

A SELF-CONFIGURABLE SERVER FOR CONTROLLING DEVICES OVER THE SIMPLE NETWORK MANAGEMENT PROTOCOL

V. Petrosyan, V. Rybnikov, DESY, Hamburg, Germany

Abstract

The Simple Network Management Protocol (SNMP) is an open-source protocol that allows manufacturers to utilize it for controlling and monitoring their hardware. More and more SNMP-manageable devices show up on the market that can be used by control systems for accelerators. SNMP devices are also being used at the free-electron laser (FLASH [1]) and planned to be used at the European X-ray Free Electron Laser (XFEL[2]) at DESY, Hamburg, Germany. To provide an easy and uniform way of controlling SNMP devices a server program has been developed. The server configuration, with respect to device parameters to control, is done during its start-up and driven by the manufacturer Management Information Base (MIB) files provided with SNMP-enabled hardware. The list of controlled parameters can also be tailored by user's configuration files.

The SNMP server integrates the various type devices into the Distributed Object-Oriented Control System (DOOCS [3]) used at FLASH and XFEL. Several SNMP devices of different types can be controlled by one SNMP server simultaneously.

SNMP AT A GLANCE

SNMP is an application-layer protocol defined by the Internet Architecture Board (IAB) in RFC1157 (May of 1990) [4] for exchanging management information between network devices. It is a part of Transmission Control Protocol/Internet Protocol (TCP/IP [5]) suite. SNMP uses User Datagram Protocol (UDP) [6] for the transport layer. SNMP requests are transmitted as UDP datagrams over a connectionless transmission interface between SNMP manager and SNMP agent. Management Information Base (MIB) is a database containing information about elements to be managed. It represents Manageable Objects (MO) as resources. Every MO (device attribute) has a unique Object Identifier (OID) that can be used by queries. It also provides a map between numeric OIDs and a human-readable text.

SNMP agent is a daemon process running on a network device. It exposes device attributes to managers. SNMP managers control the attributes via Get/Set requests. Managers can have access to those attributes that are allowed by the agent. SNMP agents can also provide traps (asynchronous messages). Traps are mainly used for significant events in the device (e.g. power-off).

SNMP SERVER IMPLEMENTATION

The SNMP server is implemented as a DOOCS server.

It allows to integrate SNMP devices into FLASH and XFEL control systems in a transparent way. The DOOCS SNMP server as an SNMP manager provides access to attributes of the controlled device by means of DOOCS properties.

Server Design

The server uses the NET-SNMP library [7] that provides all functionality required to work with SNMP agents. The library allows:

- to retrieve the list of attributes controlled by an SNMP agent,
- read and write attributes values.

The list of attributes can be acquired starting from a certain node of the MIB tree. This node is defined by one DOOCS property of the SNMP server. Once the full list of attributes is available for the SNMP server it filters out the attributes of interest and creates for every selected SNMP attribute a set of DOOCS main (see Table 1) and optional (see Table 2) properties. Every property name in the set starts with the SNMP attribute name with upper case letters. The filtering of attributes is implemented via white or black lists provided by users.

Reading/writing an SNMP attribute value is implemented via a dedicated DOOCS property. There are two ways to read the attribute value:

- periodically via the server internal update timer,
- immediately on access to the DOOCS property.

Writing the attribute value is done on the corresponding DOOCS property (NAME) change.

Table 1: Main DOOCS Properties for SNMP Attribute

Name	Purpose
NAME.MIB	MIB attribute name
NAME.DESC	Attribute XML description
NAME.ACCESS	Access mode (e.g. read-only)
NAME.TYPE	Data type (e.g. ASN_INTEGER)
NAME	Value of the attribute
NAME.SL	List of valid values to write

According to “Structure of Management Information Version2” (SMIv2, RFC2578 [8]) the data types used by SNMP agents don't include floating point or double type.

ISBN 978-3-95450-148-9

Instead, scaled integers are used to show non-integer values (e.g. temperature 25.7 °C can be expressed as 2570). For such SNMP attributes a DOOCS polynominal-parameter type can be used for converting the integer value to the corresponding float value. In addition, an archiving history can be provided for SNMP DOOCS property values. This allows to see the value changes over a long time period.

Table 2: Optional DOOCS Properties for SNMP Attributes

Name	Purpose
NAME.HIST	Value history
NAME.FILT	Filter parameters for history
NAME.FLOAT	Attribute value as float
NAME.POLYPARA	Conversion parameters to float ($a_0+a_1*V+ a_2*V^2$)
NAME.ALARM	Alarm flag
NAME.LIMIT	Value for alarm limit
NAME.ALARM.OP	Operation for comparison
NAME.ALARM.OP.SL	List of available operations
NAME.ALARM.MASK	Flags to work with alarm

The DOOCS SNMP server provides an alarm mechanism to ease the device monitoring by operators. Every SNMP attribute can be configured for an alarm detection. An alarm will be triggered if a comparison of the current value of the attribute and some user predefined limit against one of logical operations gives a true value. The attribute alarm will propagate up to the special SNMP DOOCS server 'ALARM' location. This location sums up all incoming alarms and exposes the global server alarm via the value of its ALARM property.

