Combining Mifare Card and agsXMPP to Construct a Secure Instant Messaging Software

Ya Ling Huang, Chung Huang Yang
Graduate Institute of Information & Computer Education,
National Kaohsiung Normal University
M9353705@stu93.nknu.edu.tw, chyang@computer.org

摘要

近年來，電話、傳真與郵件一直是人類溝通的主要方式。而隨著網際網路的發達與普及，溝通的方法又更見多元化，從早期的電子郵件(E-mail)，到近年來的即時通訊(Instant Messaging;IM)，網際網路不僅使溝通的方法更便利，也使得溝通可以更即時且更全面化，其中又以IM最具代表性，它具備了電話的即時性以及電子郵件的便利性。但是，大部分的IM皆不提供加密，資訊都是採用明文傳送，私密與安全問題因此衍生。因此，如何保護通訊的雙方，資訊不外洩，不被有心人竊取，是一個大家都關心的重要議題。本研究採用開放原始碼agsXMPP加上AES演算法，建置一個安全的即時通訊管道。並且建立金鑰分配中心(Key Distribution Center;KDC)來管理金鑰。另外，再搭配Mifare Card，將金鑰儲存於卡片上，私鑰能隨身帶著走，充滿便利性。

關鍵詞：IM, agsXMPP, AES, KDC, Mifare Card

Abstract

Nearly, the telephone, fax and mail have been main ways used by mankind for communicating with each other. With the development and popularization of the Internet, the way of communication is more and more pluralistic, from the E-mail to the IM (Instant Messaging). Internet make the way to communicate not only more convenient but also more immediately and more all-round. Among them, IM is the most representative. It has both the convenience of the E-mail and instant of the telephone. However, most IM do not offer encrypting, information transfer is plaintext (Encryption converts data to an unintelligible form, called ciphertext. [6] Decrypting the ciphertext converts the data back into its original form, called plaintext.[6]) type so the secret and security problem has produced. How to protect both sides of the communication? It is an important topic that everybody cares about the information is not let out or stolen by the attacker. To construct a security instant communication channel, this research adopts the open source, agsXMPP and AES algorithm. Besides, we build the KDC (Key Distribution Center) to manage the key. The secret key is stored on the Mifare Card. It enables users to take away with themselves to improve convenience.

Key words: IM, agsXMPP, AES, KDC, Mifare Card

Introduction

Recently, people get used to IM for their communication. At present, ICQ, MSN Messenger, Yahoo! Messenger, Skype and etc. can be regarded as representatives of the IM. The function of these kinds of IM software for people requirement is getting more and more, but does not do enough in the security. The attacker can eavesdrop the information by IM sniff program so that the security issue has been occurred. The motive of this research is to strengthen the security of the IM and protect both sides of the communication to avoid the information let out or stolen by the attacker. The agsXMPP is the IM software of a set of open source and it follows the XMPP (eXtensible Messaging and Presence Protocol). The XMPP is based on XML so that it can express all structured information. This research adopts the agsXMPP to build a safe IM software and it combines the Symmetric Key Encryption – AES. We assume the KDC has be trust by all users and use the KDC to manage the secret key. The secret key is stored on the Mifare Card. It enables users to take away with themselves to improve convenience. Therefore, we can make sure the process of communication is more safe and it gives users at rest.

Literary reviews

XMPP

The XMPP accorded with the IETF standard of the IMPP norm. The XMPP was based on XML so that it could express all structured information.[13]

The XMPP was originated from Jabber technology. The technology was a kind of the instant communication system solution used to transmit the real-time and in-situ information. The XMPP had developed by open source group from Internet. The first product of the Jabber technology was a service platform which was an asynchronous and high expanding. The service platform provided the message of real-time and in-situ information and offered the function of the instant communication system, such as AIM, ICQ, MSN or Yahoo! messenger etc.. The Jabber was an open source plan with the characteristics of open, free and understood easily. Now the Jabber had many products of the open source on different platforms, such as Windows, Linux, Java VM etc., and it was also including Jabber server, client and developing the procedure library, etc.. At the end of 2002, the XMPP WG was established by the foundation of the Jabber software in IETF to discuss the relevant technology publicly. They devoted to XMPP standardization so that the members of the foundation become authors of the core of the XMPP. Therefore, the core of the communication protocol of the Jabber used XMPP which was based on XML so that the Jabber had high expansion and was easy to combine with the third-party product.[13]

