RESEARCH ON SECURITY ISSUES AND SOLUTIONS OF IEC 61850 COMMUNICATION PROTOCOL ARCHITECTURE

[1]SHENG ZHAO-YONG, [2]QU HAI-PENG, [3]WANG CHAO, [4]ZHOU XIAO-MEI

[1]Department of Computer Science and Technology, College of Information Science and Engineering ,Ocean University of China, Qingdao , Shandong 266100,China

[2]Department of Computer Science and Technology, College of Information Science and Engineering ,Ocean University of China, Qingdao , Shandong 266100,China

[3]Department of Computer Science and Technology, College of Information Science and Engineering ,Ocean University of China, Qingdao , Shandong 266100,China

4Department of Computer Science and Technology, College of Information Science and Engineering ,Ocean University of China, Qingdao , Shandong 266100,China

E-mail: [1]380352606@qq.com, [2]quhaipeng@ouc.edu.cn, [3]351616342@qq.com, [4]4996017087@qq.com

ABSTRACT

IEC 61850 Communication Protocol Architecture is widely used in China's electricity system for communication between the substation automatic systems. Due to the lacking of corresponding security specification, the standards cannot guarantee the confidentiality, integrity as well as authentication in communication. This paper proposes a solution for this problem. The improved Handshake Protocol and Record Protocol are introduced between the application layer and the transport layer with less transmission of data and quick connect feature. Using this solution, the standards can meet both the electricity system for real-time and reliability requirements and the security requirements at the same time.

Keywords: *IEC 61850, Security, Solution*

1. INTRODUCTION

IEC 61850 Communication Protocol Architecture is the international standard for substation communication network and system, enacted by the 57th Technical Committee of the International Electrotechnical Commission (IEC TC57) in 2004. Three working group 10, 11, 12 (WG10/11/12) belonged to IEC began to develop IEC 61850 in 1995. Now it is the only international standard for substation automation system[1] based on general-purpose network communication platform. It is also the basis of the power industry standard in our country. The members of three working group come from different countries and regions. They have a wealth of experience. Some of them have participated in the standard formulation work of a number of countries and regions. In the development of IEC 61850 process, they refer to and fully absorb the relevant existing international standards, including IEC870-5-101 Transmission Protocols, companion standards especially for basic telecontrol tasks [2], IEC 870-5-103 Transmission Protocols, Companion standard for the informative interface of protection equipment [3], UCA2.0 [4] (Utility Communication Architecture2.0) (substation and feeder equipment communication protocol system developed by the U.S. EPRI); ISO/IEC 9506 manufacturers specifications MMS [5] (Manufacturing Message Specification).

IEC 61850 expect to communicate between IED (Intelligent Electronical Device) via Ethernet. With clear specifications in the substation network communication protocols, the IEC 61850 based substation automation system demonstrated irreplaceable advantages. Due to the lacking of corresponding security specification[6], the network security issues which are increasingly prominent today have become the short board of this standard series. It is attracting increasing attention from us when the report of errors triggered by the network security issues has frequently occurred.

2. ARCHITECTURE INTRODUCED AND CHARACTERISTICS

IEC 61850 Communication Protocol series standard (Communication Networks and Systems in Substations) regulate the communication between substation intelligent electronic devices (IED) and related system requirements. By the series standardization of equipment, IEC 61850 form a standardized output between different devices and ultimately achieve interoperability between the different equipment and different companies. Its interface model is shown as Figure 1:

