ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

Key Distribution for Symmetric Key Cryptography: A Review

Yashaswini J

Assistant Professor, Department of Studies in Computer Science, Pooja Bhagavat Memorial Mahajana Post Graduate Centre, K.R.S. Road, Metagalli, Mysuru, Karnataka, India

ABSTRACT: In today's digital communication era sharing of information is increasing significantly. The information being transmitted is vulnerable to various passive and active attacks. Therefore, the information security is one of the most challenging aspects of communication. Cryptography plays an integral role in secure communication and it provides an excellent solution to offer the necessary protection against the data intruders. One of the cryptographic technique is a symmetric cryptography; In this technique the sender and receiver use the same key to do encryption and decryption of the data. This secret key must be shared between the sender and the receiver. Therefore this paper presents a study on distribution of key among the sender and receiver in symmetric cryptography and also it gives the study on authentication of the clients in distributed network. Many protocols are used to make key distribution among the clients and authentication of the clients in distributed network. The main two are Needham-Schroeder key distribution protocol and Kerberos protocol.

KEYWORDS*:* Cryptography, Symmetric Cryptography, Key distribution

I. INTRODUCTION

Cryptography is the art of achieving security by encoding messages to make them non-readable [1]. Cryptography not only protects the information but also provides authentication to the user. Here the original information and encrypted information are referred as plaintext and cipher text respectively. The transformation of plaintext into unintelligible data known as cipher text is the process of encryption. Decryption is the reverse process of encryption i.e. conversion of cipher text into plain text. During communication, the sender performs the encryption with the help of a shared secret key and the receiver performs the decryption. Cryptographic algorithms are broadly classified as Symmetric key cryptography and Asymmetric key cryptography. In symmetric key cryptographic algorithms single key is used for both encryption and decryption process [1]. Fig 1.1 illustrate that the general procedure for symmetric key cryptography.

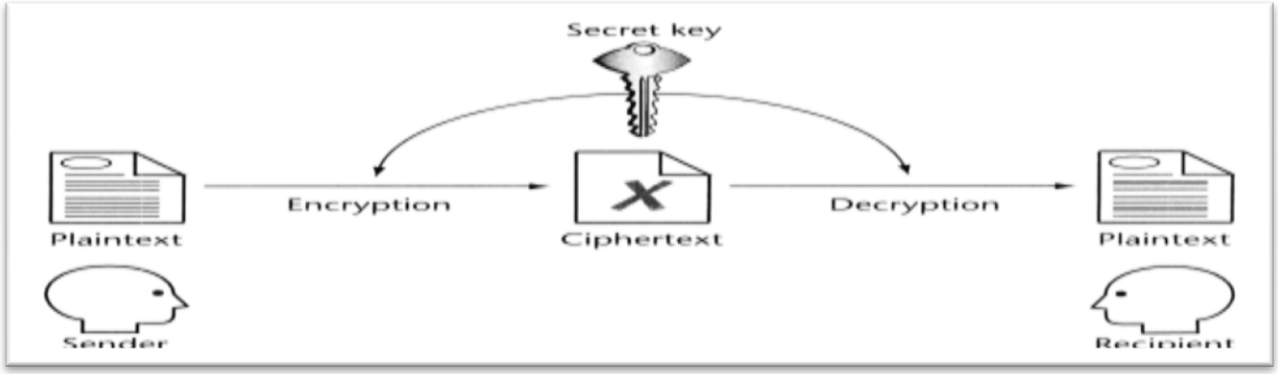

Fig1.1: Symmetric Key Cryptograph

For any cryptographic technique to work, the two parties to make a communication must share a key. If it is symmetric encryption then the secret key must share between the two parties to exchange information in secure way.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

II. KEY DISTRIBUTION FOR SYMMETRIC KEY CRYPTOGRAPHY

The major problem in symmetric key cryptography is that of the key distribution because the key must be shared secretly [2]. Keys can be distributed by any one of the following ways:

1. Sender can select the key and physically deliver it to receiver.
2. A trusted third party can select the key and physically deliver it to the sender and the receiver.
3. If sender and receiver have previously and recently used a key, one party can transmit the new key to the other, encrypted using the old key.
4. If sender and receiver each has an encrypted connection to a third party, then the third party can deliver a key on the encrypted links to sender and receiver.