The topological structure of the Jabber network was the same as the e-mail system. Every client needs a local server to receive and send the message. It used the TCP socket and port 5222 between the client and the server, the port 5269 between server and server to transmit messages and in-situ information. In the client-server module, all messages and information could reach other clients through servers. Though the transmission channel could be set up directly before the messages transmit to the client, the communication consulting of these connections was finished through Jabber server at first. Therefore, in the relational networks, it could exist a lot of Jabber servers at the same time and each server worked alone and had its own client. Both two servers could connect to each other so that it could communicate with each other and transmit information.[13]

Under the Jabber structure, the main work of Jabber server was as follows:[13]

I. To deal with the connections of Jabber client and communicate with client directly.

II. To communicate with other Jabber server.

III. To deal with client registration, authentication, in-situ information, good friend's list and taking off line information etc..

The main work of Jabber client was as follows:[13]

I. To set up the connection with Jabber server.

II. To parse and interpret the XML stream.

III. To understand the core data type of the Jabber.

There were four RFC technical standard documents made by IETF XMPP WG:[7]

I. RFC 3920

This document confirmed the core features of the XMPP which was a protocol for streaming XML elements to exchange structured information in close to real time between any two network endpoints. While XMPP provided a generalized, extensible framework for exchanging XML data, it was used mainly for the purpose of building instant messaging and presence applications that met the requirements of RFC 2779.[9]

II. RFC 3921

This document described extensions and applications of the core features of the XMPP that provided the basic IM and presence functionality defined in RFC 2779.[10]

III. RFC 3922

This document described a mapping between XMPP and CPIM specifications.[11]

IV. RFC 3923

This document defined the methods of end-to-end signing and object encryption for XMPP.[13]

AES

In 2001, NIST (National Institute of Standard and Technology) adopted the AES(Advanced Encryption Standard) for new encryption algorithm to replace the DES. AES puts forward by Dr.Joan Daemen and Dr.Vincent Rjimen. AES was designed according to the following characteristics:

I. To oppose all known attacks.

II. To carry out fast on various kinds of platforms and source code was succinct.

III. Design was easy to understand.

The input and output for the AES algorithm each consisted of sequences of 128 bits (digits with values of 0 or 1). These sequences will sometimes be referred to as blocks and the number of bits. They contained will be referred to as their length. The Cipher Key for the AES algorithm was a sequence of 128, 192 or 256 bits. Other input, output and Cipher Key lengths were not permitted by this standard.[6]

There were four stages in the Cipher process of AES:[6]

I. SubBytes

The transformation in the Cipher that processed the State using a nonlinear byte substitution table (S-box, Non-linear substitution table was used in several byte substitution transformations and in the Key Expansion routine to perform a one-for-one substitution of a byte value.) that operated on each of the State bytes independently.

II. ShiftRows

The transformation in the Cipher processed the

State by cyclically shifting the last three rows of the State by different offsets.

III. MixColumns

The transformation in the Cipher took all of the columns of the State and mixes their data (independently of one another) to produce new columns.

IV. AddRoundKey

The transformation in the Cipher and Inverse Cipher in which a Round Key(Round keys were values derived from the Cipher Key using the Key Expansion routine; they were applied to the State in the Cipher and Inverse Cipher.) was added to the State using an XOR operation. The length of a Round Key equaled the size of the State (i.e., for Nb = 4, the Round Key length equaled 128 bits/16 bytes).

There were also four stages in the inverse Cipher process of AES:[6]

I. InvShiftRows

The transformation in the Inverse Cipher was the inverse of ShiftRows.

II. InvSubBytes

The transformation in the Inverse Cipher was the inverse of SubBytes.

III. InvMixColumns

The transformation in the Inverse Cipher was the inverse of MixColumns.