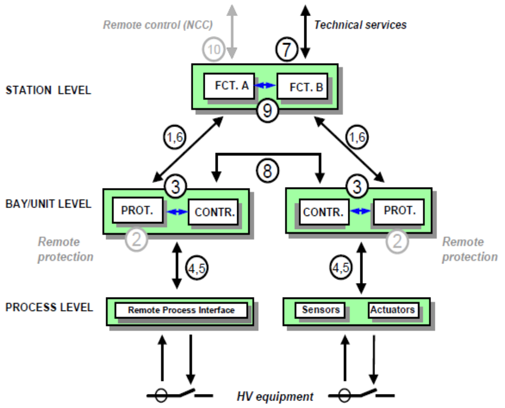

Figure 1 - Interface model of substation automatic systems

In order to achieve controlling and monitoring of substation equipment and relay protection and monitoring of the primary equipment and grid, IEC 61850 is divided into the substation layer, the spacer layer or unit layer, and process layer. Figure I details this three layers and logical interface between them. The main function of the process layer is to achieve full functionality of the interface process and communicate with a spacer layer through logical interface 4 and 5. The spacer layer can operate the primary device by using an interval of the data, achieving communications within the spacer layer through a logical interface 3, communicating with the process layer through logical interface 4 and 5, communicating with the substation layer by interface 1 and 6. The substations layer is mainly used for two-part function including the process related substation functions and interfaces related substation functions. The process related substation functions related to the process by using the interface 8, and a plurality of spaced or the whole station data, for monitoring and controlling of a plurality of intervals or the whole site of the primary equipment; the interfaces related substation functions including Man–Machine Interface, Remote Interface and some Management Interfaces. It is communicated through interface 1 and 6 with the spacer layer logical, and the logical interface 7 and the remote control interface when in communication with the outside.

IEC 61850 series standard system absorbs a lot of the international new state-of-the-art technology in the making process and introduced a large number of other international standards in other areas. Consequently, it is a very large standard series including 10 categories, 14 standards. It features:

1. The communication system of the substation is divided into three levels including the substation layer, the spacer layer and the process layer from the function logic being, and a communication interface between the layer and the other layer is fixed.

2. Using of object-oriented data modeling techniques.

3. Description of data itself and the transmission does not require pre-defined limit.

4. Network independence, designed to be used independent of the network and application layer protocol abstract communication service interface (ASCI).

3. SECURITY ISSUES FACED BY IEC 61850

The openness of IEC 61850 is an irreplaceable situation relative to traditional Substation communication protocol, but it brings many security issues to IEC 61850, which is unexpected when the protocol is made. With the rapid development of Internet technology, substation system based on IEC 61850 is faced with more and more threat from Internet. The main reason is that the general purpose of making this protocol is to solve equipment interoperability between different manufacturer rather than considering safety problems, which leaves potential safety hazard. Now accidents caused by issues of cyber security have been reported[7], and damage of each accident is huge, so solving safety problems of IEC 61850 is extremely urgent. But which measures should be taken need careful consideration because power system needs high real-time and if security measures are taken improperly, the operation of the power system will be affected seriously.

4. ANALYSIS OF THE CHARACTERISTICS OF PACKET IEC 61850

IEC 61850 communication service model is divided into two modes of client-server mode and peer-to-peer communication mode. Messages are divided into seven types; these seven types are divided into five kinds of categories. Three kinds Category are commonly used:

1. MMS service messages are mainly used to monitor the network, the less demanding real-time.

2. GOOSE message, or universal oriented substation event model packets, mainly is used for the transmission interval lockout signal and real-time trip signal, high real-time requirements.

3. SV packets, or value message, are mainly based on serial unidirectional multidrop point to point link transmission of sampled values.

The mapping relations are shown as below:

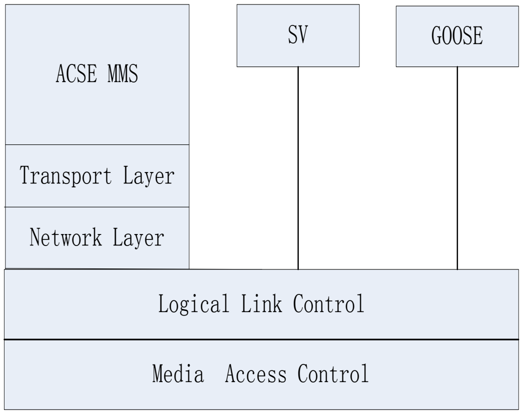

Figure 2 - Overview of functions and framework

Considering the high real-time requirements of SV packet and GOOSE messages and time consuming of encryption and decryption algorithm[8], these two agreements are based on the plaintext transmission in network with empty transport layer and the empty network layer. The message goes directly to the link layer without through the transport layer or network layer. MMS, the basis of IEC 61850 solution, is an application layer protocol with less demanding real-time. The solution of his paper is focused on MMS messages.

