A Study on Designing and Realizing MQSPT-based Open API SMS Architecture

Yong-Hee Lee¹, Jung-Ah Shim², Hyun-Jung Kwon³ and Keun-Heui Kim⁴*

¹,²,³,⁴ Department of IT Policy and Management, Soongsil University, Seoul 156-743, Republic of Korea
¹ greenyon@naver.com, ² rosapoodle@naver.com, ³ musehjkwon@gamil.com, ⁴ interbp@hanmail.net

Abstract

The Enterprise Management System is the representative system supporting information about important decision making of a company’s information system management. SMS (System Management System), which manages the various system status in the information system, uses TCP/IP communication-based SNMP (Simple Network Management Protocol) to collect, process, and save different system information, and provide information for users to make decisions. However, SNMP cannot guarantee the data transmission reliability in regard to system circuit error, delay, and short that may occur during the information collection. Consequently, it involves the issue of inaccuracy of the SMS information. In addition, it is difficult to guarantee the data transmission reliability due to different system problems in the TCP/IP communication.

In this study, we applied the SPT (Safe Proper Time) technology to guarantee the reliability of TCP/IP communication, safely linked the intranet system information to the internet network, using the MQSPT-based Open API technology. We designed it to be able to manage the system in the intranet environment efficiently even on the internet network. This study suggests designing and constructing the MQSPT-based Open API SMS architecture.

Keywords: MQSPT, SPT, ITSM, EMS, SMS

1. Introduction

The Open SMS (System Management System), which enables unrestricted control of internet network-connected IT system, is categorized into three levels: IL (Instrumentation Level), AL (Agent Level), IML (Information Manager Level). The problem of the existing SMS is the low accuracy when collecting the information about each node’s status, while being based on the SNMP. As a matter of fact, data SNMP of the first IL level has low accuracy of collecting information due to the error or the circuit delay of the node. In order to improve the SNMP problem, applying the SPT-based TCP/IP communication module to the SMS, as proposed in this study, can solve various network issues. Moreover, this study proposes an architecture, in which the MQSPT Open API is linked with the intranet to enable efficient management in the internet network. This thesis is composed as follows: Chapter 2 draws the SMS Delivery View, system hierarchy level outline, and the Process View of the process composition for each level, analyzing the understandings and the issues in regard to the SNMP and describing the necessity of the improvement. Chapter 3 compares and analyzes the SNMP and the SPT-

* Corresponding Author
based SNMP to design and propose the MQSPT-based Open API module for SMS architecture. Chapter 4 compares the speed of the SNMP of the existing SMS to that of the SPT protocol, and concludes the quantitative performance result of the system through the pilot project. Lastly, chapter 5 concludes the study and describes about the further study.

2. Related Research

Transmission procedure of the SMS (System Management System) follows three levels. Starting from the node at the Instrumentation Level, it goes through the Agent Level to collect information and the IML Information Manager Level to sort, save, and process the message information for decision making, and finally, it reaches the SMS Agent that transmits messages through the user terminal. Figure 1 illustrates the SMS process.

![Figure 1. SMS Delivery Process](image)

Main research subjects are the IL and AL sections that use the SNMP. The information generated at IL is delivered through SNMP. Meanwhile, errors of various devices at the level interferes accurate information collection.

The SMS system plays the most important role among the EMS support system supporting the system administrator to make correct decisions. It must improve vulnerabilities of the SNMP and supporting reliability, availability, and integrity of data transmission. This study applies the SPT [1] module, which secures the reliability of network communication, to the SNMP and then, it designs an architecture of the Open SMS.

![Figure 2. System Management System Over View](image)
Analyzing the third level architecture of logic modeling of the studied SMS enables the identification of the network vulnerability in question. In the first level of information collection, the NMS uses the SNMP to send "Get Request" to each node (Server, Router, etc.). Then, as a reply, each node transmits “Get Response” is transmitted. The second level, the Agent level, sorts the received information, sending it to the IML. The transmitted information is re-processed to provide information through Console, Web service, and App service interface. The administrator evaluates the statuses of the device and the system, considering the provided information, and comes into any necessary action.