IV. Inverse of the AddRoundKey

System analysis and design

This research is developing and building a security IM system that is based on XMPP to make sure of the safe communication environment.

Research framework

This main structure of research is based on agsXMPP, so that security improvement should be done.

I. agsXMPP connect method

When using agsXMPP to carry on instant communication at present, the information is transparent totally (figure 1), because there is no encryption. The talking can be taken out easily as long as the attackers use the IM sniff program.

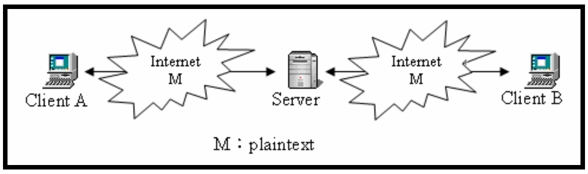

Figure 1. agsXMPP connect method

II. Modified agsXMPP connect method

The threat of IM is privacy mainly. This research is directed against privacy to provide an effective method, no matter in server system or in the communication process, make sure the content of the communication is security and privacy.

The modified method of this research is as follows:

The process of the communication with the encryption mechanism of the security makes sure the privacy. The user stores the secret key by Mifare Card, You can not decipher without the secret key so that the privacy of the information is high extremely.

About the modified agsXMPP connect method is as figure 2.

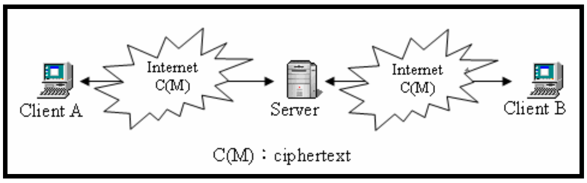

Figure 2. Modified agsXMP connect method

Research tools

I. agsXMPP

agsXMPP is a SDK for XMPP written in managed C# dedicated to .NET and Mono technologies. Our SDK could be used for XMPP client, server and component development.[1, 2]

II. C#

C#(C sharp) language is developed from Microsoft. It is for developing component of Internet application program and service on .NET platform. C# has strong function of C/C++ and simaple to use, like Visual Basic. C# is a object-oriented program language, it is same as C++ and Java.[3]

III. Visual Studio 2005

Visual Studio 2005 Professional Edition is developed from Microsoft, it is an omni-directional development environment for speciality developer. It is be used for setting up the high efficiency and the multi-layer type structure with of Windows, Web and action device application program. Its characteristics are still as follows:

1 It is an integrated Visual database tool that is used for designing the database, table, stored

procedures, etc..

2 It is an integrated Vision database design and reporting tool.

3 It is designing, debugging and disposing the multi-layer application program.

IV. Borland C++ Builder 6

The Borland C ++ Builder is a kind of vision procedure language. So long as using the tool of the toolbox, you can finish the user interface of figure in designing program phase. The Borland C ++ Builder develops already sixth edition so far and it is based on C language and combine with the concept of the object and Windows GUI relevant functions to accommodate user developing the procedure.

V. Mifare Card and reader

1. Mifare Card

Mifare Card used in this research is MIFARE 1 S50. The price is cheap and easy to use. MIFARE 1 S50 is the most extensive used at present. The external specification of Mifare 1 S50 is in accordance with the norm of ISO 7810. The internal structure of its card is as figure 3.[14]

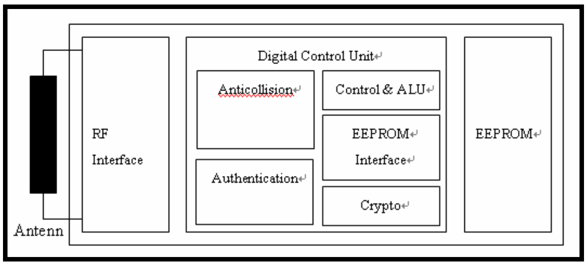

Figure 3. Internal structure of MIFARE 1 Card

2. Reader

This research adopt the "Castles Technology" reader that type is EZWAVE104U (figure 4). It can support Mifare Card and ISO 14443 type A/B card. Besides, it offers the transmission interface of USB or RS232, support Microsoft 98, ME, 2000 and XP of operation system. In addition, the purpose of the reader is extensive, the price is also reasonable and easy to buy.[5]

Figure 4. EZWAVE1043U reader

System Practice

Practice for Client system

Both sides of the communication must install the client system software to keep instant communication safety.

