A framework for authenticating with an identity management server in mobile cloud computing

Mr. Mantripragada Satya Venu Gopalarao

Asst. Professor, Dept of M.C.A,

S.K.B.R P.G College,

Amalapuram, E.G.Dt., A.P, India.

msvgopalarao@gmail.com

Abstract

The purpose of this project is to design and implement a framework that uses a unique authentication method based on third-party identification, which is required to accurately identify and authenticate legitimate users to reduce the risk of confidential data disclosure against unauthorized persons in mobile cloud computing, customer center. Cloud computing is a service provided to users to store and process their data online instead of on a local device. Mobile cloud computing, an extension of cloud computing service, allows users to remotely access their data and services previously uploaded to the cloud anywhere and anytime as long as they have an internet connection on their personal mobile device. During the era of mobile cloud computing, security has always been a concern and a daunting challenge, even for seasoned security professionals. In addition, encryption schemes such as RSA and the Diffie-Hellman cryptosystem have proven useless against quantum computer attacks. Therefore, the homomorphic signature scheme is introduced into mobile cloud computing along with Identity Management (IDM) server to address this issue by applying an implicit authentication method to distinguish between real and non-real users, allowing the system to identify the clients can accurately authenticate . The details of the framework are explained further later in this document, where the user is authenticated using IDM as the medium and no password is used throughout the authentication process, allowing the client to be securely authenticated at the end of the process.

Index Terms: Mobile Cloud Computing security, Homomorphic encryption/authentication method, Homomorphic signature, Identity Management, Authentication Framework, Post Quantum Cryptography

I. Introduction

Mobile computing provides wireless communication services that can transfer

information to other devices through the format of voice, data, or in the form of a visual presentation such as image and video. On the other hand, cloud computing is a method of utilizing the function of a remote server that is not hosted locally. Through the internet connection, the user can access the server anytime anywhere [1] to store, manage, secure and process the information outside the device once the specific user is authenticated. Examples of the few well- known cloud providers are Google Drive, Amazon S3 and DropBox. This service brings a lot of convenience to the users as they don't have to worry about financial matters such as additional costs like most others like investing money to set up the infrastructure, checking the hardware and software, maintaining and expanding their own storage server of the Works are managed by the remote cloud service provider, with customers only having to worry about how much the resources are billed based on their usage. What is MCC? Mobile cloud computing is an emerging cloud service model that follows the trend of extending the cloud to the edge of networks. [2] MCC is flexible as it is compatible with various mobile operating system platforms such as Android, IOS and Windows. By integrating mobile devices with cloud computing, MCC also enables mobile users to access the cloud, where users can monitor their smart phone, using the cloud resource to handle some of the mobile computing tasks. Since then, the number of mobile phone owners has increased significantly compared to computer users. With the implementation of cloud computing technology in mobile devices, users can now remotely access and receive services from the cloud through their individual portable device. Even though this technology brings benefits to the users by allowing them convenient remote access to the cloud services via their mobile device, it is inevitable that the MCC cannot escape the cyber security challenges, especially regarding the user authentication aspect.

II. Related Work

Apart from the project proposed in this whitepaper, which focuses on MCC authentication, which eliminates the use of passwords, there are several existing solutions that are almost identical in the authentication method proposed in this whitepaper. Even though the approaches differ, they still share the basic concept of identification, which is to provide a framework that can authenticate and identify users while reducing the need for them to disclose their SPI. Eliminating the password during the authentication process greatly reduces the chances of user credentials being stolen, as there is not enough user credential information for the attacker to snoop on and use for authentication. Below are the related frameworks that can authenticate the client with minimal to no use of passwords:

A. PRIME Known as Privacy and Identity Management for Europe, is a middleware that manages and authenticates user data by protecting privacy using anonymous credentials. The Identity Mixer protocol is provided to allow users to disclose any login attributes obtained from a third party (IdP) without revealing any information. The problem with PRIME is that implementation requires both Client Agents and Cloud Service Providers (CSP), the implementation costs are high and would require large budgets with accurate accounting to ensure the availability of ongoing funding for the Service Provider.

B. Single Sign-On This is a session and user authentication service [3] that allows a single user to access multiple applications on their device using just a set of credentials [4]. This service eliminates the need to keep asking the user to enter their SPI in any other application residing within their mobile device as long as it is in the same session. On the other hand, there is a risk that the user's session can be hijacked by third party attackers through session injection, where the injected malicious codes overwrite the session, allowing the hijacker to take control of the session and thus achieve its victims' authenticated identity.

C. Kodekey eliminates the use of passwords and authenticates the client through a mobile application that identifies the user through biometric scanners. It also links users' biometric information to their phone number and PIN and can be integrated with a web-based API. If the system detects that there is sign-in activity that uses phone number and PIN instead of a biometric scanner, it prompts the user on their phone to verify the legitimacy of the sign-in activity by requiring the user to perform a biometric scan.

