Research on Key Distribution Algorithm of a Modbus-Based Protection System for Industrial Infrastructure Networks

Jae-gu Song, Sungmo Jung, Seoksoo Kim*
Department of Multimedia
Hannam University
133, Ojeong-dong, Daedeok-gu, Daejeon 306-791
Korea
bhas9@paran.com, sungmoj@gmail.com, sskim0123@naver.com

Abstract: - Modbus is one of the major network communications protocols used by SCADA(Supervisory Control And Data Acquisition). Most SCADA networks require a high level of availability and measurement accuracy, which calls for authentication of received data and prevention of inadequate or inaccurate control operation. To that end, an accurate communication is needed between Master and Client as well as an effective algorithm for encryption/description. Therefore, this research draws requirements of a network A(authentication)-key management system and key distribution algorithm.

Key-Words: -Modbus, SCADA, Security, Infrastructure, Key Distribution Algorithms, Key Management

1 Introduction
More recently, security for industrial infrastructure is of growing interest. This is because of the current shift in terrorist attacks from physical assaults from cyber attacks targeting the major infrastructure of a society. Most industrial infrastructure involves electricity, water systems, nuclear power, and gas distribution networks, of which even a minor problem can result in tremendous impacts on a society.

There are at present many studies on protection of such social infrastructure. In particular, a greater possibility of utilizing the Internet by industrial networks calls for an effective measure to overcome security weakness among internal communications protocols[1, 2, 3]. Previous communications protocols of the infrastructure are closed, and communications are done based on reliance between physical devices. That is, the attempt to connect them with the Internet may present a significant vulnerability to threats existing in a TCP/IP environment. Therefore, the protective measure shall be focused on communications protocols used by current industrial infrastructure.

Of the communications protocols used by the industrial infrastructure, this study is focused on Modbus, and suggests an authentication key distribution algorithm of encryption for safe data transmission.

2 Related research
2.1 Modbus
Communications of Modus Serial is done by token passing, in which Master can have only one connection at a time and Client, 247 connections. The request mode of Master supports both Unicast and Broadcast. In the Unicast mode, Client replies a request of Master and 1~247 of addresses are available. In the Broadcast mode, Master sends a request to all Clients and 248~256 of addresses are available. The speed is 1200/2400/4800/19200bps, 56Kbps and 115.2Kbps while the maximum communications distance is about 1 km and termination, 150Ω(Ohms)/0.5W.

A general communication frame is composed of a header, data, and a trailer[4, 5].

In order to generate an authentication key, we need to understand characteristics of data, frame elements, and a function code.

Modbus has the following frame elements.

<table>
<thead>
<tr>
<th>Primary Tables</th>
<th>Object Type</th>
<th>Type of Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete Input</td>
<td>Single Bit</td>
<td>Read Only</td>
</tr>
<tr>
<td>Coils</td>
<td>Single Bit</td>
<td>Read Write</td>
</tr>
<tr>
<td>input Registers</td>
<td>16 bit word</td>
<td>Read Only</td>
</tr>
<tr>
<td>Holding Registers</td>
<td>16 bit word</td>
<td>Read Write</td>
</tr>
</tbody>
</table>

Table 1. Frame Elements of Modbus

In this study, data encryption keys are generated by understanding Modbus function codes. The following table shows Modbus function codes.
2.3 Key Management Methods

We need to understand key management methods for key distribution. Major key management methods are as follows.

2.3.1 Centralized Group Key Management

One of the most basic methods is that the central server shares a secret key with other members and delivers a group key to each member through encryption using the shared secret key[7].

Fig.1 Centralized Group Key Management Protocols

2.3.2 Logical Key Hierarchy

In view of the determinacy of the centralized group key management, logical key hierarchy (LKH) has been suggested with the following features [8].

• Each node maintains one key.
• The root node of a tree becomes a group key.
• Each terminal or user is connected with one node and should receive all keys of that node and its ancestor node.

Fig.2 Logical Key Hierarchy

3 Requirements for Authentication Key Management

This study has derived requirements for key management by analyzing KDC and GKMP, two of authentication key distribution methods.
As for KDC, one KDC exists and manages keys for members of all groups, which renders a simple structure. It is more effective when the groups are small but quite ineffective if the groups are large or the members frequently change. Such structure is not suitable for a sensor network with limited resources as in this study. Therefore, there should be a measure to deal with frequent changes of members by applying a single key management program.

As for GKMP, not depending on one KDC, each group has a group controller and receives the authority for key management from a security manager. It is advantageous that each group controller is in charge of only a small number of group members and bottleneck phenomenon is less expected compared with use of one KDC. However, GKMP is managed through a central control center also, which means key issuance and update process becomes complex due to frequent key generation. As a result, it requires direct authentication between groups or members.