5. SOLUTION

5.1 Overview of the Solution

In order to solve the problems of authentication, integrity and confidentiality, when these substations based on IEC 61850 communications with each other, we improved this standard. The specific approach is to add the improved Handshake Protocol and Record Protocol between the application layer and the transport layer, the position in the TCP/IP is as shown in Figure 3.

During the handshake, client and server will negotiate various parameters; these parameters are used for authentication of client and server, and as the encryption key in this session. The traditional sense of the handshake takes a very long time and there will be a large amount of data in the handshaking procedure between the client and server, obviously, this does not meet the real-time requirements of the power system, so in this article we use a fast connection in the handshaking procedure. In the power system, the model of substation are the most, and they will not change frequently, so we can specify encryption algorithms and compression algorithms between client and server, instead of negotiating in the handshaking procedure. Communicating parties are not always creating a new session in a connection; they will take a resume session Handshake instead of full handshake within a period of time when they implement a full handshake.

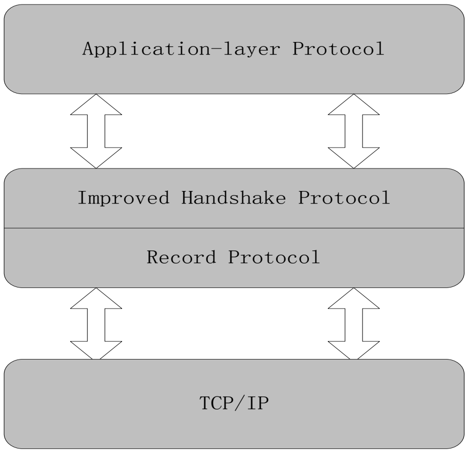

Figure 3 - The position in the TCP/IP

Record protocol use the session parameters, which negotiated in the handshake, to transport data, it will ensure the confidentiality and integrity of data.

5.2 Handshake Protocol

The main purpose of the Handshake Protocol is to complete the authentication of the communicating parties, and negotiate some parameters of the session. In this paper, the handshake process is divided into two modes: Full

Handshake and Resume Session Handshake. When communicating parties first establish a connection or Session ID exceed the prescribed period of time (in this paper for the guarantee period for 24 hours), they need use Full Handshake; in other situations, they use Resume Session Handshake. Firstly, we will introduce the Full Handshake. Figure 4 shows the detailed process.

1. The client sends a client hello message to the server, client hello message includes client protocol version, random number (used to calculate session master key), session ID(right now is empty).

2. When the server receives the client's message, it will send a server hello message too. The server hello message includes the protocol version, random number and generated session ID. The session ID, protocol version and the server random number will be stored in the client cache.

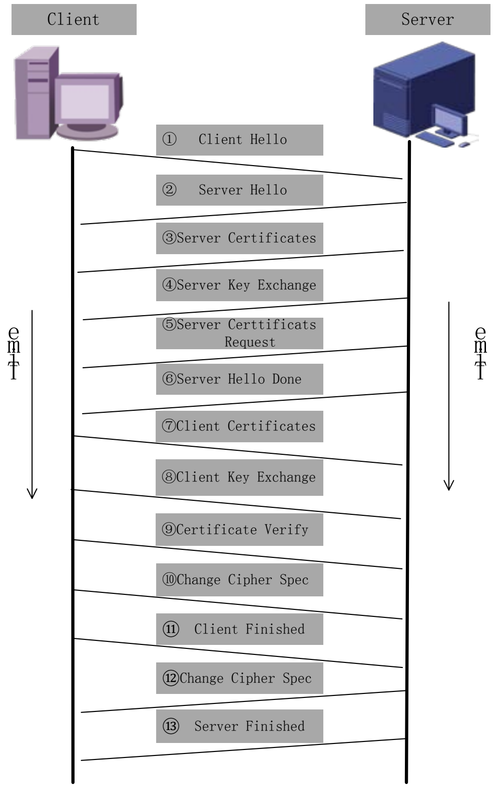

Figure 4 - Full Handshake

3. The server sends a server certificates message. After the client receives the message, it will authenticate the server, only if the server can be successfully authenticated, the client proceeds to step 4.

