How to cite this paper:

Hamdun Mohammed, Suliman Mohamed Fati, Vasaki Ponnusamy, Ooi Boon Yaik, Robithoh Annur, & Liew Soung Yue. (2017). A coherent authentication framework for mobile computing based on homomorphic signature and implicit authentication in Zulikha, J. & N. H. Zakaria (Eds.), Proceedings of the 6th International Conference of Computing & Informatics (pp 426-434). Sintok: School of Computing.

A COHERENT AUTHENTICATION FRAMEWORK FOR MOBILE COMPUTING BASED ON HOMOMORPHIC SIGNATURE AND IMPLICIT AUTHENTICATION

Hamdun Mohammed[1], Suliman Mohamed Fati[2], Vasaki Ponnusamy[3], Ooi Boon Yaik[4], Robithoh Annur[5], and Liew Soung Yue[6]

[1,3,4,5,6]Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman (UTAR), Kampar, Malaysia.
[2]Faculty of Information Technology-Math & Science, Inti Inti Internatioanl University, Persiaran Perdana BBN, Putra Nilai, 71800 Nilai, Negeri Sembilan
[1]saidhamdun@1utar.my, [2]smfati@yahoo.com, [3]vasaki.ponnusamy@gmail.com, [4]ooiby@utar.edu.my, [5]robitoh@utar.edu.my, [6]syliew@utar.edu.my

ABSTRACT. Mobile cloud computing is an extension of cloud computing that allow the users to access the cloud service via their mobile devices. Although mobile cloud computing is convenient and easy to use, the security challenges are increasing significantly. One of the major issues is unauthorized access. Identity Management enables to tackle this issue by protecting the identity of users and controlling access to resources. Although there are several IDM frameworks in place, they are vulnerable to attacks like timing attacks in OAuth, malicious code attack in OpenID and huge amount of information leakage when user's identity is compromised in Single Sign-On. Our proposed framework implicitly authenticates a user based on user's typing behavior. The authentication information is encrypted into homomorphic signature before being sent to IDM server and tokens are used to authorize users to access the cloud resources. Advantages of our proposed framework are: user's identity protection and prevention from unauthorized access.

Keywords: Mobile Cloud Computing security, Homomorphic signature, implicit authentication, Identity Management, Authentication framework.

INTRODUCTION

Cloud computing is a network based technology that comprises of a group of integrated and networked hardware, software and internet infrastructure. It provides computations and storage services to computers and other devices on demand. Now, it is considered as a key innovative technology that provides computing resources to users is a similar way as utility-based services. Majority of internet users exploit the cloud computing in processing and/or storing their data remotely over cloud storage devices anywhere and anytime (Hassan, 2011). On the other hand, mobile devices have become most popular technologies. It has allowed users to shift from PC usage and made ubiquitous computing much easier. According to (Khalil, Khreishah, & Azeem, 2014), the mobile growth statistics showed that more than 90% of the people around the world own mobile devices (mobile phones and tablets), more than 50% of mobile users depend mainly on their mobile devices to access the internet, and every year ¾ of tablet users use their tablets to access the online procurement websites. Moreover,

many people utilize the mobile devices to access the cloud services like Dropbox and Google drive. For that, we can argue that mobile cloud computing becomes one of the key technologies that change the world face and make the daily life easier.

Mobile cloud computing helps the mobile device to store and process data outside the device. All the intensive computing can be performed in the cloud. This reduces the burden of the mobile device resources. To do so, the mobile device must be connected to the internet which introduces the security challenges together with accessing of mobile devices anywhere (Dharmale & Ramteke, 2015). According to (Khan, Kiah, Khan, & Madani, 2013), there are many challenges in MCC should be studied and addressed. These challenges includes guaranteeing user privacy and the provision of mobile application security that uses cloud resources. To provide a secure MCC environment, service providers need to address issues pertaining to data security, network security, data locality, data integrity, web application security, data segregation, data access, authentication, authorization, data confidentiality, data breach issues, and various other factors. Thus, the focus of this paper is the security challenges, includes access control and identity management. According to (Sujithra & Padmavathi, 2012), the security issues in mobile devices like mobile malicious code, malware injections, and credential thefts are increasing significantly. Mobile devices are vulnerable when accessing cloud without considering security issues and it's very risky for both users and cloud providers. Therefore, one of the major issues in mobile computing field is unauthorized access. Evidently, the mobile users are more vulnerable than the other users to the unauthorized access due to many reasons (Khalil, Khreishah, & Azeem, 2014):

- The ease of hacking, capturing, and breaking down the wireless networks, which is used by mobile devices, compared with the wired networks.
- The ease of capturing or accessing the sensitive data by a third party in case of losing, stealing, or forgetting the mobile devices in anyplace.
- Most of us store the credential and sensitive data (Credit Cards, Passwords, Personal Identifiable Information) in an indecorously secure manner and therefore, these sensitive data become accessible and easy to collect.