IN OPERATION

On start-up the SNMP server reads out its DOOCS configuration file. The file contains all information required for the communication with an SNMP agent (see Table 3)

If during the start-up the SNMP agent is not reachable, the server will check the availability of the agent with some predefined time period. Once the agent is found, the sets of DOOCS properties for the required SNMP attributes will be created. After that the server monitors the attribute values either only periodically with its predefined period or additionally on a DOOCS request to the SNMP server from a DOOCS client (e.g. user panel).

One SNMP server can work with several SNMP devices of different types simultaneously. One DOOCS SNMP location is dedicated to a group of attributes within

one SNMP agent. It means that several SNMP locations of one SNMP server can control several groups of attributes from one SNMP agent as well as from several independent SNMP agents. Due to the multi-threaded [9] implementation of the SNMP server the interactions with SNMP agents within the same server are completely independent.

Table 3: DOOCS Properties for SNMP Communication

Name	Purpose
SNMP.HOSTNAME	SNMP agent host
SNMP.VERSION	SNMP version
READ_COMMUNITY	SNMP read community name
WRITE_COMMUNITY	SNMP write community name
SNMP.TIMEOUT	SNMP maximum response delay time
MIBDIR	Directory with MIB files
MIBFILE	First MIB file name to use
NODE	SNMP object name to start the attributes search

Adding a new SNMP device into the SNMP server doesn't require a server re-start. A new SNMP DOOCS location is dynamically created by a DOOCS request to the server. To complete the new SNMP device configuration the user has only to set the DOOCS properties from Table 3 with the help of a panel (see Fig. 1). Once all properties are filled with values the SNMP server tries to reach the device.

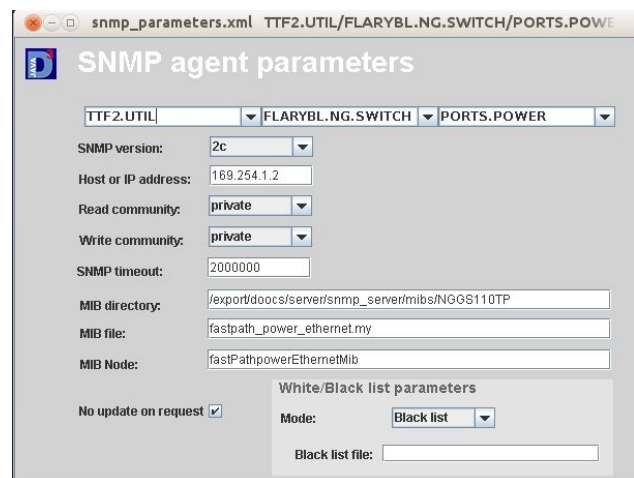


Figure 1: Panel for configuration of an SNMP agent

White/Black Attribute Lists

Usually an SNMP object tree contains many objects and attributes. To select only those that a user is interested in, a mechanism of white/black list is used. A dedicated DOOCS property defines whether the white or black list mode will be applied. A positive value means

the white list mode, negative – black list mode, zero – none of the lists will be used. The last case can be used for browsing the full SNMP tree of the device.

In case of the white list mode, the SNMP server creates DOOCS properties only for those SNMP attributes which names match the patterns from the white list. In case of the black list mode, the SNMP attributes with names, that match the patterns from the list, will be ignored.

Alarm Configuration

The white lists can also be used for the alarm configuration. In this case the line in the list contains three parameters:

- attribute pattern,
- operation for alarm detection (<, <=, =, !=, >=, >),
- alarm limit.

A few optional DOOCS properties are created for attributes with alarms (see Table 2). They provide full control over the attribute alarm configuration:

- changing operation for comparison,
- changing the limit value,
- masking or resetting the alarm.

The last operation allows to exclude malfunctioning devices from the ALARM monitoring without removing the device from the SNMP server.

Server Alarm

A dedicated DOOCS location is used for the alarm indication. Its ALARM DOOCS property is set to non zero value if there is at least one SNMP attribute with the non-masked alarm.

This tree-like propagation schema allows to build simple GUIs for monitoring alarms in many SNMP devices controlled by one SNMP server.

SERVER APPLICATIONS

NETGEAR GS110TP

Many network switches like the NETGEAR GS110TP [10] are exploited at DESY. They are used for integrating network cameras into different diagnostics systems. The switch (see Fig. 2) offers eight 1 Gigabit ports with Power-Over-Ethernet (PoE) [11] capability.



Figure 2: Front panel of GS110TP switch

The switch has an SNMP agent that provides a rich set of switch attributes. The most important switch attributes with respect to camera operations have been chosen for the SNMP server. They are:

- PoE enable,
- PoE status,
- Power Consumption,
- Temperature etc.