4. The server sends the server key exchange message, which contains the public key of the RSA encryption algorithm, it is used to encrypt data in the handshake procedure. The client saves the server's public key into the cache as write key in this session, when receives the server key exchange message.

5. The server sends request client authentication message, for the client identity authentication.

6. The server sends a server hello done message, then the client can sends message to the server.

7. The client certificates message is sent by the client, including his certificate for the server to the identity authentication.

8. Client key exchange message sent, using all data generated in the handshake thus far, the client creates the pre-master secret for the session, encrypts it with the server's public key (obtained from the server's certificate, sent in step 4), and then sends the encrypted pre-master secret to the server.

9. Send certificate verify message, the message is let the server validation messaging client and the client Certificate is the same one.

10. Send change cipher spec message, then the client will take the master key(generated by the pre-master) as the session write key. Receiving this message, the server will then take the newly generated symmetric key as the read key.

11. Then sends the client finished message.

12. The server will send its own change cipher spec message, then take the master key as session write key. And the client takes the master key as the session read key when it receives this message.

13. The server sends server finished message.

The full handshake is now completed and the session begins. The client and the server use the session keys to encrypt and decrypt the data they send to each other and validate its integrity.

Next introduce Resume Session Handshake:

1. The client send client hello message to the server, but the session ID is not null. The server search in the session ID list, find this session ID and verify the identification is still in the guarantee period, then start a resume session handshake.

2. The server find the master key in the cache, then sends the change cipher spec message,

and takes the master key as the session write key. The server sends server finished message at the end.

3. Receive the server's change cert spec message, the client takes the master key as the session read key. Then send the change cipher spec message and take the master key as the session write key, at the end sends client finished message.

4. Receive the client finished message; the server will take the master key as the session read key.

After the above four steps, the resume session handshake is now completed and the session begins. It can be found that more simpler processes than the full handshake can be proved in this way be able to save a lot of time. To achieve the above results, the client and the server must store the session ID and session master key in the cache after each full handshake, so we can use this information in resume session handshake. Meanwhile, after receiving the client's session ID, the server needs to check the validity, to decide use a full handshake or a resume session handshake.

5.3 Record Protocol

Application data will be split and compressed by record protocol, then encrypted by encryption algorithm. Its processes are as follows:

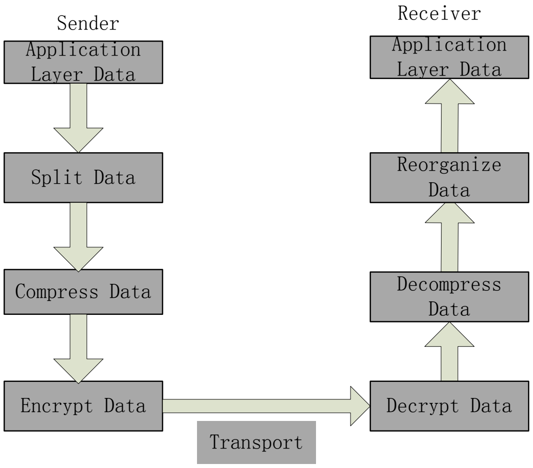

Figure 5 - Record Protocol

In the sender, first of all split data to meet the requirements. Then MAC operatess on data so that the recipient can take an integrity check for the data. Finally, encrypt the data and send to the receiver.

In the receiver, firstly decrypt the data use the master key. Then decompress and reorganize data, and send to application layer.