D. LaunchKey is a mobile application that allows users to manage the entire authentication process on their devices. It is a multi-factor authentication system that allows users to authenticate themselves using various methods such as biometrics, retina, Bluetooth proximity, etc. Apart from that, the LaunchKey engine does not contain any sensitive user authentication information and is only contained in the user's own device and the SPI is also not sent out during the authentication process, which ensures the confidentiality of the user SPI.

E. Homomorphic Authentication Encryption (HAE) HAE ensures that both the privacy and the authenticity of the data are secured at the same time. It is a symmetric key cryptography whose functions can be evaluated by the public through appropriate ciphertexts. [5] Basically, HAE is a homomorphic version of indistinguishability under selected plaintext attack (IDN-CPA), which is a security guessing game consisting of two parties who are challengers and opponents. The opponent first generates 2 different messages of similar length to allow the challenger to encrypt one

of them. The challenger can randomly choose any of the messages to be encrypted. After that, the opponent tries to guess which of the messages was encrypted by the challenger. The identical message length is used to prevent the opponent from comparing the message by their length difference and easily win the game.

III. Existing System

The most common authentication process today is username and password authentication, a process in which the server prompts the client for a name and password. Once entered, the server queries the database to find the matching username and then compares the password. Once all required inputs have been proven to match, the client is authenticated. This is a simple straightforward authentication process, but the SPI entered by the user can easily be stolen in any 3 part of the authentication process. During the first part of the process where the user enters their SPI into the device, their credentials can easily be stolen by logging tools like keyloggers that record the entire keystroke typed. During data transfer between devices, the second part of the authentication process, the information being sent can be easily stolen through the use of man-in-the-middle eavesdropping tools such as Ether Cap of the Kali Linux operating system. Finally, the server that receives the user SPI can also be malicious. The server can be spoofed, whereby attackers can create a fake server with the same domain name and then launch a denial of service attack to disable the real server, allowing them to impersonate and get their hands on user SPI. Or maybe the real server itself can't be trusted, providing services to the clients while collecting their SPI without the client's knowledge.

Disadvantages:

Encryption method such as RSA and Diffie Hellman cryptosystem, proven to be useless, against the attacks launched by quantum computer. Authentication is not safe

IV. Proposed System

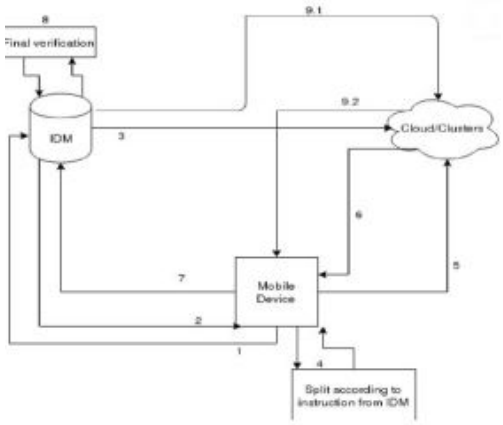
This proposed system allocates an intermediary to authenticate and represent its customer to request cloud services without the need to use the customer's own SPI, thus reducing the risk of identity theft since the customer's identity exchanged only between the IDM server and the client device. The homomorphic signature scheme is introduced in mobile cloud computing along with the Identity Management (IDM) server to solve this problem by applying an implicit authentication method to distinguish between real and non-real users, allowing the system to identify the clients can authenticate accurately.

Advantages:

Authentication is secure. This system will allow user to be authenticated in a secure

manner without the need of revealing their SPI, providing a better safeguard for the privacy and confidentiality of the user's personal information as well as their identity.

V. Process Model



IDM will also generate the new identity same as what client did for final verification purpose as well. Once the client is verified, the IDM will then represent the client to perform service request towards the cloud using the mapped dummy name along with a designated client's address.. Once the cloud verified the request, the service will be provided towards the address as instructed by the IDM. The client that listens to the designated address will then receive the service from cloud, concluding the whole authentication process. The flow diagram of the authentication process:

Authentication process:

1. Mobile device sends authentication request along with its identity (each device have their unique identity) and address to IDM.

2. IDM will then verify the identity of the mobile device and create an instruction that randomly splits out the certain part of the identity which will be sent along with the address of the assigned cloud (for encryption purpose) to the mobile device.

3. The IDM then sends out the same partial value of the identity to the Cloud (cloud won't know what it is), instructing the cloud to only encrypts the matching partial identity that will be sent by the mobile device later.

4. After the mobile device receives the address of the cloud and the split instructions, the mobile device will split out the partial identity according to the instructions received.