4 Suggested A-Key Management Methods

In this study, a secret key is generated along with Master ID and RTU Client ID/Key in a reliable and safe way. The major key authentication method suggested in this study is as follows.

\[
\begin{align*}
\text{EKd} &= \text{EKd(RtuM\_id || RtuS\_id || Kr)} - 256\text{bit} \\
\text{EKd1} &= \text{H(EKd||01128)} - 128\text{bit} \\
\text{EKd2} &= \text{H(EKd||02128)} - 128\text{bit}
\end{align*}
\]

The method described above uses EKd1 to confirm if the same key has been generated through EKd1/EKd2, generated in a way similar to SKKE, and uses EKd2 as a secret key.

The steps depicted above use a reliable secret key such as SKKE protocols in order to create a new trustworthy relationship[9].

The unique conditions of the industrial infrastructure require a stable and effective key distribution system to guarantee adequate transmission speed. To that end, the SEED algorithm, approved by the international standards, is employed.

The following figure briefly describes the key distribution process:

- Confirms authentication/Interrupts communications in case of failure of authentication
- Generates and transmits a broadcast key
- Starts encryption communications

5 System Design

The key distribution system developed in this study is composed of S/W, controlling key distribution processes though physical connection between RTU Master and Client. The following figure shows the structure of key distribution system connection.

In order to apply this system to actual H/W, the dotted lines should be at least 4 times greater than the undotted lines for stable key distribution. Message formats include...
STX, T_Length, Destination, Source, Command, D_Length, Data, ChecKSum, and ETX as follows.

1. STX : 0x02
2. From a destination to CheckSum
3. Address of a destination
4. Address of a starting place
5. Sequence no. of a transmission device(number in the box of the following page, ex: 3-SID transmission)
6. Length of the original data (2byte)
7. Transmission of data (ASCII code) 0xAA based on nibble-unit conversion, transmission of 0x41 0x41
8. Xor value(2 byte) between Data 1 ~ Data N

This system has Modbus communication structure of Unicast(1:1) and Broadcast(1:n) and the key distribution mechanism has the following purpose.

1. To be effective in terms of storage space for encryption key management, arithmetic overhead, communications overhead, and so on (considering time between key generation and update)
2. To make possible secure network through key update

This system generates pseudo-random numbers through SHA256, optimized in 32bit CPU. In case the first RTU Client attempts to log on, a connection is made through Unicast and when Client attempts to log on later, a key is generated through Broadcast.

The next figure describes data encrypted through key distribution.

![Data Flow Diagram]

In order to prevent random access of unknown users who attempt to find out an encryption key, if a communication between Master and Client fails 5 times or more, the operation goes back to generation of pseudo-random numbers to restart distribution. If this process repeats 5 times or more, a communication failure message appears, interrupting key distribution and authentication process. In addition, an idle frame is transmitted per second after the connection between Master and Client has been completed and, if a response is not confirmed 5 times or more, reading and writing is interrupted.

The average time needed for encryption/description in this system, measured by GetTickCount system timer, is as follows.

<table>
<thead>
<tr>
<th>Plaintext</th>
<th>Ciphertext</th>
<th>Decryption</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.21ms</td>
<td>0.19ms</td>
<td>0.20ms</td>
</tr>
</tbody>
</table>

Table 3. Average time of plain text encryption/description

The average time needed for a plain text to be sent from RTU Master to Client is 0.21 ms and the total time including encryption/description is as follows.

0.18 + 0.20 + 0.21 = 0.56ms

Although there is about 37.5% overhead compared to 1:1 and 1:N communications, system capacities may cause variations. And even if each key is updated through Broadcast with fully connected RTU Client, about 150ms speed is necessary for one-time transmission, indicating that overhead of RTU Master is negligible.

6 Conclusion

This research analyzes characteristics of Modbus protocols and suggests a key distribution method to protect data during data transmission of the industrial infrastructure. With this method, data cannot be decrypted even if exposed, for the data are certified through a secret key, different from previous transmission of a plain text. Furthermore, although the speed is slightly slower than previous plain text transmission, it will not affect the entire system and the increase in transmission time is negligible. This will cause no significant problem when the method is applied to an actual system.

Also, to solve the problem of A-key update when there is a great number of Clients, Broadcast key distribution method is applied, maintaining competitive performance.

Follow-up research shall be done in order to apply this system not only to Modbus but also to a DNP3 environment, as well as developing various security measures to deal with vulnerabilities arising from connection with TCP/IP protocols.

* Corresponding Author : Seoksoo Kim
References: