Providing Security to the Architecture of Presence Servers

Vimitha R Vidhya Lakshmi
PG Scholar
TKM Institute of Technology
Kollam, Kerala, India

Anju J
Assistant Professor
TKM Institute of Technology
Kollam, Kerala, India

ABSTRACT
Mobile cloud Computing is a new emerging technology. This technology supports many applications and one such application is presence enabled application. Mobile presence server is the main component in presence enabled applications and they provide presence services to all mobile users in the network. As number of mobile user increases to a great extent there arises buddy list search problem. In-order to solve this problem a scalable server architecture called presence cloud has been introduced. To provide security to the above server architecture SSL protocol is used. The performance is evaluated by using the parameters such as search latency and response time. SSL protocol involves determination and exchange of keys which increases buddy list search delay and response time. Therefore SSL session resumption is used to decrease this searching time and response time.

Keywords
Mobile cloud computing, Presence server, Presence cloud, SSL protocol, SSL session resumption.

1. INTRODUCTION
Mobile cloud computing is a new technology and is combination of both Mobile computing and Cloud computing. Mobile computing is human and computer interaction by which a computer is expected to be transported during normal usage. Cloud computing is a technology that allows sharing computing resources rather than having local servers or personal devices to handle applications. In Mobile cloud computing heavy tasks such as data processing and data storage happens in the cloud i.e. outside the mobile devices.

Presence enabled applications are supported by mobile cloud computing technology. Examples of presence enabled applications include social network applications such as Facebook, Twitter etc. and other Instant messaging applications like Yahoo Messenger etc. The main essential component of presence enabled applications is mobile presence server which provides mobile presence services. Mobile presence services include sharing of presence information to all mobile users in the network. Presence information includes information like user's availability, preferences, mood, activity etc. Each mobile user has a friends list or buddy list which contains list of contacts towards which the mobile user wants to communicate with.

When a mobile user changes its presence information it is the duty of mobile presence server to inform other mobile users in his/her buddy list about this change in presence information and also to notify itself about the change in presence information of other mobile users in the buddy list. For a small number of mobile users this buddy list searching can be done easily but as the number of mobile user increases this searching becomes more difficult. This is known as buddy list search problem. This is a scalability problem. In-order to solve this scalability problem a scalable server to server architecture is needed and it is known as presence cloud.

No security is provided to the above server architecture, so it suffers from many communication security problems such as user impersonation and man-in-the-middle attack. SSL (Secure Socket Layer) protocol is used to provide security to the above architecture. The performance of this architecture is evaluated by mainly two parameters: search latency and response time. By using SSL protocol search latency and response time increases because of key computation and exchange between client and server. To reduce search latency and response time SSL session resumption protocol is used.

This paper is organized as follow: section 2 is about related works regarding the topic, section 3 is about system model, the performance is evaluated in section 4, section 5 is about conclusion and future scope and finally section 6 is about references.

2. RELATED WORKS
Presence enabled application services in mobile devices have been growing widely over these years. An overview of the architecture, features and functions of three most popular Instant messaging (IM) systems is discussed in [1]. The traffic characteristics of AOL Instant Messenger and Microsoft Messenger are discussed in [2]. Most IM traffic is due to presence information. Mesh based technology and Distributed Hash Table (DHT) had been used earlier for buddy list searching but these technologies increases search cost and search latency therefore a scalable quorum based server to server architecture called presence cloud is introduced in [3]. Presence Network Agent is proposed to improve the presence services in [4].

As wireless and mobile communication increases the need for security also increases. In [5] the insecurity of wireless networks is discussed. [6] is about the privacy and security issues of multimedia services. Security in mobile cloud computing is studied in [7]. In [8] and [9] SSL is used as a security protocol and many cryptographic algorithms and procedures are explained here. The study about energy consumption characteristics of SSL protocol is done in [10].

3. SYSTEM MODEL

3.1 Overall View

Each mobile user is connected to presence cloud via 3G or Wi-Fi services. Then by using a secure hash algorithm that user is directed to one of the presence server in presence cloud. Then a direct TCP connection or control path is created between this mobile user and presence server. It is through this secured path all request and response for buddy list searching is done. This secured path is created by SSL protocol. Fig 1 shows the overall system architecture of presence cloud.