5. The partial identity is then being sent to cloud to be encrypted.

6. The cloud will compare the received partial identity value with the ones received from the IDM, once it is matching, it will start the homomorphic encryption process. Once it is completed, the cloud will send the encrypted partial identity back to mobile device and another one to IDM.

7. The encrypted partial identity is then sent back to mobile device to be combined with the other original part of the identity (as header) before being sent to IDM for final verification.

8. IDM will then perform final verification by first comparing the identity header with the one initially received in step 1(IDM knows the

specific header for that particular identity as it was the one initially had the identity values split) and then compare the homomorphic side of the identity.

9. Once the final verification is successful, the IDM will send the service request along with mapped dummy name and the mobile device address to the cloud server (can be different cloud server based on user's service request), allowing mobile device to receive service from the cloud.

VI. Homomorphic Encryption

Possibly, encrypting the plaintext proves too easy and has a higher chance of leaking thecontents of the plaintext, thereby compromising the confidentiality of SPI users. Homomorphic encryption allows further encryption computations to be performed on the originally encrypted data, further encrypting the received ciphertext to forge its hardness. Homomorphic encryption is used on cloud computing platforms because it requires a large amount of resources to perform sophisticated calculations on the encrypted data. A fully homomorphic cipher developed by [9]deploying lattice-based cryptography, which is a candidate for post-quantum cryptography that can be used to mitigate attacks launched by quantum computers. Attacks from quantum computers are a massive tidbit for modern security. Public- key cryptosystems that are implemented, such as RSA and the Diffie-Hellman algorithm, can be easily removed by a quantum algorithm developed by PeterShor, known as Shor's algorithm. Shor's algorithm can find out the prime factors with the given integer by finding its functional period, which allows it to easily break RSA since it generates a public/private key pair for the encryption/decryption process, respectively. Lattice-based cryptography in a fully homomorphic encryption scheme is difficult to crack due to the nature of lattice hardness. The shortest vector problem (SVP) is essentially related to the lattice-based computational problem, where it asks for an output of a non-zero lattice vector whose norm is greater than the shortest non-zero lattice of a given length, given by a approximation factor is limited. The hardness (NP-hard) or, generally speaking, at least as hard as the hardest problems in nondeterministic polynomial time that have been shown to be highly quantum-resistant to further forge security during the authentication process. The cloud serves as a platform that helps in further encrypting the partial identity sent from the client device by using homomorphic encryption so that even the client's identity is properly protected from eavesdropping. Furthermore, the cloud that received the encrypted value does not know the original value, but can determine it and validate its identity due to the homomorphic nature of the ciphertext.

VII. CONCLUSION

It is undeniable that the confidentiality of the mobile users' SPI is always at risk when used during the authentication process since they are connected to the network where they would send their information over the internet to the cloud to be verified and authenticated will. In this way, the mobile user has the potential to become a victim of a man-in-the-middle attack, where the packets sent from their device pass through the unauthorized eavesdropping filter before actually reaching the authenticator. This can leave their information in the hands of the attacker, putting both their privacy and identity at risk. Thus, the attackers grant unauthorized access to their personal accounts through impersonation since the SPI sent is the only information/evidence provided that can be used to identify the legitimacy of the user. So, instead of developing a scheme that focuses solely on improving the security of the system by strengthening the encryption algorithm, a framework for multiple authentication purposes should be proposed, developed, tested, and implemented in the actual environment. The multiple authentication process proposed in this white paper eliminates the use of passwords or other important credentials and instead uses an identity generated jointly by the IDM and client. This method allows the user to be authenticated in a secure way without the need to reveal their SPI, providing better protection for privacy and confidentiality.

REFERENCES

[1] Q. F. Hassan, “Demystifying Cloud computing”. CrossTalk(2011), 16- 21, 2011.

[2] D. Huang, Z. Zhou, L. Xu, T. Xing, and Y. Zhong,, “Secure data processing framework for mobilecloud computing”, in: Proc. IEEE INFOCOM Workshop on Cloud Computing, INFOCOM ’11, Shanghai, China, June 2011.

[3] J. Hursti, “Single Sign-On.” Retrieved December 4, 2016, from http://www.tml.tkk.fi/Opinnot/Tik-

110.501/1997/single_sign-on.html, 1997.

[4] H. Mohammed, M. F.Suliman, V. Ponnusamy, B. Y. Ooi, A. Robithoh, and S.Y .Liew,.” A Coherent Authentication framework for Mobile Computing Based on Homomorphic Signature and Implicit Authentication.” Proceedings of the 6th International Conference on Computing and Informatics, ICOCI 2017, 25-27April, 2017 Kuala Lumpur. University Utara Malaysia.

[5] C. Joo and A. Yun, "Homomorphic Authenticated Encryption Secure Against Chosen-Ciphertext Attack", in ASIACRYPT: International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C, 2014.