
        }
    ],
    "Type": "propVirtual(53)"
},
{
    "Index": 43,
    "Description": "Vlan10",
    "MACAddress": "dc:a5:f4:b5:8b:c6",
    "FEX_ID": null,
    "IPv4": [
        {
            "SubnetMask": "255.255.255.0",
            "Address": "192.168.10.254"
        }
    ],
    "Type": "propVirtual(53)"
},
{
    "Index": 44,
    "Description": "Vlan20",
    "MACAddress": "dc:a5:f4:b5:8b:d6",
    "FEX_ID": null,
    "IPv4": [
        {
            "SubnetMask": "255.255.255.0",
            "Address": "192.168.20.254"
        }
    ],
    "Type": "propVirtual(53)"
},
{
    "Index": 45,
    "Description": "Vlan30",
    "MACAddress": "dc:a5:f4:b5:8b:e5",
    "FEX_ID": null,
    "IPv4": [
        {
            "SubnetMask": "255.255.255.0",
            "Address": "172.17.0.254"
        }
    ],
    "Type": "propVirtual(53)"
},
{
    "Index": 46,
    "Description": "Vlan70",
    "MACAddress": "dc:a5:f4:b5:8b:f3",
    "FEX_ID": null,
    "IPv4": [
        {
            "SubnetMask": "255.255.255.0",
            "Address": "10.0.1.254"
        }
    ],
    "Type": "propVirtual(53)"
}

```

```

    },
    {
      "Index": 47,
      "Description": "Vlan80",
      "MACAddress": "dc:a5:f4:b5:8b:ca",
      "FEX_ID": null,
      "IPv4": [],
      "Type": "propVirtual(53) "
    },
    {
      "Index": 48,
      "Description": "Vlan200",
      "MACAddress": "dc:a5:f4:b5:8b:e2",
      "FEX_ID": null,
      "IPv4": [
        {
          "SubnetMask": "255.255.255.0",
          "Address": "192.168.50.254"
        }
      ],
      "Type": "propVirtual(53) "
    }
  ],
  "SysName": "C3850X_1.st.local",
  "SysObjectID": "1.3.6.1.4.1.9.1.1644",
  "SoftwareVersion": "15.0(1)EX2",
  "LocalIPAddress": null,
  "FEX_ID": null,
  "DefaultTargetPort": "1",
  "Location": "%f\\t"
}

```

5.1.4 Updating Group Information that Node belonging

This subsection describes the procedure to update the information of the group that the existing node belongs.

In the environment that a group is specified as the monitoring target in the status monitoring rule entry and the data collection entry, this operation is executed to add a node to or delete a node from the monitoring target of the entries .

In this subsection, the procedure is described using the following operation example.

Example: Updating the group that node “*bo_r01*” belongs from group “*Branch A*” to group “*bo_router*”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of node “*bo_r01*”.

Obtaining node IDs by executing the API for obtaining node list. Refer to ["4.1.6 Obtaining Node List \(page 44\)"](#) for the detail.

Tip

It is possible to narrow down target nodes by specifying a node name in NameFilter query parameter.

- Request:

```
GET /umf/fw/nvp/nodes?NameFilter=bo_r01 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 02:36:57 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:NO15ZdIobkFbM/iZ4xshyGin8dpz8BACyLcRJ0A6zHs=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 27
Date: Thu, 12 Nov 2015 02:36:59 GMT

[
  {
    "Name": "bo_r01",
    "ID": 68
  }
]
```

2. Updating the group that node “*bo_r01*” belongs to group “*bo_router*”.

Updating the property information of the node by executing the API for updating node. Refer to ["4.1.9 Updating Node \(page 51\)"](#) for the details.

- Request:

```
PUT /umf/fw/nvp/nodes/68 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 22
Date: Thu, 12 Nov 2015 02:38:12 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:DavXMc/B7nB4+BxCxZrrkmt3kQ1Ft9oKVNJNThCallY=

{
  "Group": "bo_router"
}
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
```

```
Content-Type: application/json;charset=UTF-8
Content-Length: 910
Date: Thu, 12 Nov 2015 02:38:14 GMT
```

```
{
  "WebAccessPort": null,
  "SNMPv3AuthenticationProtocol": null,
  "Group": "bo_router",
  "SNMPv3UserName": null,
  "IPv6Interfaces": [],
  "AgentType": null,
  "DPID": null,
  "SNMPv3SeverityOfInvalidEngineID": null,
  "MonitoringMode": "Off",
  "DiscoveryProtocol": null,
  "OSType": null,
  "SNMPPort": null,
  "TelnetServer": "Off",
  "SysDescr": null,
  "AdministrationNodeName": null,
  "Administrator": null,
  "IPv6Address": null,
  "ApplicationPath": null,
  "SNMPCommunitySet": null,
  "Memo": null,
  "HardwareType": null,
  "DeviceFrontPanel": null,
  "SNMPCommunityGet": null,
  "IconType": "host",
  "IPAddress": null,
  "SNMPv3EngineID": null,
  "SNMPv3SecurityLevel": "NoAuth/NoPriv",
  "Name": "bo_r01",
  "FloatingIP": null,
  "URL": null,
  "Alias": null,
  "SNMPv3PrivacyProtocol": null,
  "SNMPCharCode": null,
  "SNMPVersion": null,
  "ID": 68,
  "RoutingControl": null,
  "IPv4Interfaces": [],
  "SysName": null,
  "SysObjectID": null,
  "SoftwareVersion": null,
  "LocalIPAddress": null,
  "FEX_ID": null,
  "DefaultTargetPort": null,
  "Location": null
}
```

5.1.5 Suspending Node Monitoring belonging to Specified Map

This subsection describes the procedure to temporally stop all the set monitoring for all nodes belonging to the specified map.

This operation is executed to stop all monitoring during a device maintenance or a network construction without updating each monitoring setting.

In this subsection, the procedure is described using the following operation example.

Example: Temporally stopping the monitoring of all nodes belonging to map “*General Affairs Department*”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of map “*General Affairs Department*”.

Obtaining map IDs by executing the API for obtaining map list. Refer to ["4.1.1 Obtaining Map List \(page 40\)"](#) for the details.

Tip

It is possible to narrow down target maps by specifying a map name in NameFilter query parameter.

Request:

```
GET /umf/fw/nvp/maps?NameFilter=%E7%B7%8F%E5%8B%99%E9%83%A8 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 02:45:37 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfPpdVj
8U=:BMD+T+hFdPc8l6GwexEiOHxecNwdob8BzLWKXcKEaI8=
```

Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 30
Date: Thu, 12 Nov 2015 02:45:39 GMT

[
  {
    "Name": "General Affairs Department",
    "ID": 77
  }
]
```

2. Looking up IDs of nodes belonging to map “*General Affairs Department*”.

Obtaining node IDs by executing the API for obtaining node list. Refer to ["4.1.6 Obtaining Node List \(page 44\)"](#) for the details.

Tip

It is possible to narrow down target nodes by setting ID of map “*General Affairs Department*” in MapFilter query parameter.

Request:

```
GET /umf/fw/nvp/nodes?MapFilter=77 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 02:46:30 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfPpdVj
8U=:FedNy0JbAi7/EgOabLyblhESeNz2SirhGWQ9OBizV3g=
```

Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 103
Date: Thu, 12 Nov 2015 02:46:32 GMT
```

```
[
  {
    "Name": "soumu_router.0",
    "ID": 79
  },
  {
    "Name": "soumu_router.1",
    "ID": 81
  },
  {
    "Name": "soumu_router.2",
    "ID": 83
  }
]
```

3. Stopping the monitoring by updating the monitoring mode of nodes belonging to map “General Affairs Department” to “OFF”.

Updating the property information one by one by executing ["4.1.9 Updating Node \(page 51\)"](#).

Tip

The function to update the properties of all target nodes in a batch is not provided. Therefore, it needs to update the property information of all target nodes by executing ["4.1.9 Updating Node \(page 51\)"](#) repeatedly.

- a. • Request:

```
PUT /umf/fw/nvp/nodes/79 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 25
Date: Thu, 12 Nov 2015 02:52:06 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
```

```

Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD
1jfPpdVj8U=:eEN7874iy9dJwnfhFsCcPdhpY4t+tM3YiaELErMqhGU=

{
    "MonitoringMode": "Off"
}

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json;charset=UTF-8
Content-Length: 916
Date: Thu, 12 Nov 2015 02:52:08 GMT

{
    "WebAccessPort": null,
    "SNMPv3AuthenticationProtocol": null,
    "Group": null,
    "SNMPv3UserName": null,
    "IPv6Interfaces": [],
    "AgentType": null,
    "DPID": null,
    "SNMPv3SeverityOfInvalidEngineID": null,
    "MonitoringMode": "Off",
    "DiscoveryProtocol": null,
    "OSType": null,
    "SNMPPort": null,
    "TelnetServer": "Off",
    "SysDescr": null,
    "AdministrationNodeName": null,
    "Administrator": null,
    "IPv6Address": null,
    "ApplicationPath": null,
    "SNMPCommunitySet": null,
    "Memo": null,
    "HardwareType": null,
    "DeviceFrontPanel": null,
    "SNMPCommunityGet": null,
    "IconType": "host",
    "IPAddress": "1.0.0.0",
    "SNMPv3EngineID": null,
    "SNMPv3SecurityLevel": "NoAuth/NoPriv",
    "Name": "soumu_router.0",
    "FloatingIP": null,
    "URL": null,
    "Alias": null,
    "SNMPv3PrivacyProtocol": null,
    "SNMPCharacterCode": null,
    "SNMPVersion": null,
    "ID": 79,
    "RoutingControl": null,
    "IPv4Interfaces": [],

```

```

    "SysName": null,
    "SysObjectID": null,
    "SoftwareVersion": null,
    "LocalIPAddress": null,
    "FEX_ID": null,
    "DefaultTargetPort": null,
    "Location": null
  }

```

b. • Request:

```

PUT /umf/fw/nvp/nodes/81 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 25
Date: Thu, 12 Nov 2015 02:55:05 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD
1jfpPdvj8U=:vBonBhJeR7TqNEsRb1lvelV7TfA8gI+pZAlpEtg5tck=

{
  "MonitoringMode": "Off"
}

```

• Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 916
Date: Thu, 12 Nov 2015 02:55:07 GMT

{
  "WebAccessPort": null,
  "SNMPv3AuthenticationProtocol": null,
  "Group": null,
  "SNMPv3UserName": null,
  "IPv6Interfaces": [],
  "AgentType": null,
  "DPID": null,
  "SNMPv3SeverityOfInvalidEngineID": null,
  "MonitoringMode": "Off",
  "DiscoveryProtocol": null,
  "OSType": null,
  "SNMPPort": null,
  "TelnetServer": "Off",
  "SysDescr": null,
  "AdministrationNodeName": null,
  "Administrator": null,
  "IPv6Address": null,
  "ApplicationPath": null,

```



```

    "SNMPCommunitySet": null,
    "Memo": null,
    "HardwareType": null,
    "DeviceFrontPanel": null,
    "SNMPCommunityGet": null,
    "IconType": "host",
    "IPAddress": "1.0.0.1",
    "SNMPv3EngineID": null,
    "SNMPv3SecurityLevel": "NoAuth/NoPriv",
    "Name": "soumu_router.1",
    "FloatingIP": null,
    "URL": null,
    "Alias": null,
    "SNMPv3PrivacyProtocol": null,
    "SNMPCharacterCode": null,
    "SNMPVersion": null,
    "ID": 81,
    "RoutingControl": null,
    "IPv4Interfaces": [],
    "SysName": null,
    "SysObjectID": null,
    "SoftwareVersion": null,
    "LocalIPAddress": null,
    "FEX_ID": null,
    "DefaultTargetPort": null,
    "Location": null
}

```

c. • Request:

```

PUT /umf/fw/nvp/nodes/83 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 25
Date: Thu, 12 Nov 2015 02:56:29 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD
1jfPpdVj8U=:aGpXq4LpMIiV4nKYLqijmJfYMBwsbQKRQ3b93sR9hNg=

{
    "MonitoringMode": "Off"
}

```

• Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 916
Date: Thu, 12 Nov 2015 02:56:31 GMT

{

```

```

    "WebAccessPort": null,
    "SNMPv3AuthenticationProtocol": null,
    "Group": null,
    "SNMPv3UserName": null,
    "IPv6Interfaces": [],
    "AgentType": null,
    "DPID": null,
    "SNMPv3SeverityOfInvalidEngineID": null,
    "MonitoringMode": "Off",
    "DiscoveryProtocol": null,
    "OSType": null,
    "SNMPPort": null,
    "TelnetServer": "Off",
    "SysDescr": null,
    "AdministrationNodeName": null,
    "Administrator": null,
    "IPv6Address": null,
    "ApplicationPath": null,
    "SNMPCommunitySet": null,
    "Memo": null,
    "HardwareType": null,
    "DeviceFrontPanel": null,
    "SNMPCommunityGet": null,
    "IconType": "host",
    "IPAddress": "1.0.0.2",
    "SNMPv3EngineID": null,
    "SNMPv3SecurityLevel": "NoAuth/NoPriv",
    "Name": "soumu_router.2",
    "FloatingIP": null,
    "URL": null,
    "Alias": null,
    "SNMPv3PrivacyProtocol": null,
    "SNMPCharCode": null,
    "SNMPVersion": null,
    "ID": 83,
    "RoutingControl": null,
    "IPv4Interfaces": [],
    "SysName": null,
    "SysObjectID": null,
    "SoftwareVersion": null,
    "LocalIPAddress": null,
    "FEX_ID": null,
    "DefaultTargetPort": null,
    "Location": null
}

```

5.1.6 Deleting Node

This subsection describes the procedure to delete the node that is not needed to be managed.

This operation is executed to exclude the node that is not needed to be managed from management targets due to causes such as reviewing facilities.

In this subsection, the procedure is described using the following operation example.

Example: Deleting all information of node “*bo_r10*” that is not needed to be managed.

⚠ Caution

In the status monitoring rule entry or the data collection entry, when specifying the node name of the deleting target as the monitoring target, it is recommended to update the contents of the monitoring target of the entry before deleting the node.

The contents specifying the monitoring target node in the entry is not updated even when the node is deleted.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of node “*bo_r10*” node.

Obtaining node IDs by executing the API for obtaining node list. Refer to ["4.1.6 Obtaining Node List \(page 44\)"](#) for the details.

Tip

It is possible to narrow down target nodes by specifying a node name in NameFilter query parameter.

- Request:

```
GET /umf/fw/nvp/nodes?NameFilter=bo_r10 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 03:03:20 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:i4Z6/eqH2PiAFsKAozAT06vcYBtqlTfl+w2LArs61o8=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 28
Date: Thu, 12 Nov 2015 03:03:22 GMT

[
  {
    "Name": "bo_r10",
    "ID": 113
  }
]
```

2. Deleting all information of node “*bo_r10*”.

Deleting the information of the node by executing the API for deleting node. Refer to ["4.1.10 Deleting Node \(page 55\)"](#) for the details.

Tip

It is possible to delete all information by setting "true" in All query parameter even when the icon of the node is registered in multiple maps.

- Request:

```
DELETE /umf/fw/nvp/nodes/113?All=true HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 03:05:00 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:jThrSey06kT+xWfue8aOBEw0NIz/+v38vVjjJhFI7KU=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Content-Length: 0
Date: Thu, 12 Nov 2015 03:05:02 GMT
```

5.1.7 Confirming Fault Condition of Nodes belonging to Specified Map

This subsection describes the procedure to confirm fault conditions of all nodes belonging to the specified map.

This operation is executed to confirm whether the current status of the management target node is normal or not.

In this subsection, the procedure is described using the following operation example.

Example: Confirming whether an alert whose severity level is higher than Warning is occurring or not for all nodes belonging to map "Headquarters".

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of map "Headquarters".

Obtaining map IDs by executing the API for obtaining map list. Refer to ["4.1.1 Obtaining Map List \(page 40\)"](#) for the details.

Tip

It is possible to narrow down target maps by specifying a map name in NameFilter query parameter.

- Request:

```
GET /umf/fw/nvp/maps?NameFilter=%E6%9C%AC%E7%A4%BE HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 03:07:18 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
```

```
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:accoWwkdYATIlo/A3eNj5xW7Q6KoepDOHR3JOQtkSfg=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 28
Date: Thu, 12 Nov 2015 03:07:20 GMT

[
  {
    "Name": "Headquarters",
    "ID": 120
  }
]
```

2. Confirming whether an alert whose severity level is higher than Warning is occurring or not for all nodes belonging to map *"Headquarters"*.

Obtaining the severity status information of nodes by executing the API for obtaining severity state list of node. Refer to ["4.1.12 Obtaining Severity Status List of Node \(page 58\)"](#) for the details.

Tip

- It is possible to obtain the information of nodes whose severity level is higher than Warning by setting Warning in Severity query parameter.
- It is possible to narrow down the range of obtained information by specifying a map ID in MapFilter query parameter.

- Request:

```
GET /umf/fw/nvp/nodes/status?MapFilter=120 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 03:34:29 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:BHIYi06+jNwJ6FUwk3K9X6mjahmBDC11q22aSrGmmE=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
```

```

Content-Length: 181
Date: Thu, 12 Nov 2015 03:34:31 GMT

[
  {
    "Status": "Warning",
    "Type": "node",
    "Name": "main.0",
    "ID": 124
  },
  {
    "Status": "Warning",
    "Type": "node",
    "Name": "main.1",
    "ID": 126
  },
  {
    "Status": "Warning",
    "Type": "node",
    "Name": "main.2",
    "ID": 128
  }
]

```

5.2 Operating Status Monitoring Rule Entry

This section describes the way to operate the status monitoring rule entry by using each API.

5.2.1 Adding New Status Monitoring Rule Entry

This subsection describes the procedure to add a new status monitoring rule entry.

This operation is executed to add a new status monitoring rule entry with addition of a new node and reviewing of monitoring contents, and so on.

In this subsection, the procedure is described using the following operation example.

Example: Monitoring **updown:UpDownCheck** of nodes belonging to group “*bo_router*”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Confirming that status monitoring rule **updown:UpDownCheck** is built-in MasterScope Network Manager in operation just in case.

Obtaining the list of the status monitoring rules built-in NervisorPro by executing the API for obtaining status monitoring rule. Refer to ["4.2.6 Obtaining Status Monitoring Rule List \(page 67\)"](#) for the details.

Tip

It is recommended to execute this confirmation process in the operation environment that a new status monitoring rule is added or an existing status monitoring rule is removed.

- Request:

```

GET /umf/fw/nvp/stsmon/rules HTTP/1.1
Host: 192.168.10.121:12080

```

```

Accept-Encoding: identity
Date: Thu, 12 Nov 2015 03:42:26 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:BrQDqZ803XfwHIRP/cuY8IYrO59eZ/UtMZkSeQfcnFg=

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 1226
Date: Thu, 12 Nov 2015 03:42:28 GMT

```

```

[
  {
    "Status": "Attached",
    "Name": "AverageBusy5m_for_Catalyst"
  },
  {
    "Status": "Attached",
    "Name": "CpuLoad_for_IP8800"
  },
  {
    "Status": "Attached",
    "Name": "icmperr"
  },
  {
    "Status": "Attached",
    "Name": "ifDescr"
  },
  {
    "Status": "Attached",
    "Name": "ifdown"
  },
  {
    "Status": "Attached",
    "Name": "ifload64"
  },
  {
    "Status": "Attached",
    "Name": "ifload"
  },
  {
    "Status": "Attached",
    "Name": "ifName"
  },
  {
    "Status": "Attached",
    "Name": "ifOper"
  },
  {

```

```

        "Status": "Attached",
        "Name": "ifup"
    },
    {
        "Status": "Attached",
        "Name": "mw0521alertcheck"
    },
    {
        "Status": "Attached",
        "Name": "updown"
    },
    {
        "Status": "Attached",
        "Name": "mw0522alertcheck"
    },
    {
        "Status": "Attached",
        "Name": "valchange"
    },
    {
        "Status": "Attached",
        "Name": "nvtp-bandchk"
    },
    {
        "Status": "Attached",
        "Name": "icmperrv6"
    },
    {
        "Status": "Attached",
        "Name": "nvtp-stpstat"
    },
    {
        "Status": "Attached",
        "Name": "updownv6"
    },
    {
        "Status": "Attached",
        "Name": "nvtp-topchk"
    },
    {
        "Status": "Attached",
        "Name": "host_cpuload"
    },
    {
        "Status": "Attached",
        "Name": "PortCheck_for_Catalyst2900"
    },
    {
        "Status": "Attached",
        "Name": "host_disk_usage"
    },
    {
        "Status": "Attached",
        "Name": "PortCheck_for_Catalyst"
    },
    {
        "Status": "Attached",
        "Name": "host_pmem_usage"
    }

```



```

    },
    {
      "Status": "Attached",
      "Name": "snmpchk"
    },
    {
      "Status": "Attached",
      "Name": "host_vmem_usage"
    },
    {
      "Status": "Attached",
      "Name": "thresh"
    },
    {
      "Status": "Attached",
      "Name": "host_process_check"
    }
  ]

```

2. Adding a new status monitoring rule entry.

Adding a status monitoring rule entry by executing the API for adding status monitoring rule entry. Refer to ["4.2.3 Adding Status Monitoring Rule Entry \(page 64\)"](#) for the details.

- Request:

```

POST /umf/fw/nvp/stsmon/entries HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 85
Date: Thu, 12 Nov 2015 03:44:58 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:uXmQB+CQ57gpB+T76x7WF/NDzFMAHDkyMpSUFGDg8hg=

{ "Status": "Running", "Nodes": "grp:bo_router", "Name": "tutorial
", "Rule": "updown" }

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 152
Date: Thu, 12 Nov 2015 03:45:00 GMT

{
  "Status": "Running",
  "Name": "tutorial",
  "Interval": "5m",
  "Rule": "updown",
  "Arguments": [
    "",

```

```

        "",
        ""
    ],
    "AlertSeverity": "Warning",
    "Nodes": "grp:bo_router",
    "ID": 138
}

```

Tip

Executing the API for expanding standard component name specification format to confirm actually monitored nodes when a group is specified as the monitoring target. Refer to ["4.1.15 Expanding Standard Component Name Specification Format \(page 62\)"](#) for the details.

- Request:

```

GET /umf/fw/nvp/nodes/std-expand?Format=grp%3Abo_router HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 04:54:22 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfPpdVj8U=:Ge7orkxA2L9ReDkeYAjSTdIwl39uJcE88BuJ2KHZAN0=

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 42
Date: Thu, 12 Nov 2015 04:54:25 GMT

[
  {
    "Type": "node",
    "Name": "h_sw20",
    "ID": 145
  }
]

```

5.2.2 Updating Monitoring Target of Status Monitoring Rule Entry

This subsection describes the procedure to update the monitoring target of the existing status monitoring rule entry.

This operation is executed when the update of the contents of the existing status monitoring rule entry is needed with addition or delete of nodes, and so on.

In this subsection, the procedure is described using the following operation example.

Example: Adding nodes “h_sw20” and “h_sw21” to monitoring targets of the status monitoring rule entry whose title is “Headquarters_Alive monitoring”.

Tip

It is possible to add and delete monitoring targets in the procedure described in "[5.1.4 Updating Group Information that Node belonging \(page 116\)](#)" without following the procedure described in this subsection for the status monitoring rule entry that specifies a group as its monitoring target.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Confirming ID of the status monitoring rule entry whose title is "*Headquarters_Alive monitoring*".

Obtaining the list information of the registered status monitoring rule entry by executing the API for obtaining status monitoring rule entry list. Refer to "[4.2.1 Obtaining Status Monitoring Rule Entry List \(page 63\)](#)" for the details.

Looking up the status monitoring rule entry whose title(Name) is "*Headquarters_Alive monitoring*" and obtaining its ID based on the obtained information.

- Request:

```
GET /umf/fw/nvp/stsmon/entries HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 03:49:03 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:/iN9++odRJtqb7hNFzSYyTXfm+Sn5yZVs6H0/mJ6pWc=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 423
Date: Thu, 12 Nov 2015 03:49:06 GMT

[
  {
    "Status": "Stopped",
    "Name": "",
    "Interval": "5m",
    "Rule": "snmpchk",
    "Arguments": [],
    "AlertSeverity": "Warning",
    "Nodes": "*",
    "ID": 25
  },
  {
    "Status": "Running",
    "Name": "Headquarters_Alive monitoring",
    "Interval": "5m",
```

```

        "Rule": "updown",
        "Arguments": [
            "",
            "",
            ""
        ],
        "AlertSeverity": "Warning",
        "Nodes": "h_sw10,h_sw11",
        "ID": 148
    },
    {
        "Status": "Running",
        "Name": "",
        "Interval": "2s",
        "Rule": "updown",
        "Arguments": [
            "",
            "",
            ""
        ],
        "AlertSeverity": "Warning",
        "Nodes": "*",
        "ID": 118
    }
]

```

2. Updating the monitoring status of the status monitoring rule entry whose title is *"Headquarters_Alive monitoring"* to the status of monitored.

Updating the monitoring status(Status) of the status monitoring rule entry to the status of not monitored(Stopped) by executing the API for updating status monitoring rule entry. Refer to ["4.2.4 Updating Status Monitoring Rule Entry \(page 65\)"](#) for the details.

Tip

To update the property contents of the status monitoring rule entry, it needs to update the monitoring status(Status) of the target status monitoring rule entry to the status of not monitored(Stopped).

- Request:

```

PUT /umf/fw/nvp/stsmon/entries/148 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 21
Date: Thu, 12 Nov 2015 03:49:49 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:7AAGBnBiL7ngfYeT5eCC6/6luw0ni2LKG9lFrk4tJk4=

{
    "Status": "Stopped"
}

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0

```

```

Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 163
Date: Thu, 12 Nov 2015 03:49:52 GMT

{
  "Status": "Stopped",
  "Name": "Headquarters_Alive monitoring",
  "Interval": "5m",
  "Rule": "updown",
  "Arguments": [
    "",
    "",
    ""
  ],
  "AlertSeverity": "Warning",
  "Nodes": "h_sw10,h_sw11",
  "ID": 148
}

```

3. Updating the monitoring status to the status of monitored by adding nodes “h_sw20” and “h_sw21” to monitoring targets of the status monitoring rule entry whose title is “Headquarters_Alive monitoring”.

Updating each property and the monitoring status(Status) of the status monitoring rule entry to the status of monitored(Running) by executing the API for updating status monitoring rule entry. Refer to ["4.2.4 Updating Status Monitoring Rule Entry \(page 65\)"](#) for the details.

- Request:

```

PUT /umf/fw/nvp/stsmon/entries/148 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 61
Date: Thu, 12 Nov 2015 03:51:17 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:lweQrlfYPHjF1KUJgzgJIEBuR76+dWWlqeP2WKYZlno=

{
  "Status": "Running",
  "Nodes": "h_sw10,h_sw11,h_sw20,h_sw21"
}

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 177

```

```
Date: Thu, 12 Nov 2015 03:51:19 GMT

{
  "Status": "Running",
  "Name": "Headquarters_Alive monitoring",
  "Interval": "5m",
  "Rule": "updown",
  "Arguments": [
    "",
    "",
    ""
  ],
  "AlertSeverity": "Warning",
  "Nodes": "h_sw10,h_sw11,h_sw20,h_sw21",
  "ID": 148
}
```

5.2.3 Deleting Status Monitoring Rule Entry

This subsection describes the procedure to delete the status monitoring rule entry that is no longer needed.

This operation is executed when the status monitoring rule entry becomes unnecessary due to delete of the monitoring target node, and so on.

In this subsection, the procedure is described using the following operation example.

Example: Deleting the status monitoring rule entry that specifies “test_*” as its monitoring targets.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of the status monitoring rule entry that specifies “test_*” as its monitoring targets.

Obtaining the list information of the registered status monitoring rule entry by executing the API for obtaining status monitoring rule entry list. Refer to ["4.2.1 Obtaining Status Monitoring Rule Entry List \(page 63\)"](#) for the details.

Looking up the status monitoring rule entry that specifies “test_*” as its monitoring targets(Nodes) and obtaining its ID based on the obtained information.

- Request:

```
GET /umf/fw/nvp/stsmon/entries HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 04:16:35 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:XOQLED7gIO+yCtaNFw9o6f/aAheQDsT/rGR0g2Vr5EA=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
```

```

X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 529
Date: Thu, 12 Nov 2015 04:16:37 GMT

```

```

[
  {
    "Status": "Stopped",
    "Name": "",
    "Interval": "5m",
    "Rule": "snmpchk",
    "Arguments": [],
    "AlertSeverity": "Warning",
    "Nodes": "*",
    "ID": 25
  },
  {
    "Status": "Running",
    "Name": "",
    "Interval": "5m",
    "Rule": "updown",
    "Arguments": [
      "",
      "",
      ""
    ],
    "AlertSeverity": "Warning",
    "Nodes": "test_*", "ID": 153
  },
  {
    "Status": "Stopped",
    "Name": "",
    "Interval": "5m",
    "Rule": "ifdown",
    "Arguments": [
      ""
    ],
    "AlertSeverity": "Warning",
    "Nodes": "test_*", "ID": 154
  },
  {
    "Status": "Running",
    "Name": "",
    "Interval": "2s",
    "Rule": "updown",
    "Arguments": [
      "",
      "",
      ""
    ],
    "AlertSeverity": "Warning",
    "Nodes": "*",
    "ID": 118
  }
]

```

2. Updating the monitoring status of the status monitoring rule entry that specifies “test_*” as its monitoring targets to the status of monitored.

Updating the monitoring status(Status) of the status monitoring rule entry to the status of not monitored(Stopped) by executing the API for updating status monitoring rule entry. Refer to ["4.2.4 Updating Status Monitoring Rule Entry \(page 65\)"](#) for the details.

Tip

It is not allowed to delete the status monitoring rule entry when the monitoring status(Status) of the status monitoring rule entry is in the status of monitored(Running).

- a. • Request:

```
PUT /umf/fw/nvp/stsmon/entries/153 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 21
Date: Thu, 12 Nov 2015 04:17:53 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD
1jfPpdVj8U=:vqCJ8U68EUjtIO5sYXDvVn1wMQtPxxyu8whnyuYlpY=

{
  "Status": "Stopped"
}
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 137
Date: Thu, 12 Nov 2015 04:17:55 GMT

{
  "Status": "Stopped",
  "Name": "",
  "Interval": "5m",
  "Rule": "updown",
  "Arguments": [
    "",
    "",
    ""
  ],
  "AlertSeverity": "Warning",
  "Nodes": "test_*",
  "ID": 153
}
```

- b. • Request:


```
PUT /umf/fw/nvp/stsmon/entries/154 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 21
Date: Thu, 12 Nov 2015 04:18:17 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD
1jfPpdVj8U=:yugicBjqDPtHDUUTMg5M42/7SBR1li1sol3ApI2mCz4=

{
  "Status": "Stopped"
}
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 131
Date: Thu, 12 Nov 2015 04:18:20 GMT

{
  "Status": "Stopped",
  "Name": "",
  "Interval": "5m",
  "Rule": "ifdown",
  "Arguments": [
    ""
  ],
  "AlertSeverity": "Warning",
  "Nodes": "test_*",
  "ID": 154
}
```

3. Deleting the status monitoring rule entry that specifies “test_*” as its monitoring targets.

Deleting the target status monitoring rule entry by executing the API for deleting status monitoring rule entry. Refer to ["4.2.5 Deleting Status Monitoring Rule Entry \(page 67\)"](#) for the details.

- a. • Request:

```
DELETE /umf/fw/nvp/stsmon/entries/153 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 04:18:41 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD
1jfPpdVj8U=:DIWYu67QtF/8W4a4ooj17XUhnU/QaV5XeXzQMGuYuLQ=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Content-Length: 0
Date: Thu, 12 Nov 2015 04:18:43 GMT
```

- b. • Request:

```
DELETE /umf/fw/nvp/stsmon/entries/154 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 04:18:44 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD
1jfPpdVj8U=:ITvBliVtHNLdzn13K0PBySIujKt0tDf/+WdysRGCuCO=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Content-Length: 0
Date: Thu, 12 Nov 2015 04:18:47 GMT
```

5.3 Operating Data Collection Entry

This section describes the way to operate data collection entries using each API.

5.3.1 Adding New Data Collection Entry

This subsection describes the procedure to add a new data collection entry.

This operation is executed to add a new data collection entry with addition of a new node and reviewing of monitoring contents, and so on.

In this subsection, the procedure is described using the following operation example.

Example:

Collecting “*cpmCPUTotal5minRev(1.3.6.1.4.1.9.9.109.1.1.1.1.8)*” and monitoring the threshold based on **General** rule for the nodes whose node name is “*sys_sw**”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Adding a new data collection entry.

Adding a data collection entry by using the API for adding data collection entry. Refer to ["4.3.3 Adding Data Collection Entry \(page 70\)"](#) for the details.

- Request:

```
POST /umf/fw/nvp/datacol/entries HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 214
Date: Thu, 12 Nov 2015 04:25:05 GMT
```

```
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:AEnar53JlQvz0eEK/guvs1ekf40QXWAhG8WrmX55QMw=

{ "Status": "Running", "MIB": "1.3.6.1.4.1.9.9.109.1.1.1.1.8", "ReportType": "General", "Name": "test", "Format": "Percent", "Threshold": "80", "Nodes": "sys_sw*", "RecoveryThreshold": "75", "OperationMode": "Both" }
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 406
Date: Thu, 12 Nov 2015 04:25:08 GMT

{
  "Status": "Running",
  "Interval": "10m",
  "ReportType": "General",
  "RiseOrFall": "Rise",
  "ContinuousTimes": 1,
  "Format": "Percent",
  "ReportDataOutput": "On",
  "RecoveryThreshold": "75",
  "MIB": "1.3.6.1.4.1.9.9.109.1.1.1.1.8",
  "AlertSeverity": "Warning",
  "MibExprName": null,
  "Instances": null,
  "AlertOutputCondition": "Continuous",
  "Threshold": "80",
  "RecoveryAlert": "On",
  "Nodes": "sys_sw*",
  "OperationMode": "Both",
  "ID": 156,
  "Name": "test"
}
```

Tip

Executing the API for expanding standard component name specification format to confirm actually monitored nodes when the monitoring target is specified with in the format of standard component name specification. Refer to ["A.2 Standard Component Name Specification Format \(page 155\)"](#) and ["4.1.15 Expanding Standard Component Name Specification Format \(page 62\)"](#) for the details.

Request:

```
GET /umf/fw/nvp/nodes/std-expand?Format=sys_sw%2A HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 04:28:40 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
```

```
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfPpdVj8U=:ROvC5KdVaum3MDRpWKBIPWex76va1/1045klvJ0pfuE=
```

Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json;charset=UTF-8
Content-Length: 44
Date: Thu, 12 Nov 2015 04:28:42 GMT

[
  {
    "Type": "node",
    "Name": "sys_sw10",
    "ID": 140
  }
]
```

5.3.2 Updating Collection Interval of Data Collection Entry

This subsection describes the procedure to update the collection interval of the existing data collection entry.

This operation is executed when the update of the contents of the existing data collection entry is needed with reviewing the monitored contents.

In this subsection, the procedure is described using the following operation example.

Example: Updating the collection interval of the data collection entry whose title is “*WAN Traffic of Branch*” from “*10 minutes*” to “*5 minutes*”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of the data collection entry whose title is “*WAN Traffic of Branch*”.

Obtaining the list information of the registered data collection entry by executing the API for obtaining data collection entry list. Refer to ["4.3.1 Obtaining Data Collection Entry List \(page 68\)"](#) for the details.

Looking up the data collection entry whose title(Name) is “*WAN Traffic of Branch*” and obtaining its ID based on the obtained information.

- Request:

```
GET /umf/fw/nvp/datacol/entries HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 04:32:29 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfPpdVj8U=:LfIXqjWoQKAr7vdbvbxRCDenCnWvO4M2B6Rx71cc3Eo=
```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json;charset=UTF-8
Content-Length: 787
Date: Thu, 12 Nov 2015 04:32:32 GMT

[
  {
    "Status": "Running",
    "Interval": "10m",
    "ReportType": "General",
    "RiseOrFall": "Rise",
    "ContinuousTimes": 1,
    "Format": "Percent",
    "ReportDataOutput": "On",
    "RecoveryThreshold": "75",
    "MIB": "1.3.6.1.4.1.9.9.109.1.1.1.1.8",
    "AlertSeverity": "Warning",
    "MibExprName": null,
    "Instances": null,
    "AlertOutputCondition": "Continuous",
    "Threshold": "80",
    "RecoveryAlert": "On",
    "Nodes": "sys_sw*",
    "OperationMode": "Both",
    "ID": 156,
    "Name": "test"
  },
  {
    "Status": "Running",
    "Interval": "10m",
    "ReportType": "TrafficOfWAN",
    "RiseOrFall": null,
    "ContinuousTimes": null,
    "Format": null,
    "ReportDataOutput": "On",
    "RecoveryThreshold": null,
    "MIB": null,
    "AlertSeverity": null,
    "MibExprName": null,
    "Instances": "*",
    "AlertOutputCondition": null,
    "Threshold": null,
    "RecoveryAlert": null,
    "Nodes": "*",
    "OperationMode": "Store",
    "ID": 158,
    "Name": "WAN Traffic of Branch"
  }
]
```

2. Updating the collection status of the data collection entry whose title is “*WAN Traffic of Branch*” to the status of not collecting.

Updating the collection status(Status) of the data collection entry to the status of not collecting(Stopped) by executing the API for updating data collection entry. Refer to ["4.3.4 Updating Data Collection Entry \(page 74\)"](#) for the details.

Tip

To update properties of the data collection entry, it needs to update the collection status(Status) of the target data collection entry to the status of not collecting.

- Request:

```
PUT /umf/fw/nvp/datacol/entries/158 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 21
Date: Thu, 12 Nov 2015 04:32:42 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:xdC5UXCvfUy//a62kvMu6rGGyUfE80pllX6cAmdyaVE=

{
  "Status": "Stopped"
}
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 378
Date: Thu, 12 Nov 2015 04:32:45 GMT

{
  "Status": "Stopped",
  "Interval": "10m",
  "ReportType": "TrafficOfWAN",
  "RiseOrFall": null,
  "ContinuousTimes": null,
  "Format": null,
  "ReportDataOutput": "On",
  "RecoveryThreshold": null,
  "MIB": null,
  "AlertSeverity": null,
  "MibExprName": null,
  "Instances": "*",
  "AlertOutputCondition": null,
  "Threshold": null,
  "RecoveryAlert": null,
  "Nodes": "*"
}
```

```

    "OperationMode": "Store",
    "ID": 158,
    "Name": "WAN Traffic of Branch"
}

```

3. Updating the collection status to the status of collecting by updating the collection interval of the data collection entry whose title is *“WAN Traffic of Branch”* to *“5 minutes”*.

Updating each property and the collection status(Status) of the data collection entry to the status of collecting(Running) by executing the API for updating data collection entry. Refer to ["4.3.4 Updating Data Collection Entry \(page 74\)"](#) for the details.

- Request:

```

PUT /umf/fw/nvp/datacol/entries/158 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 39
Date: Thu, 12 Nov 2015 04:33:00 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:9P3odttckTLf17iM459o8DQy9ifhRGsOoe3i54+YPgw=

{
  "Status": "Running",
  "Interval": "5m"
}

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 377
Date: Thu, 12 Nov 2015 04:33:03 GMT

{
  "Status": "Running",
  "Interval": "5m",
  "ReportType": "TrafficOfWAN",
  "RiseOrFall": null,
  "ContinuousTimes": null,
  "Format": null,
  "ReportDataOutput": "On",
  "RecoveryThreshold": null,
  "MIB": null,
  "AlertSeverity": null,
  "MibExprName": null,
  "Instances": "*",
  "AlertOutputCondition": null,
  "Threshold": null,
  "RecoveryAlert": null,
  "Nodes": "*"
}

```

```

"OperationMode": "Store",
"ID": 158,
"Name": "WAN Traffic of Branch"
}

```

5.3.3 Deleting Data Collection Entry

This subsection describes the procedure to delete the data collection entry that is no longer needed.

This operation is executed when the data collection entry becomes unnecessary due to the delete of the monitoring target node.

In this subsection, the procedure is described using the following operation example.

Example: Deleting the data collection entry whose title is “*LAN Traffic of Branch A*”.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Looking up ID of the data collection entry whose title is “*LAN Traffic of Branch A*”.

Obtaining the list information of the registered data collection entry by executing the API for obtaining data collection entry list. Refer to ["4.3.1 Obtaining Data Collection Entry List \(page 68\)"](#) for the details.

Looking up the data collection entry whose title(Name) is “*LAN Traffic of Branch A*” and obtaining its ID based on the obtained information.

- Request:

```

GET /umf/fw/nvp/datacol/entries HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 04:38:23 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:HgL6Djm78PQ0fm+YYBbEppieknWMULVx8DTcz40NAQ=

```

- Response:

```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 1161
Date: Thu, 12 Nov 2015 04:38:26 GMT

[
  {
    "Status": "Running",
    "Interval": "5m",
    "ReportType": "TrafficOfWAN",
    "RiseOrFall": null,
    "ContinuousTimes": null,

```



```

        "Format": null,
        "ReportDataOutput": "On",
        "RecoveryThreshold": null,
        "MIB": null,
        "AlertSeverity": null,
        "MibExprName": null,
        "Instances": "*",
        "AlertOutputCondition": null,
        "Threshold": null,
        "RecoveryAlert": null,
        "Nodes": "*",
        "OperationMode": "Store",
        "ID": 158,
        "Name": "WAN Traffic of Branch"
    },
    {
        "Status": "Running",
        "Interval": "10m",
        "ReportType": "TrafficOfHub",
        "RiseOrFall": null,
        "ContinuousTimes": null,
        "Format": null,
        "ReportDataOutput": "On",
        "RecoveryThreshold": null,
        "MIB": null,
        "AlertSeverity": null,
        "MibExprName": null,
        "Instances": "*",
        "AlertOutputCondition": null,
        "Threshold": null,
        "RecoveryAlert": null,
        "Nodes": "branchA_*",
        "OperationMode": "Store",
        "ID": 175,
        "Name": "LAN Traffic of Branch A"
    },
    {
        "Status": "Running",
        "Interval": "10m",
        "ReportType": "TrafficOfHub",
        "RiseOrFall": null,
        "ContinuousTimes": null,
        "Format": null,
        "ReportDataOutput": "On",
        "RecoveryThreshold": null,
        "MIB": null,
        "AlertSeverity": null,
        "MibExprName": null,
        "Instances": "*",
        "AlertOutputCondition": null,
        "Threshold": null,
        "RecoveryAlert": null,
        "Nodes": "branchB_*",
        "OperationMode": "Store",
        "ID": 177,
        "Name": "LAN Traffic of Branch B"
    }
]

```

2. Updating the collection status of the data collection entry whose title is “*LAN Traffic of Branch A*” to the status of not collecting.

Updating the collection status(Status) of the data collection entry to the status of not collecting(Stopped) by executing the API for updating data collection entry. Refer to ["4.3.4 Updating Data Collection Entry \(page 74\)"](#) for the details.

Tip

It is not allowed to delete the data collection entry when the collection status(Status) of the data collection entry is in the status of collecting(Running).

- Request:

```
PUT /umf/fw/nvp/datacol/entries/175 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Content-Length: 21
Date: Thu, 12 Nov 2015 04:38:48 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:dlf+3QULkj3Z97wESQWhqlYWxS8CdTJZS7VlFQa41zM=

{
  "Status": "Stopped"
}
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Pragma: no-cache
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Cache-Control: no-store
Content-Type: application/json; charset=UTF-8
Content-Length: 390
Date: Thu, 12 Nov 2015 04:38:51 GMT

{
  "Status": "Stopped",
  "Interval": "10m",
  "ReportType": "TrafficOfHub",
  "RiseOrFall": null,
  "ContinuousTimes": null,
  "Format": null,
  "ReportDataOutput": "On",
  "RecoveryThreshold": null,
  "MIB": null,
  "AlertSeverity": null,
  "MibExprName": null,
  "Instances": "*",
  "AlertOutputCondition": null,
  "Threshold": null,
  "RecoveryAlert": null,
  "Nodes": "branchA_*,
```

```
"OperationMode": "Store",
"ID": 175,
"Name": "LAN Traffic of Branch A"
}
```

3. Deleting the data collection entry whose title is “*LAN Traffic of Branch A*”.

Deleting the target data collection entry by executing the API for deleting data collection entry. Refer to ["4.3.5 Deleting Data Collection Entry \(page 79\)"](#) for the details.

- Request:

```
DELETE /umf/fw/nvp/datacol/entries/175 HTTP/1.1
Host: 192.168.10.121:12080
Accept-Encoding: identity
Date: Thu, 12 Nov 2015 04:39:16 GMT
X-NVP-API-VERSION: 1
Content-Type: application/json; charset=utf-8
Authorization: SharedKeyLite 38PEmnk7Jzd2zugAo1GydWyWcymFAulsD1jfp
pdVj8U=:xDtfr0YXLeEk+koGP3mtOrmJqxVA88ib+zPWsFZ0bQ=
```

- Response:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-NVP-API-VERSION: 1.0.0
Content-Length: 0
Date: Thu, 12 Nov 2015 04:39:18 GMT
```

5.4 Operating Alert

This subsection describes the way to operate alerts by using each API.

5.4.1 Confirming Alert of Target Node

This subsection describes the procedure to obtain and confirm the list of occurred alerts.

This operation is executed to confirm the SNMP trap that occurred on the target node in a specific time.

In this section, the procedure is described using the following operation example.

Example: Displaying and confirming the list of alerts occurred on node "Catalyst3750" during the period from 21th March 2017 to 23th March 2017.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Obtaining the list information of alerts meeting the specified conditions.
 - a. Executing the API for obtaining alert list.
 - Request:

```
GET /umf/fw/nvp/alerts?ComponentName=Catalyst3750&
SinceOccurTime=2017-03-21T00:00:00+09:00&
UntilOccurTime=2017-03-23T00:00:00+09:00&
Type=SNMPTrap&RecoveryStatus=NeedRecover&Severity=Warning&
Fields=Alerts/ID,Alerts/Type,Alerts/SummaryMessage
```

```
Accept: application/json
Accept-Charset: utf-8
Connection: Keep-Alive
Host: server.aaa.bbb.ccc
```

- Response:

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 1000
Connection: Keep-Alive
{
  "ResultNextID" : 1200,
  "ResultNextTime" : "2017-03-21T13:11:00+09:00",
  "Alerts" : [
    {
      "ID" : 100,
      "Type" : "SNMPTrap",
      "SummaryMessage" : "Interface 11 is down."
    },
    {
      "ID" : 101,
      "Type" : "SNMPTrap",
      "SummaryMessage" : "Interface 11 is down."
    },
    (Snip: The same number of alert information as MaxCount are displayed.)
  ]
}
```

Non-displayed alerts exist when some value other than “0” is set in "ResultNextID".

- Executing the API for obtaining alert list again.

Setting values in "IPAddress", "NextID"(value of "ResultNextID") and "SinceOccurTime"(value of "ResultNextTime").

Repeating this process until the value of "ResultNextID" becomes “0”.

- Request:

```
GET /umf/fw/nvp/alerts?ComponentName=Catalyst3750&
SinceOccurTime=2017-03-21T00:00:00+09:00&
UntilOccurTime=2017-03-23T00:00:00+09:00&
Type=SNMPTrap&RecoveryStatus=NeedRecover&Severity=Warning&
Fields=Alerts/ID,Alerts/Type,Alerts/SummaryMessage&
NextID=1200&SinceOccurTime=2017-03-21T13:11:00+09:00
Accept: application/json
Accept-Charset: utf-8
Connection: Keep-Alive
Host: server.aaa.bbb.ccc
```

- Response:

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 1000
Connection: Keep-Alive
```

```

{
  "ResultNextID" : 0,
  "ResultNextTime" : "",
  "Alerts" : [
    (Snip: The same number of alert information as MaxCou
nt are displayed.)
  ]
}

```

"0" is set in "ResultNextID" and an empty character is set in "ResultNextTime" when all alerts are already obtained.

2. Confirming the details of the alert with the target ID.

- Request:

```

GET /umf/fw/nvp/alerts/100 HTTP/1.1
Accept: application/json
Accept-Charset: utf-8
Connection: Keep-Alive
Host: server.aaa.bbb.ccc

```

- Response:

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 1000
Connection: Keep-Alive
{
  "ID" : 100,
  "Type" : "SNMPTrap",
  "Sender" : "SNMP",
  "Severity" : "Normal",
  "RecoveryStatus" : "NeedRecover",
  "ConfirmStatus" : "Unconfirmed",
  "ComponentType" : "Node",
  "Component" : "Catalyst3750",
  "IPAddress" : "10.21.1.3",
  "OccurredTime" : "2017-03-20T10:00:00+09:00",
  "RecoveryTime" : "",
  "SummaryMessage" : "Interface is down.",
  "SummaryMessage" : "Interface 11 is down.",
  "ActionMessage" : null,
  "SNMPVersion" : "1/2c",
  "SNMPCommunity" : "public",
  "AgentAddress" : "10.21.1.3",
  "EnterpriseOID" : "1.3.6.1.6.3.1.1.5.1",
  "GenericCode" : 0,
  "SpecificCode" : 0,
  "TimeStamp" : 21001234567
}

```

3. Updating the confirmation status of the alert with the target ID to "Confirmed".

- Request:

```

PUT /umf/fw/nvp/alerts/100 HTTP/1.1
Content-Type: application/json
Content-Length: 1000
Accept: application/json
Accept-Charset: utf-8
Connection: Keep-Alive
Host: server.aaa.bbb.ccc
{
    "ConfirmStatus" : "Confirmed"
}

```

- Response:

```

HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 1000
Connection: Keep-Alive
{
    "ID" : 100,
    "Type" : "SNMPTrap",
    "Sender" : "SNMP",
    "Severity" : "Normal",
    "RecoveryStatus" : "NeedRecover",
    "ConfirmStatus" : "Confirmed",
    "ComponentType" : "Node",
    "Component" : "Catalyst3750",
    "IPAddress" : "10.21.1.3",
    "OccurredTime" : "2017-03-20T10:00:00+09:00",
    "RecoveryTime" : "",
    "SummaryMessage" : "Interface is down.",
    "SummaryMessage" : "Interface 11 is down.",
    "ActionMessage" : null,
    "SNMPVersion" : "1/2c",
    "SNMPCommunity" : "public",
    "AgentAddress" : "10.21.1.3",
    "EnterpriseOID" : "1.3.6.1.6.3.1.1.5.1",
    "GenericCode" : 0,
    "SpecificCode" : 0,
    "TimeStamp" : 21001234567
}

```

5.4.2 Recovering Target Alert

This subsection describes the procedure to update the fault status of the alert to the status of "Recovered".

This operation is executed to recover the alert that is not automatically recovered (i.e. the alert without state transition) such as SNMP trap and syslog.

In this section, the procedure is described using the following operation example.

Example: Recovering the "Confirmed" alert that occurred on node "Catalyst3750" at 13:00 on 21th March 2017.

Tip

The body response part is displayed with adding line feeds and indents for improving readability.

1. Obtaining the list information of the alert meeting with the specified conditions.

Executing the API for obtaining alert list with specifying "IPAddress".

- Request:

```
GET /umf/fw/nvp/alerts?ComponentName=Catalyst3750&
    SinceOccurTime=2017-03-21T13:00:00+09:00&
    UntilOccurTime=2017-03-23T13:00:59+09:00&
    Type=SNMPTrap&Severity=Warning&
    RecoveryStatus=NeedRecover&ConfirmStatus=Confirmed&
    Fields=Alerts/ID,Alerts/Type,Alerts/RecoveryStatus
    ,Alerts/ConfirmStatus,Alerts/SummaryMessage
Accept: application/json
Accept-Charset: utf-8
Connection: Keep-Alive
Host: server.aaa.bbb.ccc
```

- Response:

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 1000
Connection: Keep-Alive

{
  "ResultNextID" : 0,
  "ResultNextTime" : "",
  "Alerts" : [
    {
      "ID" : 100,
      "Type" : "SNMPTrap",
      "RecoveryStatus" : "NeedRecover",
      "ConfirmStatus" : "Confirmed",
      "SummaryMessage" : "Interface 11 is down."
    },
    {
      "ID" : 101,
      "Type" : "SNMPTrap",
      "RecoveryStatus" : "NeedRecover",
      "ConfirmStatus" : "Confirmed",
      "SummaryMessage" : "Interface 11 is down."
    }
  ],
  (Snip: The same number of alert information as MaxCount are displayed.)
]
```

2. Updating the status of the alert with the specified ID to the status of "Recovered".

- Request:

```
POST /umf/fw/nvp/alerts/batch-update?ID=100,101,105,500 HTTP/1.1
Content-Type: application/json
Content-Length: 1000
Accept: application/json
Accept-Charset: utf-8
```

```
Connection: Keep-Alive
Host: server.aaa.bbb.ccc
{
  "RecoveryStatus" : "Recovered"
}
```

- Response:

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 1000
Connection: Keep-Alive

{
  "ResultNextID" : 0,
  "ResultNextTime" : ""
}
```


Appendix A. Standard Specification Format

A.1 Standard Matching Specification Format

When specifying a component-type name or component name, you can use the "standard matching specification format".

In matching, the asterisk (*), question mark (?), left bracket ([), right bracket (]), hyphen (-) and backslash (or yen symbol) (\) have meanings. These characters with meanings are referred to as metacharacters.

1. The asterisk (*) matches any character string of zero or more characters.
 - Example: Switch*
This example matches any name starting with "Switch", such as "Switch1" or "SwitchABC".
 - Example : *
This example matches all registered names.
2. The question mark (?) matches any single character.
 - Example: Router?
This example matches any five-letter name starting with "Router", such as "Router1" or "RouterA".
3. Brackets match any characters contained within the brackets.
 - Example: Host[123]
This example matches "Host1", "Host2" or "Host3".

When there are two characters connected by a hyphen (-) within the brackets, any characters within that character range are matched. These characters can be specified using alphabetic characters. Multiple specifications are also possible.

- Example: Host[1-3]
This example matches "Host1", "Host2" and "Host3".
- Example: Host[a-zA-Z0-9]
This example matches any name that starts with "Host" and ends with one alphanumeric character, such as "SwitchA" or "Switch1".

A.2 Standard Component Name Specification Format

When specifying a component, you can use the standard component name specification format. The standard component name specification format is a format with combination of the component-type name and component name, and is followed by the rules below.

1. The component-type name is delimited from the component name by (:).

- Example: node:Switch1

Component type name has the following types.

- node
Device icon name registered in the Map View
- grp
Group name specified in the **Group** column of the Properties dialog box
- map
Map icon name registered in the Map View

You can use standard matching specification format expressions such as the asterisk (*) to specify component-type names or component names.

- Example: node:Switch*
This example indicates node components, all of which names begin with "Switch".
 - Example: grp:Group*
This example indicates group components, all of which names begin with "Group".
2. Multiple components can be specified by connecting component names with a comma (.).
 - Example: node:Node1,node:Node2,grp:Group*,node:Node3
This example indicates node components of which names are "Node1", "Node2", and "Node3", and a group component of which name begins with "Group".
 3. Component type can be omitted. When the specified beginning of the component type is omitted, it is considered as a node component type. When a component type connected by comma (,) is omitted, it is considered as a type equal to the previous component.
 - Example: Node1,Node2
This example indicates node component of which name is "Node1" and "Node2".
 - Example: grp:Group1,Group2
This example indicates a group component of which name is "Group1" and "Group2"
 - Example : *
This example indicates all node components.
 - Example: Node1,grp:Group*,node:HostNode*
This example indicates node component of which name is "Node1", node component of which name begins with "HostNode", and all group components of which names begin with "Group".

In this example, "node" before "HostNode*" can not be omitted. If it is omitted, the result is the same as when "grp:HostNode*" is specified, then it indicates not node but group component.
 4. A colon (:) or comma (,) must not be preceded or followed with a space.
 - Incorrect: Host32, Host33, Host34

Appendix B. Specifying wildcards

This section describes the wildcards used in the Filter Settings dialog bar.

- The asterisk (*) matches any character string of zero or more characters.

The following example matches with all names that start with "nvpc", such as "nvpc32" and "nvpcx".

Example: nvpc*

- The question mark (?) matches with a given single character.

The following example matches with all five-character names that start with "nvpc", such as "nvpc3" and "nvpcx".

Example: nvpc?

- Multi-byte characters can be matched.

**MasterScope Network Manager WebAPI 1.1
Reference Manual**

NVP00AE0111-03

November, 2018 3 Edition

NEC Corporation

© NEC Corporation 2015 - 2018