By the above procedure we can find that this protocol provides two services for SSL connections:

1. Confidentiality - using conventional encryption.

2. Message Integrity - using a Message Authentication Code (MAC).

6. IMPLEMENTATION BASED ON THE OPENSSL

OpenSSL is a powerful cryptography-based security software development kit. It is the open source, and freely available. Eric A. Young and Tim J. Hudson started to write it in 1995[9], succeed in the implementation of the Secure Sockets Layer protocol (SSL) and Transport Layer Security protocol (TLS), which function is quite powerful and comprehensive, encompassing the major cryptographic algorithms, commonly used keys and certificates package management features, also including a wealth of applications for testing[10]. The whole OpenSSL package probably can be divided into three main parts: cryptographic algorithm library, SSL protocol library, as well as application[11].The OpenSSL directory structure is planning naturally around three functional parts. OpenSSL provides a total of eight symmetric encryption algorithm, including seven kinds of block cipher algorithm, only a stream encryption algorithm is RC4. The quality of the random number has an important impact on the security of the key. OpenSSL provides a set of solutions to achieve a random number generation and management. OpenSSL also provides other auxiliary functions, such as API key generated from the password, certificate issuance and management of the configuration file mechanisms and so on.

It is the characteristics of the OpenSSL that saves us the large amount of work in achieving. We use the command applications, the source code and libraries API of OpenSSL. The former is mainly used for the key and certificate management. We don’t make too much description about that. We have done a lot of work in OpenSSL source code modifications to reach the requirements. The first step is to modify the Client Hello and Server Hello message structure, remove the list of cipher suites and compression algorithm list that we don’t need any more. Just as the previously mentioned, these two parts are fixed in each release. Secondly, since the encryption algorithm specified in the front, i.e., in the key exchange algorithm using the RSA algorithm, the symmetric encryption algorithm using DES. As we can see from Table 1, with the increase of the length of the key of the RSA algorithm, the encryption speed decreases gradually, the same data required more time, so in order to meet the real-time requirements, we decided to use 512 - bit key length(If the security requirements are particularly high, and have a faster processor, you can use 1024 or more key length). And the DES algorithm encryption speed can reach 2 Mbit/s, it can completely meet the demand. So in the fourth step of the whole process handshake process, the server needs to send its own public key. In the eighth step, the client is required to produce a 48-bit random number as the pre-master, using server public key to encrypt then sent it out.

Finally, and most importantly, in order to use Resume Session Handshake, we must save the session ID and the pre-master in the cache for a long time. And the session ID default valid modification for 24 hours, the server will test and verify the session ID when using the Resume Session Handshake.

Table 1 –The RSA algorithm's performance with different key lengths

RSA key length(bit)	sign/s	verify/s
512	2677.0	30759.1
1024	504.6	9243.1
2048	74.7	2520.4
4096	10.5	663.7

Table 2 – Two kind of handshake need time contrast

NO. of handshake	Traditional handshake protocol	Improved handshake protocol
First handshake(ms)	153	150
Second handshake(ms)	149	83
Third handshake(ms)	159	79
Average time(ms)	153	104

Remark: Intel(R) Core(TM)2 DUO CPU, 2.10GHZ

When the server and client communicate with each other, the handshake process takes a lot of time, the article uses the improved handshake protocol can significantly save the time of the handshake process, we compared with traditional handshake procedure, the results are shown in Table 2. As can be seen, in the first handshake, two ways consume almost the same time, but improved handshake in the subsequent two times in the handshake shows a greater advantage, and the average time is less than the traditional process of handshake.

This paper involved a lot of function in OpenSSL SDK, in this one no longer say, only list the main function in the Full Handshake:

1. int Client_Hello(SSL *s) Function: Send the Client Hello message to the server.

2. int Server_Hello(SSL *s) Function: Send the Server Hello message to the server.

3. Int Send_Server_Certificate(SSL *s) Function: The server throughs the function to send its own certificate to the client, the certificate information has been loaded in ssl_ctx initialization process.