Options 1 and 2 call for manual delivery of a key, for end-to end encryption over a network, manual delivery is awkward. In a distributed system, any given host or terminal may need to engage in exchanges with many other hosts and terminals over time .Thus, each device needs a number of keys supplied dynamically. The problem is especially difficult in a wide-area distributed system. Options 3 is a possibility for either link encryption or end-to-end encryption, but if an attacker ever succeeds in gaining access to one key, then all subsequent keys are revealed. Even if frequent changes are made to the link encryption keys, these should be done manually. To provide keys for end-to-end encryption, option 4 is preferable [3].

For option 4, two kinds of keys are used:

- Session key: When two end systems (hosts, terminals, etc.) wish to communicate, they establish a logical connection (e.g., virtual circuit). For the duration of that logical connection, called a session, all user data are encrypted with a one-time session key. At the conclusion of the session the session key is destroyed.
- Permanent key: A permanent key is a key used between entities for the purpose of distributing session keys.

A necessary element of option 4 is a key distribution center (KDC). The KDC determines which systems are allowed to communicate with each other. When permission is granted for two systems to establish a connection, the key distribution center provides a one-time session key for that connection. In general terms, the operation of a KDC proceeds as follows:

1. When host A wishes to set up a connection to host B, it transmits a connection request packet to the KDC. The communication between A and the KDC is encrypted using a master key shared only by A and the KDC.
2. If the KDC approves the connection request, it generates a unique one-time session key. It encrypts the session key using the permanent key it shares with A and delivers the encrypted session key to A. Similarly, it encrypts the session key using the permanent key it shares with B and delivers the encrypted session key to B.
3. A and B can now set up a logical connection and exchange messages and data, all encrypted using the temporary session key[3].

Next section gives the different approaches used in distribution of keys for end-end encryption i.e., in distributed network.

II.I Key Distribution and Authentication Protocols

Cryptography is also used to support the mechanisms for authenticating communication between pairs of parties. Authentication protocols are all about distribution and management of secret keys. Key distribution in a distributed environment is an implementation of distributed authentication protocols. Based on this idea many key distribution and authentication protocols have been proposed. Generally, all protocols assume that some secret information is held initially by each principal. Authentication is achieved by one principal demonstrating the other that it holds that secret information. All protocols assume that system environment is very insecure and is open for attack. So any message received by a principal must have its origin authenticity, integrity and freshness verified. To achieve these goals, most protocols need to rely on an authentication server and this server should have the following features [4].

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

- Capable: An Authentication server delivers good-quality session keys and distributes them to the requesting principals securely.
- Trustworthy: Authentication server maintains a table containing a name and a secret key for each principle. The secret key is used only to authenticate client processes to the authentication server and to transmit messages securely between client processes and the authentication server.

Key distribution and authentication Protocols are divided into two categories to verify the authentication of a message. First category uses nonce and challenge/ response handshake to verify freshness, example is Needham-Schroeder Protocol. Second category uses timestamps and assumes that all machines in distributed system are clock-synchronized; example is Kerberos Protocol [4].

II.I.I Needham-Schroeder Key distribution Protocol

It is a secret-key protocol based on nonce session keys i.e., number used once session keys and also provides a solution to authentication and key distribution based on an authentication server [5]. This protocol is based on the generation and transmission of ticket by the authentication server. A ticket is an encrypted message containing a secret key for use in communication between A and B [5]. Table 1 gives the summary of message exchange in Needham-Schroeder Key distribution Protocol.

