International Journal of Advanced Technology in Engineering and Science
Vol. No.4, Issue No. 04, April 2016
www.ijates.com
ijates
ISSN 2348 - 7550

SECURE TEXT AND IMAGE MESSENGER SYSTEM

Mr. Ganesh Pai[1], Mr. Abhijeet Bajpayee[2], Mr. Derek Noronha[3]

[1,2,3]*Computer Engineering, St. John College of Eng & Tech, Mumbai University, Palghar*

ABSTRACT

The primary goal of this paper is to focus on securing text and images while sending or storing in a multimedia messaging network. Encryption algorithm plays a major role in security of images. Images can be highly confidential and nowadays images are used in several processes. We have seen in the past that in most of the application in smart phones, the images can be viewed easily as there is no process to keep the images secured. The images received can be directly viewed in the gallery this is a major problem in chat applications. In our project we are developing a chat application for military purposes such that the image received will be encrypted using blowfish algorithm. Blowfish algorithm consumes less power and is efficient. The encrypted image will be stored in the database so even if the hacker hacks our smart phone he will not be able to retrieve the information as it will be in the encrypted form.

Keywords: Encryption, SmartPhone, confidential, Blowfish Algorithm

I. INTRODUCTION

In cryptography, encryption is used which is the process of encoding messages or information in such a way that only authorized entities can read it. In an encryption scheme, by using the encryption algorithm the information can be encrypted using the encryption algorithm that will generate the cipher text. [1] An authorized receiver with the help of key provided by the sender can easily decrypt the message. It will be hard for an unauthorized user to decrypt the message as he will not have the key. The purpose of encryption is that only authorized person can access the data and will be able to read it using the decryption key. Somebody, who is not allowed to access the information, can be avoided from doing that because he does not have the required decryption key. Without the decryption key it is almost impossible to read the information.

Anything that offers a transmission of messages from sender to receiver or vice versa is known as online chat. To respond quickly by the other user chat messages are generally shorter.

There are a lot of chat applications but the security provided by these applications is not up to the mark as the images received by the users can be directly viewed in the gallery.

For data storage and transmission information security is very important and we know that images are widely used in several processes. Therefore, the security of image data from unauthorized access is very important.

The images received in the smart phone can be viewed easily in the gallery and some of the images might be important for the users.

II. EXISTING SYSTEM

The existing system fails to deliver security. In the existing system, the sender is sending the data not in an encrypted way. So any intruder can read the message, also the server contains the data backup so if server goes

down eventually the security of data will go down. The data can be easily accessed in the smart phone.

Presently we have seen that in most of the applications the images that are received are directly stored in the gallery. There is no security provided for the images received over such application. These images can be viewed easily by a third party. The third party can easily break the app lock. The images traveling through Internet are also not in encrypted form and can be easily accessed by intruders.

III. WORKING

Encryption is basically a system which can be used for hiding a message so that it can be transferred safely over the internet.

Encrypt a plain text with the help of a key (secret). You can then decrypt the message using the same key. You can send the data over the network by encrypting it using a key and then at the other end the same key is used to decrypt the data. None of the intruders in between can read it because they don't know the key.

[6] By using a key Alice encrypts a message for Bob, then by using the same key Bob decrypts it. Here after sending the encrypted message Bob needs to know the key to decrypt it, but we cannot take the risk of sending it over an unprotected or open network. The key cannot be encrypted because we will require another key to encrypt it.

When a user gets registered on our chat application then a key pair is generated. The key pair which is generated consists of a public key and a private key. So now both Alice and Bob have a key pair. The private key which is generated is stored on the device of our chat application server does not need it and it will not even have access to it. The server stores the public key and it gives your public key to the user to that person with whom you wish to exchange messages. So it ensures that Alice gives Bob her public key and Bob gives Alice his public key. The key pair algorithm that it is using allows the following mathematics to happen. Once both the users wish to chat with each other Alice gets Bob's public key and by using its private key it generates a secret key. Bob derives the same secret key by taking Alice's public key and with his own private key. The shared secret key which is generated is unique to both Alice and Bob, and as the private keys will remain private no one but only they will know about it. This shared will never be exposed to the application server. To exchange information safely this secret key can be used.

The images received by this chat application will not be viewed in the gallery. User will have to log in into his account to view the images.

Proposed architecture is showed in figure 1

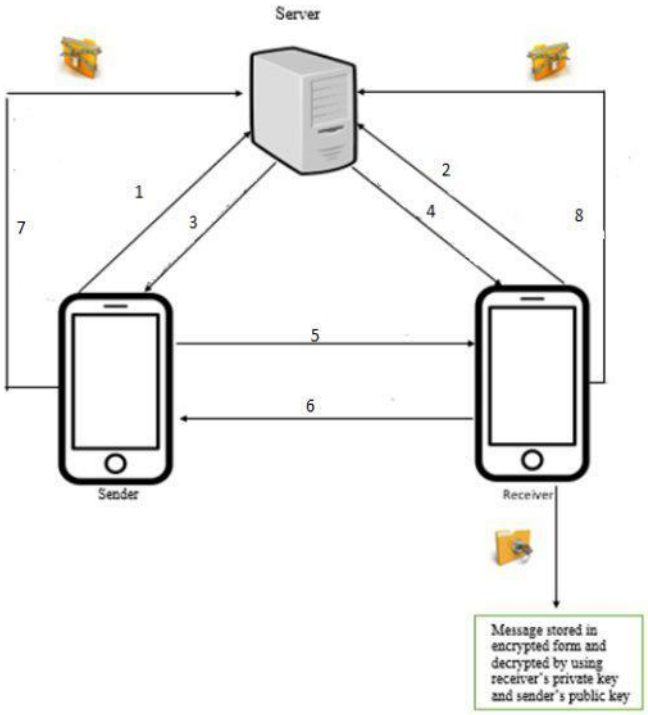

Figure 1: Proposed Architecture

Note: From the above figure.