SMS Process View provides a reply to the first level, the IL, while each device provides replies for the requests to the SNMP. The second level (AL) classifies information from each device, according to the message classification code, and converts it in the proper form for the IL. The classification criteria mainly fall into four categories; (1) collecting the account and the access log as the software information of the operation system, (2) collecting different information of CPU, Memory, Disk, Network Device, etc. as the hardware information, (3) collecting the process information of the Kernel and application layers of the operating system as the system information, and (4) processing instructions to control the IL. In the IML, it collects and processes bulk information for the system administrator to make decisions.

$$\text{IL (Instrumentation Level)}$$

$$\text{AL (Agent Level)}$$

$$\text{IML (Information Manager Level)}$$

$$\text{Management}$$

$$\text{Console}$$

$$\text{SMS Web}$$

$$\text{SMS App}$$

$$\text{Sender-Receiver}$$

$$\text{Request Process}$$

$$\text{SMS Engine}$$

$$\text{Mdc Agent}$$

$$\text{Software}$$

$$\text{Hardware}$$

$$\text{System Information}$$

$$\text{Response Process}$$

$$\text{Data Process Engine}$$

$$\text{Sending Processing}$$

$$\text{Listen}$$

$$\text{Convert}$$

$$\text{Control}$$

$$\text{Sender-Receiver}$$

$$\text{SNMP: Get Request}$$

$$\text{SNMP: Get Response}$$

$$\text{Request/Report}$$

$$\text{Approval/Order/Denial}$$

$$\text{Device}$$

$$\text{Router}$$

$$\text{Server}$$

$$\text{Access Point}$$

**Figure 3. System Management System Process View**

### 3. Open Architecture Proposal & MQSPT Mechanism

The Open API SMS architecture provides the API that enables collecting, processing, controlling, and searching the information about every system in the world that is connected to the internet network. It also registers systems to be managed, based on the proposed architecture, and collects, analyzes, and processed different information.

Existing intranet-based SMS has security issues, which cause difficulties in managing and controlling the system from the external internet network. In order to solve the problem of controlling the system in the closed network, reliability and stability of bulk information must be secured. Therefore, the MQSPT architecture was imported into the
Open API SMS to enable collecting and controlling various system information in the internet network. The following figure shows the MQSPT architecture.

![MQSPT Architecture Diagram]

**Figure 4. System Management System Process View**

The above figure shows the MQSPT communication middleware. As the Safe Proper Time module for data transmission, it transmits data through the module. It secures reliability and confidentiality of data transmission by handling encrypted data transmission (AES, ARIA, SEED) and data compression all at once. Moreover, when there are multiple access nodes, it manages sessions of each node stably. Transmission delay and disconnection due to abnormal behavior of the node are under control, as well. As Figure 5 demonstrates, the composition of the architecture secures the flexibility. It uses the MQSPT module, which guarantees reliability and confidentiality of data transmission, being able to alter the 3-Tear environment into any structure from sending to receiving, as it wants. One of its important features is that there is no data loss, which prevents potential issues in transmitting bulk data.

As the above figure shows, the SPT was introduced as a module to deal with message information generated from the internal system to be managed at the IL (Instrumentation Level). As the Open API SMS system as architecture, and the MQSPT architecture was introduced as sub-architecture. The MQSPT architecture is a communication architecture that secures stability and reliability of the message. The MQSPT Broker has AES, ARIA, and SEED algorithms for data transmission confidentiality. In addition, for rapid data transmission, it compresses data before sending it and releases the compression after receiving it, reducing the network traffic.
Once the data transmitted through the MQSPT module are on the internet circuit, with its confidentiality and stability secured, the SMS at the AL sorts and writes the data according to its type. Then, necessary information is collected, processed, saved, or searched on the dashboard. One of the most important features of the architecture is that all different users may collect, process, and control necessary information in the system that registered the SMS in the internet network. In addition, it imported the MQSPT architecture, meeting reliability and speed, to process the bulk message. Vulnerabilities of the existing SNMP were complemented by the SPT technology, enabling accurate control over the status information about the system to be monitored.

Figure 6 shows the Open API SMS sequence diagram, in which the MQSPT and SMS sequence diagram are combined. As a main feature of the MQSPT domain, the sender and the receiver access the transmission broker, carry out loop checking the circuit for data transmission, and secure reliability of the communication circuit. In addition, its
composition enables selective message transmission. When the SMS domain receives the message that the MQSPT domain requests through the NMS, it requests (Get Request) the message to the SNMP Agent. Then, the SNMP Agent collects the information and responds (Get Response). When there is any information, messages are sent repeatedly. The NMS sends the response to the MQSPT domain to complete the ultimate message delivery. The communication module for delivering the message the module, which overcomes delay in data transmission and circuit error.

4. Architecture Test and Comparison Analysis

4.1. Composition of Test Environment

Environment of development for the architecture test which is suggested in this study is as shown in the following Table 1. Using the open source software Linux kernel version 2.6.18-1.2798 as the operating system and using Xeon™ CPU 2.50 GHz and 2G byte memory as the hardware components minimize possible side-effects by operating the minimum process to raise reliability of test results.