Table 1: Summary of message exchange in Needham-Schroeder Key distribution Protocol

1. A->S : A, B, N_A

A requests S to supply a session key for communication with B

2. S->A: $\{N_A, B, K_{AB}, \{A, K_{AB}\} K_B \} K_A$

S returns a message encrypted in A's secret key, containing a newly generated key K_{AB}, and a ticket encrypted in B's secret key

3. A->B: $\{A, K_{AB}\} K_B$

A sends the ticket to B

4. B->A: $\{N_B\} K_{AB}$

B decrypts the ticket and uses the new key K_{AB} to encrypt another nonce N_B

5. A->B: $\{N_B - 1\} K_{AB}$

A demonstrates to B that it was the sender of the previous message by returning an agreed transformation of N_B

II.I.II Kerberos Protocol

Kerberos is a key distribution and user authentication service developed at MIT. Kerberos is a computer network authentication protocol which works on the basis of 'tickets' to allow nodes communicating over a non-secure network and nodes prove their identity to one another in a secure manner. It is aimed primarily at a client–server model and it also provides mutual authentication. It gives protection against eavesdropping and replay attacks. The need for Kerberos protocol is, when using the services of an open distributed network, the service providing server must identify the authorized workstation otherwise there will be a possibility of three types of threats [3], they are given as,

- A user may gain access to a particular workstation and pretend to be another user operating from that workstation.
- A user may alter the network address of a workstation so that the requests sent from the altered workstation appear to come from the impersonated workstation.
- A user may eavesdrop on exchanges and use a replay attack to gain entrance to a server or to disrupt operations.

In any of these cases, an unauthorized user may be able to gain access to services and data that he or she is not authorized to access. Rather than building elaborate authentication protocols at each server, Kerberos provides a centralized authentication server whose function is to authenticate users to servers and servers to users. Kerberos relies exclusively on symmetric encryption [3]. There are five versions of Kerberos present, the first three are internal to MIT, the version 4 and versions 5 are available commercially. A Kerberos server is known as a Key Distribution Centre (KDC). Each KDC has an authentication service (AS)

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

and a Ticket Granting service (TGS) [6].Figure 2.2 gives the overview of the Kerberos protocol[1]. Kerberos deals with three kinds of security object:

- Ticket: a token issued to a client by the Kerberos ticket-granting service(TGS) for presentation to a particular server, verifying that the sender has recently been authenticated by Kerberos. Tickets include an expiry time and a newly generated session key for the use by the client and the server .
- Authenticator: a token constructed by a client and sent to a server to prove the identity of the user and the currency of any communication with a server. It contains client's name and a timestamp and is encrypted in the appropriate session key.
- Session key: a secret key generated by Kerberos and issued to a client for use when communicating with a particular server.

Working module of Kerberos version 5: The client authenticates to AS using a secret key (User login password) and receives a ticket from the AS. Later the client can use this ticket to get additional tickets from TGS for server. A Kerberos ticket has a fixed period of validity starting at time t1 and ending at time t2 [7]. A ticket for a client C to access a server S takes the form:

{C, S, t1, t2, Kcs} Ks, which we denote as {ticket(C,S)} Ks.

To obtain a ticket for any server S, C constructs an authenticator encrypted in KcT of the form:

{C, t} KcT, which we denote as {auth(C)} KcT

In a first step client obtain a session ticket and TGT ticket by passing his secret key to AS and TGS and these tickets are once per login session.

1. C->A: C,T, n.

Client C requests the Kerberos authentication server AS to supply a ticket for communication with the TGS T

2. A->C: {KcT, n, {ticket(C, T)} KT} Kc.

Returns a message containing a ticket encrypted in its secret key and a session key for C to use with T.

In a second step, Client obtain ticket for a server S, once per client-server session

3. C->T: {auth(C)}KcT, {ticket(C, T)} KT, S, n

C requests the ticket-granting server T to supply a ticket for communication with another server S

4. T->C: {Kcs, n, {ticket(C,S)} Ks} KcT

T checks the ticket. If it is valid T generates a new session key Kcs and returns it with a ticket for S (encrypted in the server's secret key Ks).