4. int Send_Certificate_Request(SSL *s) Function: To request the client certificate.

5. int Send_Server_Done(SSL *s) Function: The server has been sent, waiting for the client response.

6. int Send_Client_Certificate(SSL *s) Function: The client sends its own certificate for the server to verify.

7. int Send_Client_Key_Exchange(SSL *s) Function: The client use this function to send pre-master to the server.

8. int Send_Cert_Verify(SSL *s) Function: Let the server validation message client is the client certificate true owner.

9. int Send_Change_Cipher_Spec(SSL *s, int state_a, int state_b) Function: Notice to the other, has replaced the session write key.

10. int Send_Finished(SSL *s, int a, int b, const char *sender, int slen) Function : Following change cipher spec message sending out, marking the success of the party session negotiation end.

7. INNOVATION

The paper introduces improved handshake protocol and record protocol in IEC 61850 for the first time. The improved handshake protocol has the advantages of less transmission data and quick connect feature, and it can adapt to the requirements of real-time and security in power agreement. In the first stage of the whole process of handshake, communication will transmit less data relative to the traditional handshake protocol. That is because in this process, both two sides do not need to transfer the list of cipher suites and compression algorithm. Encryption algorithm and compression algorithm remind constant in every version and they do not need to negotiate by communication. Take cipher suites for example, each one need two bytes to represent, up to as high as 124 KB which will no doubt occupy a great deal of flow. Another innovation of this paper is fast connection. Both sides do not need to perform the Full Handshake before each connection. Here the author assumes that Full Handshake will be performed every 24 hours and in other time periods, the Resume Session Handshake will be performed. From the above statement, we can be find that it just needs to change a session identifier relative to the whole process of handshake. After the Validation of the server-side, the process of handshake is done. The time of this process, regardless of the exchange of data, or the handshaking process are greatly reduced.

8. CONCLUSIONS

This is a huge security risk to the power system that the Communication information transmitted in plaintext forms between the substations based on the IEC 61850. But this is a new topic in the field of information security research which is still in its initial stage. This paper introduces the improved Handshake Protocol and Record Protocol to the IEC 61850 communication protocol system for the real-time and reliability requirements of the power system, as well as the security requirements of communications. It has important practical significance.

REFERENCES:

[1] REN Yan-ming, QIN Li-jun, YANG Qi-xu. Study On IEC 61850 Communication Protocol Architecture [J]. Automation of Electric Power Systems, 2000, 24(8).

[2] IEC 870-5-l01． Telecontrol Equipment and Systems--Companion Standard for Basic Telecontrol Tasks． 1995.

[3] IEC 870-5-103． Telecontrol Equipment and Systems--Companion Standard for the Informative Interface of Protection Equipment ． 1997.

[4] UCA 2．0． Utility Communication Architecture. 1998.

[5] ISO / IEC 9506． Industrial Automation Systems—Manufacturing Message specification． 1990.

[6] LONG Lin-de，LI Jing，LIU Li-li. Research on communication security of substation automation system based on IEC 62351 [J]. Journal of Changsha Telecommunications and Technology Vocational College, 2010, 9(3).

[7] MO Jun, TAN Jian-cheng, Research on network security in substations based on IEC 61850[J]. Telecommunications for Electric Power System, 2009, 30(198).

[8] YIN Zhi-liang, LIU Wan-shun, YANG Qi-xun, QIN Ying-li. Generic Substation Event Model Based on IEC 61850[J]. Automation of Electric Power Systems, 2005, 29(19).

[9] GONG Shao-lin. Research on Realization Mechanism of Cryptography Security Platform Based on OpenSSL[J]. Computer & Digital Engineering, 2011, 39(6).

[10] GUO Jing, WANG Ying-guan. CA certification and SSL communication based on openssl[J]. Modern Electronics Technique, 2012, 35(3).

[11] CAO Zi-jian，RONG Xiao-feng. Study of data security application based on OpenSSL[J]. Journal of Xi'an Polytechnic University, 2011, 25(6).