<table>
<thead>
<tr>
<th>Table 1. Hardware Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware Main</strong></td>
</tr>
<tr>
<td>CPU</td>
</tr>
<tr>
<td>Memory</td>
</tr>
<tr>
<td>Network</td>
</tr>
</tbody>
</table>

The most important information about the development software is the limit facts about resources, and it becomes the standard value of usable resources. There are the usable resources of development software in Table 2. The most important information is the maximum number of open files and Message Queue byte.

<table>
<thead>
<tr>
<th>Table 2. Software Resource Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware Main</strong></td>
</tr>
<tr>
<td>Compiler</td>
</tr>
<tr>
<td>Thread</td>
</tr>
<tr>
<td>Kernel</td>
</tr>
<tr>
<td>Open Files(1024)</td>
</tr>
<tr>
<td>Message Queue(819200 byte)</td>
</tr>
</tbody>
</table>

4.2. Test Results

When the MQSPT transmission method is used, the status of the data transmission circuit can be inspected. When a circuit error is in the case, it detects the error immediately, records it in the message queue, which is a temporary buffer, to prevent data
loss. When the error is restored, the status result and data are transmitted immediately, securing reliability of data transmission. MQSPT guarantees the maximum mean transmission speed of 20 (us) in the case of 1000 byte data transmission.

![Comparison of Speed between MQSPT and NMS](image)

**Figure 7. Timeliness of MQSPT Data Processing**

The results of the study on the timeliness of data processing are shown above.

Timeliness that guarantees the reliability of data processing means sending data within the specified time. The above figure visualizes the timeliness of data processing for NMS and MQSPT communication. As shown here, MQSPT processes all data within 20 (us). In this figure, you can see that the mean processing speed of NMS is longer than 100 (micro seconds) and reaches up to 300 (micro seconds). However, in the case of MQSPT, the mean transmission speed does not exceed 20 (micro seconds). When expressed in ratio, the timeliness of MQSPT is 73.00 %, which is high. Therefore, MQSPT guarantees the timeliness of data processing.

![Successfull Data Transmission Ratio](image)

**Figure 8. Comparison between MQSPT and SNMP**

The data transmission reliability means sending given data within the specified time without loss. The above figure visualizes the data transmission reliability of NMS and MQSPT communication. As shown in this figure, MQSPT guarantees 100% reliability.

However, the NMS communication generates data loss due to session disconnection resulting from delayed data transmission. In this study, data transmission was tested for 100,000 times. The existing SNMP had 99.21% of the success rate of data processing, while that of the MQSPT was 100%.
5. Conclusion
This study analyzed the commonly used SMS system through the reverse engineering, concluded design architecture. To solve the issues found in the architecture, we combined the MQSPT communication architecture to finally design the MQSPT-based Open API SMS. It became a foundation to construct the Open API SMS system to take efficient control of a system to be managed, collecting, processing, saving and searching designated system information with the internet connection. For the data processing test, test unit was increased in terms of 10,000, from 10,000 to 100,000. The standard data packet size was 1,000 Bytes. Reliability of the communication based on the MQSPT guaranteed 100% of success, while the success rate of the SNMP was 99.21%. Future studies will be carried on to compute the maximum data throughput by the network bandwidth due to the access node increase of global Open API SMS for quantitative analysis of the maximum accessible node, and in turn, to compute the size of the Open API SMS system offer.

References

Authors
Yong-Hee Lee, He received his bachelor’s degree of biotechnology in Korea University, Seoul (1998) and Master’s degree of information processing in Sogang University, Seoul (2012). Now he is studying at the Soongsil University majoring in IT policy&management. His research interests are in the areas of IT disaster&recovery, national competency standards.
**Jung-Ah Shim**, She received her MBA in Culture and Art Management from Hongik University in Korea (2013). She has worked as an expert in Communicology and Social Network Service Public-Relations field for 8 years. Now she is an administrator in the Ministry of the Interior.

**Hyun-Jung Kwon**, She received her bachelor's degree in Computer Science from Seoul Women’s University in Korea, (1999). She worked in the IT field and Cyber Security Center. Her research interests are in areas of Cyber Security, Personal Information Protection Policy and Technology.

**Keun-Heui Kim**, He received his Master degree in Software Engineering from Sogang University in Korea, (2013). He worked in the IT field as a System Engineer over 16 years. His research interests are in areas of Software Engineering, Open Source Software, Security and Low latency Network.