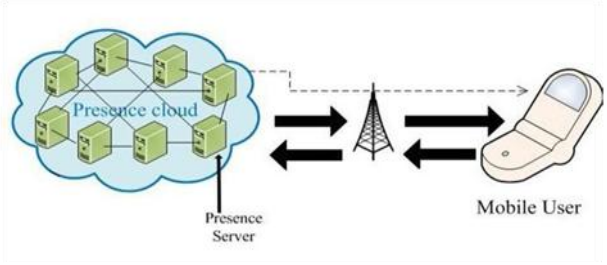

Fig 1: System architecture of presence cloud

3.2 Presence Cloud Design

The presence cloud is designed by three main parts: presence cloud server overlay, one hop caching strategy and directed buddy search. Presence cloud is a quorum based server to server architecture and is based on the concept of low diameter property. Every presence server maintains a presence server list or pslist. Here each user is a mobile user so they change their position from time to time which in turn makes changes in pslist. In-order to maintain pslist a stability process should be carried out which is done in presence cloud stabilization algorithm. Each presence server maintains its own user list and cache list. This is known as one hop caching strategy. Then a directed buddy list searching algorithm is used to solve the buddy list search problem. Thus presence cloud solves the scalability problem. The detailed working and algorithm explanation is given in [3].

3.3 Presence Cloud Security

No security is given to the above designed architecture. Therefore the above server to server architecture suffers from communication security problems such as user impersonation and man-in-the-middle attack. User impersonation is a method where an unauthorized user acts as an authorized one. Man-in-the-middle attack is an attack in cryptography where the attacker secretly and possibly alters the communication data between two parties who believe that they are directly communicating with each other. These attacks can be solved by proper authentication of user and end to end encryption of communication path. Both this is done by SSL protocol.

SSL protocol is developed by Netscape. SSL protocol mainly consists of two phases: SSL handshake phase and Data transfer phase. In SSL handshake phase the secret key for encryption and decryption is determined and exchanged between SSL client and SSL server. Whereas in data transfer phase the communication data is encrypted and decrypted using this determined secret key and this secured data is exchanged between SSL client and SSL server. SSL mainly contribute three features:

- SSL protocol permits server authentication to client
- SSL protocol permits client authentication to server
- Permits end to end encrypted secured path between client and server

3.3.1 SSL Architecture and Working

The SSL protocol lies above the TCP layer. SSL protocol is divided into four sub protocols. They are SSL record protocol, SSL handshake protocol, SSL change cipher spec protocol and SSL alert protocol. SSL record protocol decides in which format data is to be transmitted between client and server. The two main phases occurs in the SSL handshake protocol. SSL cipher spec protocol is a part of handshake protocol and is used to create new cipher spec for data transfer. SSL alert protocol is used to create alert messages such as fatal or warning messages. The SSL architecture is shown in fig.2.

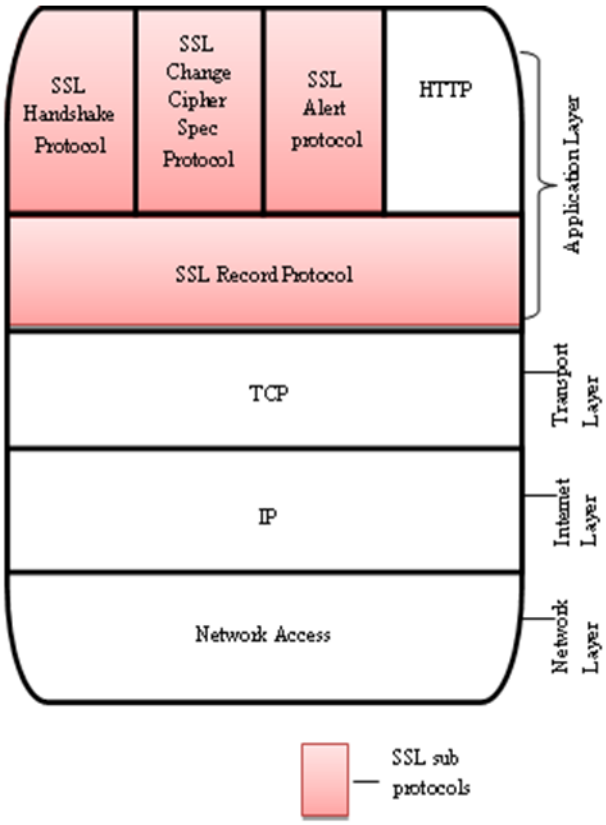

Fig. 2: SSL Architecture

Certificates are issued by certificate authority and are issued in-order to check the authenticity of the client or server. When a server needs a certificate, it sends request to certificate authority along with its public key. Certificate authority on receiving this request creates certificate and this certificate is converted to finger print by using message digest algorithm. This certificate is then converted to digital signature by encrypting the fingerprint with certificate authority's private key. Then this digital signature is send to client by server for authentication. Client on receiving this digital signature checks for server authenticity by decrypting this digital signature to fingerprint with the help of certificate authority's public key. Then this digital signature is converted to fingerprint by client independently using some other mechanisms. These two fingerprints are then checked by client whether they are same. If same then the server is an authorized server if not alert messages are transmitted using SSL alert protocol. This happens in the SSL handshake phase.

In the SSL data transfer phase the communication data is divided into fragments or blocks and each fragment or block is attached with Message Authentication Code (MAC) and record header. The above combination is then encrypted with the determined secret key obtained in handshake phase and transmitted to client or server as SSL packet. On receiving part the data is decrypted using the secret key to get the Message Authentication Code and this MAC is checked to determine the originality of the communication data.