1) The user(sender) will login if he is registered or the user will first register.

2) The second user(receiver) will login if he is registered or the user will register.

3&4) As soon as both users are registered the server will generate keys the server will send the private key to the user and public key of the user will be stored in the server. The server does not store the private key of the users'

5) The sender will add the receiver.

6) If the receiver accepts, the server will send the public key of the receiver to the sender and vice versa.

7) The sender will encrypt the message using receiver's public key and his private key.

8) The receiver will decrypt the message using sender's public key and his private key.

IV. DIFFIE-HELLMAN

Diffie-Hellman is basically a method for two computers to generate a secret key and exchange of information over an insecure channel.

Here we are using Diffie-Hellman algorithm for the exchange of public key and for the calculation of the secret key. The secret key that will be generated by using the private and public key of the user will be common for both the user.

V. USER AUTHENTICATION

Applications always need to know the identity of a user. By knowing a user's identity it allows an app to grant

them permission to access their data and provide a customized experience. Authentication is a process of proving a user's identity.

When the client logs in it generates a signature using the user's private signing key with the information provided such as name of the user that's the user name, password of the user and randomly generated data. Then the server validates the client's signature, password and username and then checks it against the stored public signing key. After successful verification, the client is issued with a session cookie which is used for authentication until the user logs out or till the session expires.

Once session cookies are exchanged over SSL they are thought to be a secure mechanism, but in the future every request could be checked with the signature. In end to end encryption a session hijacker will be unable to decrypt the messages.

Figure 2.1: Login page

Figure 2.2 Conversation taking place

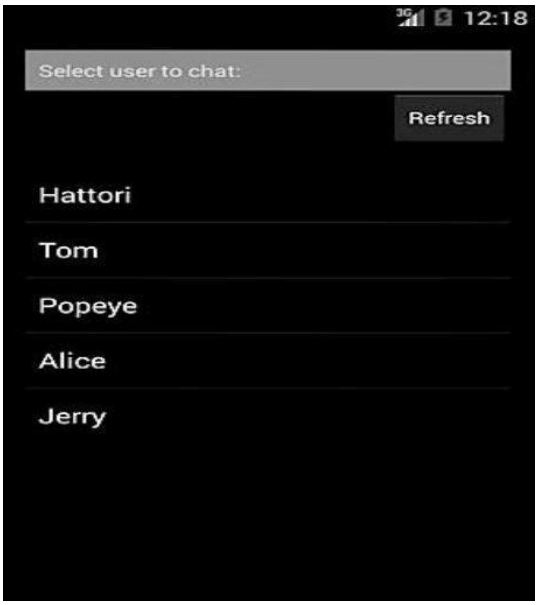

Figure 2.3 User's to chat

VI. CONCLUSION

In this paper, we have provided an open specification for a secure and privacy preserving chat service. The aim of this paper is basically to develop mobile chat services and try to explore any complexities involved in such a service providing privacy and protection to its customers from intruders. In this work we explored whether the technical challenges regarding the privacy protection can be built into a chat service. We also found that most of the technical components were already available and with some modifications a strong privacy preserving chat application could be constructed.

This is an end-to-end chat application which will provide the user's protection and privacy for their images and text. There were no serious issues faced during the implementation of the framework regarding the technology or performance.

We would also like to experiment with scalability and performance of the chat server in the future that might reveal some complexities and also maintaining a privacy based server. Here another potential aspect could be that how a text and image chat application could be developed into a voice and video chat application. Voice and video chat application will be more challenging than this.

REFERENCES

[1] Hiral Rathod, Mahendra Singh Sisodia, Sanjay Kumar Sharma "Design and Implementation of Image Encryption Algorithm by using Block Based Symmetric Transformation Algorithm", Dec 2013.

[2] Prof. Mrinmoy Ghosh and Prof. Pranam Paul "An Application to ensure Security through Bit-level Encryption" IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.11, November 2009.

[3] Diaa Salama Abdul, Elminaam, Hatem Mohamed Abdul Kader and Mohie Mohamed Hadhoud "Performance Evaluation of Symmetric Encryption", May 2010 .

[4] Priya Mehrotra, Tanshi Pradhan and Payal Jain "Instant Messaging Service on Android Smartphones and Personal Computers", May 2014.

[5] Rahul Verma, Ruchit Gupta, Manas Gupta, Rahul Singh "A Complete Study of Chatting Room System based on Android Bluetooth", Dec 2015.

[6] Raja Naeem Akram and Ryan K. L. Ko "End-to-End Secure and Privacy Preserving Mobile Chat Application", June 2014.