I. System function

The function of the client system is as follows:

1. It can read the secret key from Mifare Card.

2. It can decrypt the ciphertext.

3. It can encrypt the plaintext.

II. Decryption and Encryption process

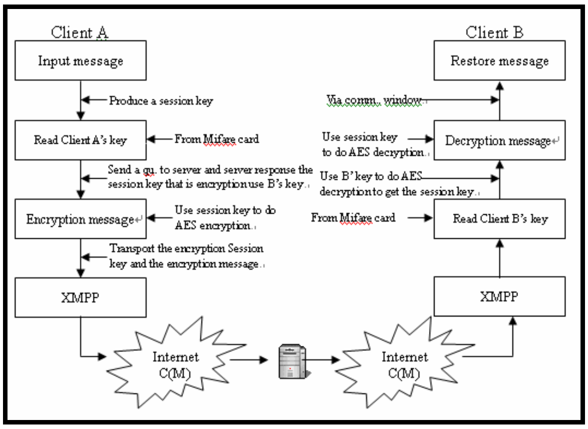
Client A
Client B
Input message
Restore message
Produce a session key
Via comm.. window
Read Client A's key
From Mifare card
Use session key to do AES decryption
Decryption message
Send a qu. to server and server response the session key that is encryption use B's key
Use B' key to do AES decryption to get the session key
Encryption message
Use session key to do AES encryption
From Mifare card
Read Client B's key
Transport the encryption Session key and the encryption message
XMPP
XMPP
Internet C(M)
Internet C(M)

Figure 5. Decryption and encryption process

The Decryption and encryption process is as figure 5. If client A would like to establish instant communication with client B, client A will input message in a dialogue box and press the “send” button. This message will make AES encryption calculations with the secret key of client B so that the ciphertext could be generated. The ciphertext will stand on XMPP to produce the xml document which was send to server and from server to client B through Internet.

When client B received the xml document, it will extract the ciphertext and read secret key from Mifare Card. By decryption calculations with the secret key and the ciphertext, we can get the plaintext message.

III. Development environment

It is such as table 1.

Table 1. Client system Dev. environment

Purpose	Component
Operation system	MS XP Professional
Dev. environment	Visual studio 2005
Compiler	Borland C++ Builder 6
Encryption Algorithm	AES
Mifare Card Dev.	Mifare Card Reader(EZWAVE104U)

Practice for KDC server system

I. System function

The function of the KDC server system is as follows:

1. All users in the system trust the KDC server.

2. The KDC server system stores all secret keys that are privacy between the KDC and every user.

3. The KDC server system provides the secret key for users to make communication.

II. Key distribution process

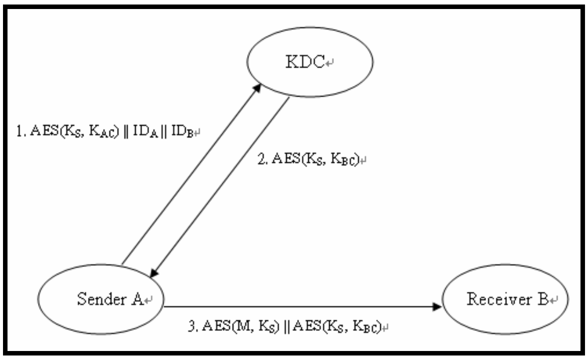
KDC
1. AES(K_S, K_{AC}) || ID_A || ID_B
2. AES(K_S, K_{BC})
Sender A
Receiver B
3. AES(M, K_S) || AES(K_S, K_{BC})

Figure 6. The key distribution process

The key distribution process is as figure 6. Assume:[4,8]

1. The client A would like to establish instant communication with the client B。

2. There is no secret key between the client A and the client B.

3. The client A and the client B trust the KDC server system.

Then:

1. The Client A find the client B in connecter list and push right button of the mouse to choose dialogue box, then:

(1)The client A read the secret key K_{AC} from the client A's Mifare Card to generate a session key K_S (the session key is a random number, length is 128 bytes). The client A use the secret key K_{AC} and session key K_S to encrypt AES(K_S, K_{AC}), then transfer AES(K_S, K_{AC})||ID_A||ID_B to KDC server.