In a third step, client allowed to access a service on server requested with a ticket

5. C->S: {auth(C)} Kcs, {ticket(C, S)} Ks, request, n

C sends the ticket to S with a generated authenticator for C and a request.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

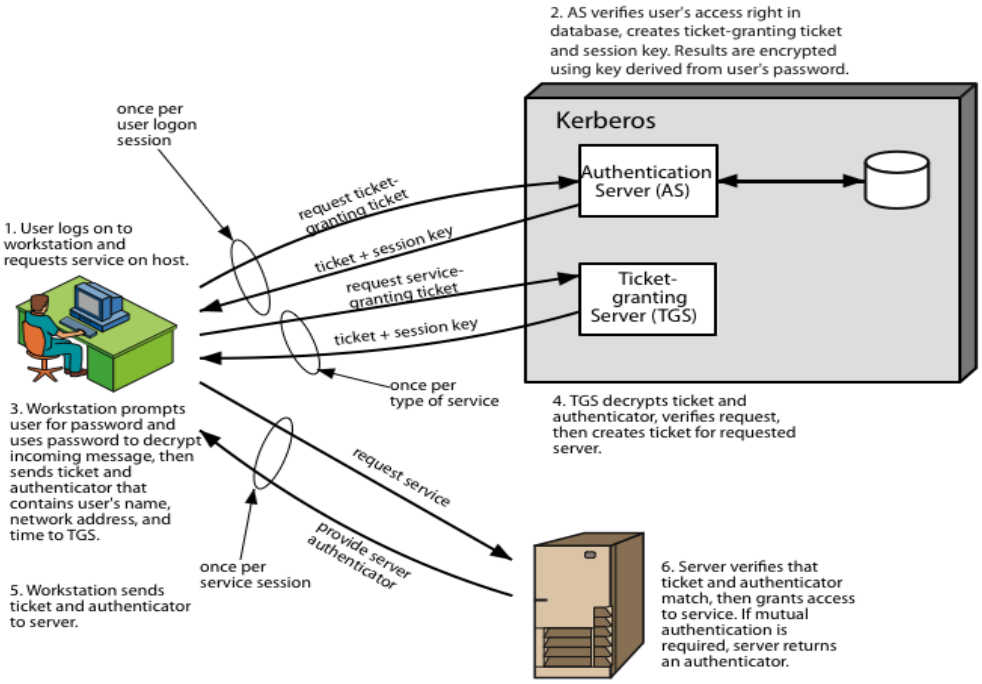

Figure 2.2: Overview of Kerberos.

III. CONCLUSION

In this paper, key distribution for symmetric key cryptography is studied. Cryptography is a technique to convert a plain data in to non-readable form. We can convert the plain data into non-readable from using symmetric key cryptography. Symmetric cryptography uses only one secret key to encrypt and decrypt the data. To do the encryption and decryption in symmetric cryptography we need to share a key between two parties. The key can be distributed in different ways between two parties. If there is a point to point encryption the key can easily distributed, but if the key need be share in end to end encryption then we can use trusted third party to distribute the keys (key distribution center) between the sender and receiver. In a distributed network, there is also need for authentication of the client who requesting the services of a server. Many authentication and key distribution protocols are used; the main two are Needham-Schroeder key distribution protocol and Kerberos protocol. In Needham-Schroeder key distribution protocol, the key distribution center generates a number once used session keys to allow access to the server services by the client. By the number of the session key allotted by KDC, sever can identify the authorized work satiations. In Kerberos protocol, each session key generated by the Kerberos KDC server will have time stamp associated with it, so that after some time it is going to expire. The server can easily identify the authorized client by checking the validity of the session key.

REFERENCES

[1] Ayushi, "A Symmetric Key Cryptographic Algorithm", International Journal of Computer Applications, ISSN: 0975 – 8887, Vol. 1, No. 15, 2010
[2] Christof Paar, JanPelzl, and Bartpreneel, "Understanding Cryptography: A Text book for student and Practitioners",Springer, 2010.
[3] William Stallings, " Network Security Essentials Application and standards", 4th edition published by Prentice Hall.
[4] Randy Chow, Theodore Johnson and Addison-Wesley, "Distributed Operating Systems and Algorithms" 1997.
[5] http://en.wikipedia.org/wiki/Needham-Schroeder_protocol.
[6] http://web.mit.edu/kerberos/, 2007/10
[7] http://en.wikipedia.org/wiki/Kerberos_(protocol), October 2008