Evaluation of Security Algorithms for the SCADA System based on IEC 61850

Dae-Yong Shin, Sugwon Hong, Il Hyung Lim, and Seung-Jae Lee

***Abstract*— The IEC 61850 standard not only defines a new structure for substation automation, but also can provide more enhanced communication functionalities. The standard defines several kinds of messages for data exchange between nodes in the substation. Among them, two messages, sample values and GOOSE, play critical roles for substation protection. Because of their importance in the secure substation operation, any compromising of these messages should be prohibited. Along with the requirement of secure transmission, these messages have very stringent performance requirement. So any security algorithm that can be applied for secure exchange of these messages should not violate the performance criteria. In this paper we consider the security protocol for guaranteeing the message authentication and integrity. And we evaluate the security protocol and examine whether the protocol can meet this criteria by experiment.**

***Index Terms*-- SCADA system, IEC 61850, security algorithm.**

I. INTRODUCTION

The Supervisory control and data acquisition (SCADA) system is widely used in the critical infrastructure utilities. Recently the security issues in the SCADA system are getting attention because of its catastrophic impact when the system security is breached [1].

In this paper we overview the communication architecture of the SCADA system and the IEC 61850 standard communication protocol between nodes in the substation [2]. Among the several types of messages defined in the IEC 61850, the sample value message and the General Object Oriented Substation Event (GOOSE) message are critical for secure operation of protection relays in the substation. These messages have stringent performance requirements since they have to be processed in real time.

We consider the message authentication and integrity as the security goal for secure substation operation, since false sample value messages or malicious GOOSE messages may cause disastrous result in the substation protection operation. The security objectives are to assure only authorized access within the system, and prevent spoofing and playback of captured data from non-trusted entities.

This work was supported by the ERC program of MOST/KOSEF (Next-generation Power Technology Center).

Dae-Yong Shin, Sugwon Hong, Il Hyung Lim, Seung-Jae Lee are with the Next-generation Power Technology Center, Department of Computer Software, and Department of Electrical Engineering, Myongji University, Yongin 449-728, Korea (e-mail:, dosuser@naver.com, swhong@mju.ac.kr, mschoi@mju.ac.kr, sojoo2jan@mju.ac.kr, sjlee@mju.ac.kr)

The paper proposes a security protocol to achieve these security goals. The security algorithms that we consider in this paper are the message authentication algorithms based on the message authentication code (MAC). We also consider the key distribution protocols which include the peer-to-peer communication mode. Next, we do some experiments to evaluate the security algorithms.

In the following section we explain the communication architecture of the IEC 61850-based substation. In section 3, we consider the efficient ways of adapting the current security algorithms to achieve the security goals. In section 4, we show some experiment results to evaluate the protocols.

II. IEC 61850-BASED SUBSTATION COMMUNICATION

Main components of the substation are intelligent electronic devices (IED), power equipments, and substation controller. The substation controller, located in a central site, monitors and supervises a large number of IEDs which are field devices located in physical environments. IEDs gather data from sensors or power equipments which measure current and voltage, and send data to the substation controller. The actuator as a part of IED controls the operation of power equipments by commands issued by other IEDs. The substation controllers have a hierarchical structure. A high-level master station can control several sub-master stations.

The data and command transfer takes place between the substation controller and IEDs, between IEDs, or between IED and power equipments. These transfers are carried over substation networks. The substation network is based on various communication channels and network technologies including Ethernet, serial links, wireless communication, and so on.

The communication between the substation devices is governed by the standard communication protocols. The most commonly used protocols are IEC 60870-5, DNP3 which is the derivative of IEC 60870-5, and Modbus [3]. Recently the International Electrotechnical Commission (IEC) is working on the new protocol, IEC 61850, which not only defines a new structure for substation automation, but also can provide more enhanced communication functionalities [2].

The IEC 61850 defines seven different types of messages which are exchanging between the substation nodes. The IEC 61850 assumes the IEEE 802.3 LAN as an underlying communication network. The communication stacks depending on the message types are shown in figure 1.

Among the messages, the message of sampled values is

intended to deliver samples of 960 Hz signal from measuring devices to IEDs. The General Object Oriented Substation Event (GOOSE) message is an urgent message which conveys protection information between IEDs and circuit breakers or other protection devices. Upon detecting an event, the IED multicasts GOOSE messages to notify other IEDs of the events, and cause the actuator to do protection action. The GOOSE message transmission has stringent performance requirement. No more than 3 ms is allowed to elapse from the time an event occurs to the time a message is transmitted. Collision is possible since the IEC 61850 is based on IEEE 802.3 network. So the GOOSE messages are retransmitted several times by IED.

The Manufacturing Message Specification (MMS) message is intended to configure and supervise the different devices in the substation. For this reason, unlike the other two messages, the MMS message has low or medium speed, consequently loose performance requirement.

There are two possible communication modes. The first one is the controller-IED communication mode where data transfer is done over a path between the substation controller and IEDs. The other one is the peer-to-peer mode where data can be delivered between IEDs directly. The current communication protocols only support the controller-IED communication mode, while the IEC 61850 protocol support the peer-to-peer mode too. In this paper we will consider the security protocols which can be applied to both modes.

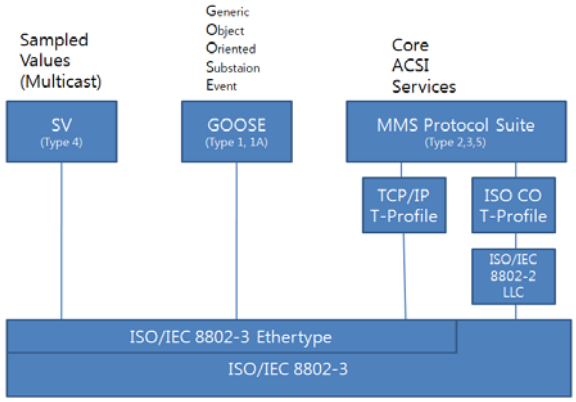

Fig. 1. The protocol stacks for the IEC 61850 messages [2]

III. SECURITY PROTOCOL

A. Goal

In this paper we focus on secure transfer of the sample value message and the GOOSE message. The secure transmission of these two messages plays a critical role for normal protection operation in the substation.

Receivers need to verify that messages are sent from claimed senders. Attackers can inject malicious messages to the IEDs or breakers, consequently causing system malfunctions. To authenticate the owner of messages is one of the most important security requirements in many applications in the SCADA system.

Receivers also need to make sure that the messages they receive are not altered on the way by attackers. In particular, sample values are used for the IEDs to decide whether voltage, current, or frequency anomalies happen. If these values are modified, the IEDs are mistaken to understand the current status. For this reason we consider the security protocol to guarantee the message authenticity and integrity.

B. Notation

A, B, S : communicating nodes, S denotes a server
K_{AB} : an session authentication key between A and B
AK_{SA}: a master authentication key between S and A
EK_{SA}: a master encryption key between S and A
$\{X\}_K$: encryption of X using key K
H : a hash function
M_A: message sent by A
$<M_A|M_B>$: concatenation of messages M_A and M_B
N_A: a nonce generated by A
C: a sync code used for verifying message freshness
.

C. Message Authentication and Integrity

In this paper we are using the message authentication code (MAC) to verify the authenticity of the sender and the integrity of the message. Two methods are used for computing MAC. One is just to use a block cipher. The procedure is simply to encrypt a message in cipher block chaining (CBC) mode and discard all cipher text block except the last block [4]. The last encrypted block is used for MAC. The block cipher algorithms we use here are Advanced Encryption Standard (AES) and SEED [5, 6].

The other method is to use the Keyed-Hashing (HMAC) [7]. In the HMAC, first, the sender A concatenates the Sync Code C, the original message M_A, and the (session) authentication key, K_{AB}, then computes MAC by applying a one-way hash function, H, to the concatenated. The Sync Code is non-decreasing number. The detailed procedures are explained in [6]. Next, the sender replaces the authentication key by the MAC and finally delivers the message. The complete message that A sends to B is:

$$MAC = H(C \mid M_A \mid K_{AB},)$$
$$A \rightarrow B : <C \mid M_A \mid MAC>$$

The node B applies the same hash function to obtain new MAC on the message it received with the authentication key. If these two MACs are the same, B can trust that the message was sent by the claimed sender A, and also the message was not modified on the way.

The authentication key can be of any length. But it is recommended that the key length should not be less than L bytes which is the byte-length of the hash function output, since it would decrease the security strength. Keys longer than L bytes are acceptable but the extra length would not provide significant increase of security [7]. In the implementation we use L=16 as a default secrete key length since the default keyed hash function is MD5 [8].

The Sync Code is used to verify the freshness of the message. Since the Sync Code is a non-decreasing number,

the value of a new message should be bigger than the one of an old message. Comparing these two values reveals whether the message was resent or not, thus ensuring that no attackers replay old messages.

D. Key Distribution

Because the substation controller or server is the base node which communicates with the other nodes in the network, compromising the server will cause the whole network to be out of service. Generally the server is deployed in the protected location.

We assume that the server is a trusted base, and all IEDs trust the server at the initial setup. At the creating time the server is given two master keys. One key is the master encryption key used for encrypting the session key which is the authentication key that two communicating parties share with. The other is the master authentication key used when the server delivers the session authentication key to IED.

Each IED is also given two same master keys which are shared with the server at the creation time, and the session authentication keys are distributed from the server whenever necessary. And we assume that each node keep the master keys without any harm.

For the key distribution between the server and IEDs, we apply the same simple and secure algorithm proposed in [9]. So here we focus only on the peer-to-peer communication mode. Direct message exchange between IEDs will be possible if the IEC 61850 standard is prevalent as the standard communication protocol.

Many protocols have been proposed for the three party server-based key distribution protocol from the Needham-Schroeder protocol to the ISO/IEC 11770-2 server-based protocols and the Kerberos protocol which are widely accepted as the de facto standard protocol in the network community [10, 11].

Even though the protocol proposed here has some similarities with the ISO/IEC 11770-2 and the Kerberos protocol, it has unique characteristics, reflecting the constraints of the system components in the SCADA system.

First, Any IED can initiate the key request to the server as the same way carried in the server-to-IED communication mode. This provides consistent operation to the server to process the key request regardless of the communication modes. According to the address fields in the key request, the server can differentiate which operation it should take.

Second, every difficult and delicate job is given to the server's shoulder, including random number generation and key generation. The case the IED has to generate a nonce is only the first step. Other than that, IEDs can get away with the obligation of clock synchronization or random value generation which are the delicate parts of doing security operation.

At every message exchange, message authenticity is guaranteed from MAC using the master authentication keys. The final message exchange at step 3 and 4 provides the key confirmation to both nodes.

1 A → S: $< A, B, N_A, H(A \mid B \mid N_A \mid AK_{SA}) >$

2 S → A: $< \{K_{AB}\}_{EK_{SA}}, H(N_A \mid B \mid \{K_{AB}\}_{EK_{AB}} \mid AK_{SA}) >$

2' S → B: $< N_A, A, \{K_{AB}\}_{EK_{SB}}, H(N_A \mid A \mid \{K_{AB}\}_{EK_{AB}} \mid AK_{SB}) >$

3 B→ A: $< N_A, B, H(N_A \mid B \mid K_{AB}) >$

4 A→ B: $< N_A, A, H(N_A \mid A \mid K_{AB}) >$

IV. EXPERIMENT

A. Configuration

The test configuration consists of four PCs each of which simulates a merging unit (MU), a substation controller, an IED, and an actuator IED which send a trip signal to a circuit breaker (CB) as shown in figure 2. All devices are operating on the 1G Ethernet. The MU sends sample values to the IED with 960 Hz speed. If the IED senses any anomalies in the sample values, it notifies the actuator of the event by sending the GOOSE messages. Then the actuator opens or closes the CB based on the information of the GOOSE message.

The security protocol for the message authentication and integrity is applied to every sample value and GOOSE message at the MU and the IED. Table 1 and 2 show the specifications of PC and software used in this experiment.

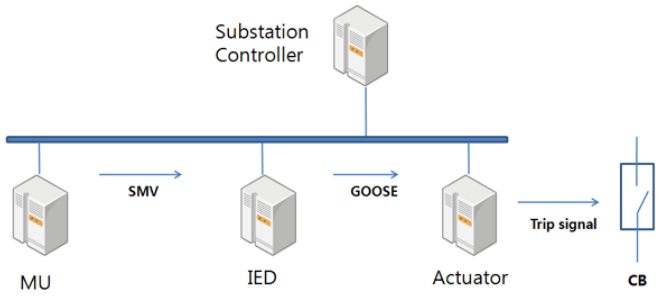

Fig. 2. The system configuration

TABLE I
SPECIFICATION OF PCS IN THE EXPERIMENT

CPU	Intel Pentium 4, 3Ghz
Memory	1GB
HDD	80GB
OS	Windows XP, SP2

TABLE 2
SPECIFICATION OF CRYPTOGRAPHIC SOFTWARE

Algorithms	Library	Optimazation
AES, SEED	openssl-0.9.8k	Not optimized
MD5, HMAC-MD5	openssl-0.9.8k	Not optimized

B. Results

In this experiment we measure the time which it takes to execute the security algorithms for the message authentication and integrity. The result is shown in table 3. The values are the average of 70,000 measurements, discarding the first 100,000 measurements.

As explained in section III-C, we implement two methods

for computing MAC. One is just to use a CBC of the block ciphers. The other method is to use the keyed-hashing (HMAC). In this table the AES256 which denotes the key length of 256 bits is the result by using the first method, since the MAC is automatically computed as the last encrypted block by the CBC mode. And the HMAC-MD5 denotes the second method. The keyed-hashing is giving the slightly better result than the block cipher method even though the difference is not significant. The reason of this meager difference is that the lengths of the sample values and GOOSE messages are extremely small, compared to normal messages. The message length of sample values in the experiment is 96 byte fixed size.

Based on the result, we can derive the following facts. First the time to be required for processing the message authentication and integrity algorithm is reasonably small satisfying the stringent timing requirement of the GOOSE message transmission. Also the security algorithms do not pose any obstacle to generate the sample value messages with 960Hz speed.

Second, the keyed-hashing (HMAC) method gives the better performance than the block cipher MAC, since HMAC can avoid any encryption procedure. But the improvement is not significant because the message length is small.

Finally, as expected, when we do the message encryption as well as the MAC computation, we encounter the significant increase in the message processing time. Therefore we need to avoid the message encryption unless the message confidentiality is required.

TABLE 3
COMPARISON OF MESSAGE PROCESSING TIME FOR SECURITY

Encryption	MAC	Time(μs)
AES256	-	5.38400
AES256	SEED	11.79450
AES256	MD5	7.23000
AES256	HMAC-MD5	9.15900
SEED		5.36000
SEED	AES256	10.13950
SEED	MD5	8.54600
SEED	HMAC-MD5	11.66550
-	MD5	2.82750
-	HMAC-MD5	4.68950

V. CONCLUSION

In this paper we consider the security protocol to provide the message authentication and integrity for transmitting the sample value message and GOOSE message defined in the IEC 61850 standard. By experiment we examine whether the security algorithm can meet the performance criteria specified in the standard. As explained in the section IV-B, the security algorithm can satisfy the performance requirement given the platform which we use in this experiment. We also show that the HMAC algorithm give the better performance than other message authentication algorithm even considering small size of the sample value or GOOSE messages. However, we need further investigation for validity of the security algorithm over the other system configuration with more restraint in computing power and memory, since the algorithm eventually will be implemented on the embedded, microprocessor-based platform in real world.

VI. REFERENCE

[1] J. Slay and M. Miller, “Lessons learned from the Maroochy Water Breach,” IFIP Springer Boston, vol. 253, pp73-82, 2007.

[2] IEC, “Communication networks and systems in substation ,” IEC 61850-8-1, 2004.

[3] IEC technical committee 57, “Data and Communications Security, Part 5: Security for IEC 60870-5 and derivatives,” IEC 62351-5 Second Committee Draft, December 2005.

[4] M. Stamp, *Information Security: Principles and Practice*. New York: Wiley, 2006, p. 55.

[5] M. Stamp, *Information Security: Principles and Practice*. New York: Wiley, 2006, p. 45.

[6] H.J.Lee, S.J.Lee, J.H. Yoon, D.H.Cheon, and J.I.Lee, “The SEED Encryption Algorithm,” IETF rfc 4269, December 2005.

[7] H. Krawcryk, M. Bellare, and R. Canetti, “HMAC: Keyed-Hashing for Message Authentication,” RFC 2104, IETF, February 1997.

[8] F. Baker and R. Atkinson, “RIP-2 MD5 Authentication,” RFC 2082, IETF, January 1997.

[9] I.H.Lim, S. Hong, M.S.Choi, S.J. Lee, S.W.Lee, B.N.Ha, "Applying Security Algorithms against Cyber Attacks in the Distribution Automation System," IEEE PES, April 2008.

[10] ISO, Information Technologies – Security Techniques –Key Management – Part 2: Mechanisms Using Symmetric Technique ISO/IEC 11770-2, 1996, International Standard.

[11] B. Clifford Neuman and T. Tso, “Kerboros: An authentication service for computer networks”, IEEE Communications Magazine, Vol. 32, No. 9, pp33-28, September 1994

VII. BIOGRAPHIES

Dae-Yong Shin(SM’09) was born in Bucheon in Korea, on September 26, 1985. He earned BS in computer science at Myongji Univ. in 2008. He is now working for MS in computer science. His interests are semantic web, web security, and network security.

Sugwon Hong(M’95) was born in Incheon in Korea, on January 7, 1957. He earned BS in physics at Seoul National Univ. in 1979, MS and Ph.D. in computer Science at North Carolina State Univ. in 1988, 1992 respectively.

His employment experience included Korea Institute of Science and Technology (KIST), Korea Energy Economics Institute (KEEI), SK Energy Ltd., and Electronic and Telecommunication Research Institute (ETRI). Currently he is a professor at Dept. of Computer Software, Myongji University since 1995. His major research fields are network protocol and architecture, network security.

Il Hyung Lim(SM’05) was born in Seoul, Korea, in 1979. He received his B.E., and M.S. degrees in Electrical Engineering from Myongji University, Yongin, Korea in 2007. He is now working for his Ph.D. in Myongji University. His research interests are power system control and protective relaying, including artificial intelligence application.

Seung-Jae Lee(M’88) was born in Seoul, Korea, in 1955. He received his B.E. and M.S. degrees in Electrical Engineering from Seoul National University, Korea, in 1979 and 1981, respectively. He received his Ph.D. degree in Electrical Engineering from the University of Washington, Seattle, USA in 1988. Currently, he is a Professor at Myongji University and a Director at NPTC (Next-Generation Power Technology Center). His major research fields are protective relaying, distribution automation and AI applications to power systems.