(2)The KDC server receives AES(K_S, K_{AC})||ID_A||ID_B and then use A's secret key K_{AC} to decrypt and get the session key K_S. The KDC server use ID_B to find the client B's secret key K_{BC} and then use the client B's secret key K_{BC} and session key K_S to encrypt AES(K_S, K_{BC}). Finally, the KDC server response AES(K_S, K_{BC}) to the client A.

(3)Final, the dialogue box is opened and waited for the client A to input message.

2. The client A input the plaintext message M in the dialogue box and press the "send" button, then:

(1)The client A use the plaintext message M and the session key K_S to encrypt AES(M, K_S). Finally, it transfer AES(M, K_S)||AES(K_S, K_{BC}) to the client B.

3. When the client B receive AES(M, K_S)||AES(K_S, K_{BC}), then:

(1)The client B read the secret key K_{BC} from the client B's Mifare Card first and then use the secret key K_{BC} and AES(K_S, K_{BC}) to decrypt and get the session key K_S. Finally, the client B use the session key K_S and AES(M, K_S) to decrypt and get the plaintext message M.

III. Development environment

It is as table 2.

Table 2. KDC server system dev. environment

Purpose	component
Operation system	MS Windows XP Professional
Dev. environment	Visual studio 2005
Compiler	Borland C++ Builder 6
Database	MS SQL server 2005
Encryption Algorithm	AES
Mifare Card Dev.	Mifare Card Reader(EZWAVE104U)

Conclusion

This research adopts the open source agsXMPP and AES algorithm to build a security instant communication channel. Besides, we also build the KDC to manage the key and combine the Mifare

Card to store the secret key. It enables secret key to take away with oneself to improve convenience. However, when the attacker invades and attacks the KDC system on purpose, the KDC system will have no security. In the future, the system can be improved further so that we can use the control key mechanism, like CA(Certification Authority) system instead of the KDC system, to make the security more stronger.

Acknowledgement

This work was supported in part by TWISC@NCKU, National Science Council under the Grants NSC 94-3114-P-006-001-Y.

References

[1] agsXMPP SDK,

http://www.ag-software.de/index.php?page=agsxmpp-sdk

[2] agsXMPP Version 0.92,

http://www.ag-software.de/index.php?page=download

[3] C sharp, http://en.wikipedia.org/wiki/C_Sharp

[4] Chung Huang Yang, Network Security : Theory and Practice, Key Management, pp.7-2~7-8, May, 2006。

[5] Contacnless Reader -- EZWAVE1043U reader,

http://www.casauto.com.tw/contents/products2.asp?minicidx=24&SN=26

[6] FIPS 197: Advanced Encryption Standard, November, 2001

[7] Jabber, http://en.wikipedia.org/wiki/Jabber

[8] POPE79, Popek, G.L., and C.S. Kline, "Encryption and Secure Networks," ACM Computing Surveys, Vol. 11, No. 4, Dec. 1979, pp. 331-356.

[9] RFC 3920: Extensible Messaging and Presence

Protocol (XMPP): Core, October, 2004

[10] RFC 3921: Extensible Messaging and Presence Protocol(XMPP): Instant Messaging and Presence, October, 2004

[11] RFC 3922: Mapping the Extensible Messaging and Presence Protocol (XMPP) to Common Presence and Instant Messaging (CPIM), October, 2004

[12] RFC 3923: End-to-End Signing and Object Encryption for the Extensible Messaging and Presence Protocol (XMPP), October, 2004

[13] XMPP Standards Foundation,

http://www.xmpp.org/

[14] Ying jing Chiou, RFID practice – contacnless smart card system develop”, Mifare Card, pp.7-1~7-21, July, 2005Chung Huang Yang, Network Security : Theory and Practice, Key Management, pp.7-2~7-8, May, 2006