An easy access to PoE-enabled attributes of the switches via the SNMP server allows to perform a remote power reset for the connected cameras.

Raritan's Smart Rack Controller

Raritan's environment rack controller [12] is an IP-based appliance that allows to use any of Raritan's sensors such as temperature, humidity, air pressure and airflow. EMX2-111 (see Fig. 3) is used for performing the climate control of the Schroff racks (VARISTAR LHX20/40 [13]) at the FLASH and XFEL facilities.

All monitored attributes are accessible via an SNMP agent. The device parameters are split into two groups. Every group is controlled by one location of the DOOCS SNMP server. The first location is responsible for the rack climate parameters (humidity and temperature sensors of the rack). The second one gets the information from the cooling equipment (air and water temperature, speed of fans). DOOCS archiving histories are provided for all attributes.



Figure 3: Raritan's smart rack controller

Rittal's LCP-Smart and CMC Rack Climate Control

Rittal's Liquid Cooling Package (LCP smart) [14] is a rack system that serves to dissipate high heat losses and for effective cooling of devices built into the enclosure. The package is equipped with a control unit (Basic CMC). Its task is to monitor all connected sensors and to control the cooling devices (fans, etc.) in order to reach the temperature set points. CMC can be controlled remotely via the embedded SNMP agent.

Along with LCP smart, CMC rack climate control devices (see Fig. 4) are being used at FLASH and XFEL. They are:

- CMC-TC processing unit [15],
- CMC-TC climate unit [16],
- CMC-TC I/O unit [17].

Equipped with the required sensors and cooling devices they perform the climate control in a rack. CMC-TC processing unit has an SNMP agent. Only one DOOCS location is used for managing one Rittal's control unit attributes available for the SNMP server.



Figure 4: CMC-TC processing , I/O and climate units

STATUS

The SNMP server design allows to deploy the server for controlling any device running an SNMP agent. One SNMP server can operate several SNMP agents of different devices simultaneously.

The white/black list approach for SNMP attributes selection is proven to be very powerful to reduce the total number of properties to operate on. It allows the operator to focus only on the parameters required by users.

Having a set of DOOCS properties per one SNMP attribute provides a flexible way of controlling and monitoring of the SNMP attribute. It also allows to build the smart operator GUIs (e.g. based on jDDD [18]) that can provide all detailed information about the SNMP attribute (e.g. type, description, access, allowed values to write etc.).

The-tree like ALARM propagation schema offers a very simple way of monitoring alarms in all SNMP devices controlled by one SNMP server.

Twelve instances of SNMP servers are currently used in different diagnostic systems exploiting network cameras with the PoE capability. The DOOCS camera software [19] makes use of the SNMP server to perform automatic camera recovery in case of the camera disconnection is detected.

FLASH and XFEL exploit one SNMP server each for controlling currently used racks. Every server controls all available racks within the facility. At FLASH: Schroff – 3, CMC – 9, at XFEL: Schroff – 3, CMC – 5. The final

number of CMC racks at XFEL is about 190 pieces. It is not yet decided how many SNMP servers will be used to control the parameters of all XFEL racks.

REFERENCES

- [1] <http://flash.desy.de>
- [2] <http://xfel.desy.de>
- [3] K. Rehlich, “Status of the TTF VUV-FEL Control System”, PCaPA2005 proceedings, Hayama, Japan; <http://conference.kek.jp/pcapac2005/paperindex.html>
- [4] <https://datatracker.ietf.org/doc/rfc1157/>
- [5] W.Richard Stevens, “UNIX Network Programming”, (Prentice Hall, 1990), 199.
- [6] W.Richard Stevens, “UNIX Network Programming”, (Prentice Hall, 1990), 200.
- [7] <http://www.net-snmp.org/>
- [8] <https://datatracker.ietf.org/doc/rfc2578/>
- [9] Steve Kleiman, Devang Shah and Bart Smaalders, “Programming with Threads”,(Prentice Hall, 1996)
- [10] <http://support.netgear.com/product/GS110TP>
- [11] IEEE 802.3af, IEEE 802.3at
- [12] <http://www.raritan.com/products/environmental-monitoring/emx-smart-rack-controller>
- [13] http://www.varistar.de/src/varistar_lhx20_d.pdf
- [14] <http://www.rittal.com/>
- [15] http://www.rittal.com/imf/none/3_407/
- [16] http://www.rittal.com/imf/none/3_413/
- [17] http://www.rittal.com/imf/none/3_409/
- [18] G. Grygiel, V. Rybnikov, “DOOCS Camera Server”, ICALEPCS07 proceedings, Knoxville, Tennessee, USA, October 2007, p.359 (2007)
- [19] E. Sombrowski et al., “jddd: A Tool for Operators and Experts to Design Control System Panels”, ICALEPCS2013 proceedings, San Francisco, CA, USA, October 2013, p.544 (2013)