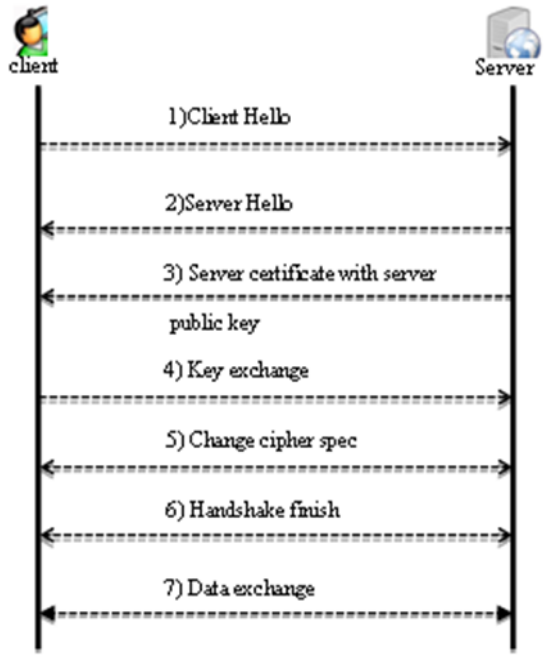

Fig. 3: SSL handshake protocol

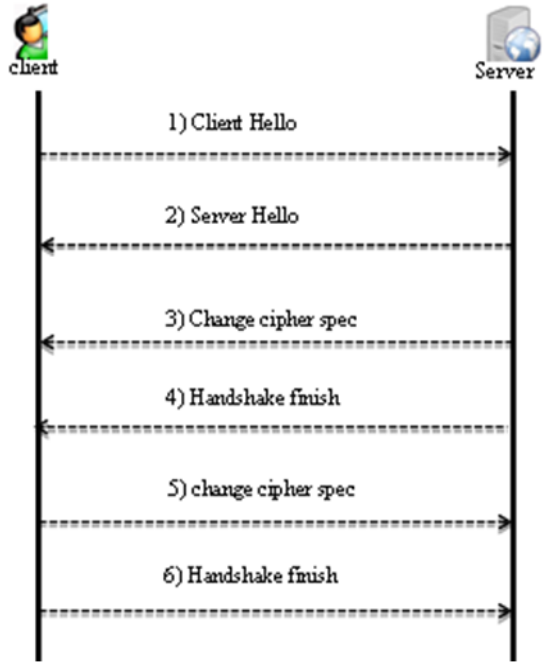

Fig. 4: SSL Session resumption protocol

The stepwise procedure of SSL handshake protocol is shown in fig.3. First the Hello message is send from client to server. This Hello message contains information like encryption algorithms, authentication algorithms and other cryptographic information supported by client. These cryptographic informations are also known as cipher spec or cipher suite. The server on receiving this message selects one of the cipher spec and replies to client a Hello message with this selected cipher spec. The server also sends a server certificate to client along with server's public key for server authentication. Client authentication is optional. Client authentication is done if server request for client certificate. The client on receiving server certificate checks for server authenticity. If the server is authorized the client sends its secret key encrypted by server's public key. The server decrypts this message to get the secret key. Thus now secret keys are determined and exchanged between client and server. This is the end of first phase of handshake protocol. End of first phase is denoted by handshake finish message; this finish message is exchanged between client and server. Next comes the second phase, the data transfer phase, in this phase the communication data is exchanged between client and server using the secret key as SSL packets.

The presence cloud is designed to decrease the buddy list search latency but by providing security to presence cloud the search latency increases. This increase in search latency is due to the computation of keys. And in SSL handshake protocol client makes several new connections with same server within short period of time this is also a reason for increase in search latency. If search latency increases the response time also increases. This is a drawback to designed server architecture. This drawback can be solved to smaller extent by using SSL session resumption protocol.

SSL Session resumption protocol is an abbreviated form of SSL handshake protocol. A slight change is done in the handshake phase of SSL handshake protocol. Here client and server maintain a cache list which contains list of session IDs and secret keys. The stepwise working of session resumption protocol is shown in fig. 4. Here a session ID is created and exchange between client and server along with Hello message. So any new connections made within a given time frame can simply refer to this session ID and can use the same secret key. This reduces the overhead of carrying out the full SSL handshake protocol each time within given time frame. The use of same session ID for any period increases security risk. Therefore new session ID and new secret key can be generated on request. This is done by the change cipher spec protocol. Content switching happens in session resumption protocol using same session ID i.e. all successive messages with the stored secret key should be exchanged between client and server which are done by change cipher spec protocol. This again increases latency which can be solved by SSL accelerators. The handshake finish protocol indicates the server or client is ready to enter the data transfer phase.

4. PERFORMANCE EVALUATION

The evaluation is done in terms of buddy list search latency and response time. The average buddy searching time of a mobile user is called search delay and total time it takes to respond to buddy list search request is called response time. If search latency is small then response time will also be small. The above server architecture is simulated using NS-2 simulator and following tables are obtained. Then taking the values from the tables corresponding graphs are drawn.

Table 1: Number of buddies versus Search Latency

Number of Buddies	Search Latency for System (ms)		
	Without security	With security, Without session resumption	With security, With session resumption
100	155	306	205
200	160	310	212
300	163	313	215
400	165	316	218
500	168	319	220

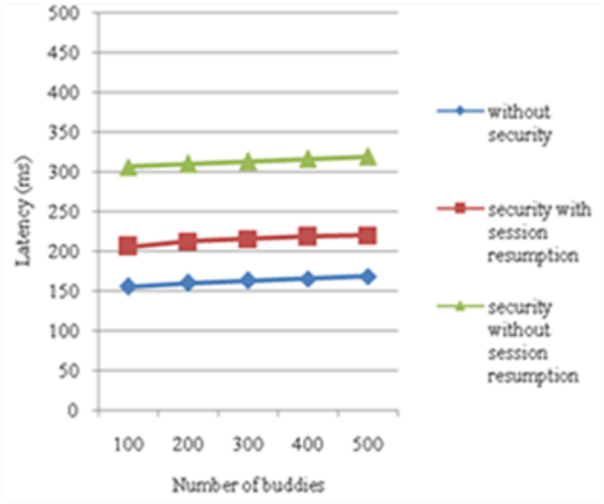

Fig. 5: Number of buddies versus search latency

Table 2: Number of presence servers versus Search Latency

Number Presence Servers	Search Latency for System (ms)		
	Without security	With security, Without session resumption	With security, With session resumption
100	125	325	200
200	150	360	250
300	190	400	290
400	225	450	356
500	280	480	390

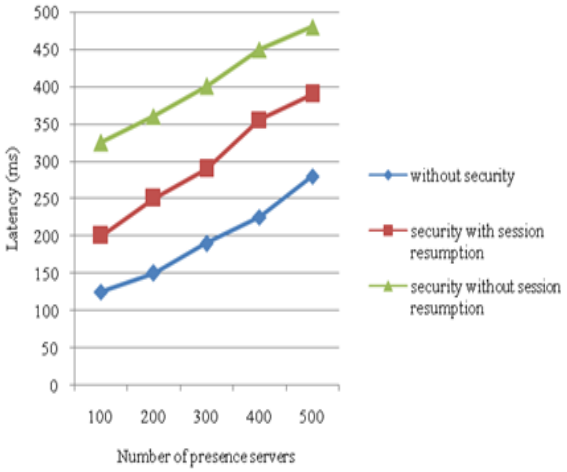

Fig. 6: Number of presence servers versus search latency

From fig.5 it is clear that when number of buddies increases there will not be much change in search latency because the search latency depends upon the number of presence servers in server overlay diameter. This small increase in latency is due to real time implementation problems. From fig.6 it is clear that when number of presence servers increases there will be increase in search latency also. This increase in latency is due to cryptographic procedures such as key computation, exchange etc. When search latency increases search response time also increases. For system without security search latency will be less when compared to the system with security. And again for system with security and with session resumption the search latency will be less when compared to the system with security and without session resumption.

5. CONCLUSION AND FUTURE WORK

Thus secured server architecture is proposed by using presence cloud and SSL protocols. Presence cloud is designed in such a way that it reduces network traffic of buddy search messages. With the help of one-hop caching strategy and directed buddy search, buddy searching is done easily and with few number of buddy search messages. Therefore search latency is reduced and response time is improved. Also scalability problem is solved for large number of mobile users. The search overhead is increased by providing security to presence cloud. This overhead is decreased to smaller extend by using SSL session resumption protocol.

As future scope, the search latency can be further decreased by using SSL accelerators or some other methods. The quality of service provided by this present architecture such as user satisfaction, search precise and search response time can also be improved.

6. REFERENCES

[1] R. B. Jennings, E. M. Nahum, Olshefski D.P, Saha, D, Zon-Yin Shae, Waters C. ”A study of internet instant messaging and chat protocols” IEEE Network, Page(s): 16 – 21, Volume: 20, 2006.

[2] Z. Xiao, L. Guo, J. Tracey, ”Understanding instant messaging traffic characteristics” Proc. of IEEE ICDCS, Page(s): 51, 2007.

[3] Chi-Jen Wu, Jan-Ming Ho, Ming-Syan Chen, "A scalable server architecture for mobile presence services in social network applications" IEEE Transactions on Mobile Computing, Page(s): 386 - 398, Volume: 12, 2013.

[4] Loreto S., Eriksson G.A., "Presence Network Agent: A Simple Way to Improve the Presence Service", IEEE Communications, Page(s): 75- 79, Volume: 46, 2008.

[5] Sheldon F.T., Weber John Mark, Seong-Moo Yoo, Pan, W.David "The Insecurity of Wireless Networks" IEEE Security and Privacy, Page(s): 54- 61, Volume: 10, 2012.

[6] Kuan Zhang, Xiaohui Liang, Xuemin Shen, Rongxing Lu "Exploiting multimedia services in mobile social networks from security and privacy perspectives" IEEE Communications Magazine, Page(s): 58 – 65, Volume: 52, 2014.

[7] Honggang Wang, Shaoen Wu, Min Chen, Wei Wang "Security protection between users and the mobile media cloud" IEEE Communications Magazine, Page(s): 73-79, Volume: 52, 2014

[8] Chou W "Inside SSL: the secure sockets layer protocol" IEEE IT Professional, Page(s): 47- 52, Volume: 4, 2002.

[9] Alfred C. Weaver "Secure Sockets Layer" IEEE Computer, Page(s): 88- 90, Volume: 39, 2006.

[10] Potlapally, N.R., Ravi S., Raghunathan A., Jha N.K. "A study of the energy consumption characteristics of cryptographic algorithms and security protocols" IEEE Transactions on Mobile Computing, Page(s): 128 – 143, Volume:5, 2006.