Accordingly, authenticating the mobile users based on the credentials is not secure at all. Moreover, transferring the sensitive information like users' credential and/or credit cards information to third party for the authentication purpose is risky even if such information have been encrypted. Therefore, the need for a coherent authentication framework to overcome such security issues remains present and unsolved. This paper investigates the authentication issue for mobile cloud computing and proposes a coherent framework based on **implicit authentication** and **homomorphic signature** to secure protect the mobile users and cloud service providers from unauthorized access in mobile cloud computing environment..

BACKGROUND AND LITERATURE REVIEW

Identity Management is a security discipline that deals with identifying individuals in a system and controlling their access to resources within that system by associating user rights and restrictions with the established identity. In short, it enables the right individuals to access the right resources at the right time and for the right reasons (Gartner, n.d.). Identity management comprises of how users gain an identity, identity protection and technologies involved in supporting the protection. Furthermore, access control is a security technique used to control who or what can view or use resources in a computing environment. It's a restriction of a location or usage of resources (Sandhu & Samati, 1994). Access control systems conduct authorization, identification, authentication, and access confirmation with the help of login credentials like passwords, personal information number (PIN), biometric scans, etc. As the

focus of this paper is on identity management, we will highlight some of identity management techniques that are already in place. We will try to explain the pros and cons of such IDM techniques.

- **OAuth -** It's a service that provides internet users to authorize third party applications or websites to access their information on other websites without exposing the passwords. They share information about their accounts with the third party applications or websites. It is designed mainly to work with Hypertext Transfer Protocol (HTTP) (OAuth, 2016). It uses tokens issued by authorization server to access the protected resources hosted by resource server. The tokens are issued to third party applications or websites after the approval from resource owner. The tokens has a timestamp with expiry time. OAuth can be attacked in four ways: lack of data confidentiality and server trust, insecure storage of secrets, implementations with flawed session managements and session fixation attack (Kiani, 2011). Furthermore, it is vulnerable to timing attacks. When does not provide any mechanism for data security when the mobile device is stolen except users using key lock on their mobile devices.
- **OpenID -** It's an open standard service that allows users to sign-in into different websites with a single username and password. User has greater control of information shared with websites visited. The password is only given to identity provider who confirms your identity to the websites you visit. No one can disclose your password other than identity provider (OpenID, 2016). It facilitates login in different sites via Single Sign-on. However, it has many vulnerabilities to malicious code attack. The code is injected to the server that uses OpenID which deceives the user by redirecting to different identity authentication page that requests for credentials. It is also vulnerable to timing attacks especially when there exist a combination of OAuth with OpenID. This is considered as lethal to user's private data (Khalil, Khreishah, & Azeem, 2014).
- **Single Sign-on -** Session and user authentication service that allows a user to access multiple applications using a set of login credentials. It authenticates the user to use all the applications given right to and eliminates further prompts when switching applications as long it's during the same session (What is SSO, n.d.). It uses protocols like Kerberos. Kerberos authenticates requests between trusted hosts across an untrusted network like internet. It's built in all major operating system. Kerberos consists of client, Kerberos server, Ticket Granting server and Application server. Kerberos server acts like a broker which centrally authenticates users and granting them electronic identity as per their given credentials. The authentication process in Kerberos starts with the client authenticating itself to the Kerberos server and receiving the Ticket Granting Ticket. The client submits the Ticket Granting Ticket to the Ticket Granting server to receive Server Ticket which later use it to request service from an Application Server (Hursti, 1997). Single Sign-on has vulnerabilities that can lead to serious attacks especially when user's identity has been compromised. Illegitimate user can successfully sign-in once and never be verified again. Therefore, leading to huge amount of information leakage (Khalil, Khreishah, & Azeem, 2014).

Furthermore, there are several authentication frameworks for MCC proposed. The authors in (Alizadeh, Abolfazli, Zamani, & Baharun, 2016) conducted a survey that categorizes authentication methods based identity and context on both cloud and user side. On the cloud side, most of the authentication process is performed in the cloud server.

Multifactor-based authentication method, authenticates mobile users based on: ID/password, international mobile equipment identity (IMEI), international mobile subscriber identity (IMSI), voice and face recognition. It encrypts the IMEI and IMSI in case of mobile

loss or theft which are used to protect the user. But the bio-information is unencrypted which raises the privacy issue. There is no mutual authentication between the mobile user and MCC server and it focuses more on performance and time computation and neglect security issues (Alizadeh, Abolfazli, Zamani, & Baharun, 2016).

Message digest authentication is an authentication method that uses message digest (MD) to protect mobile user from various potential security attacks. It considers mutual authentication as important for the method to be effective. Mutual authentication ensures secure authentication and is done via two phases. First phase, mobile device sends authentication request message using hashed ID/password. Cloud server verifies the authenticity of the mobile device by matching the message digests sent during the authentication request. If it matches, then the cloud server initiates authentication by sending its encrypted digital signature to the mobile device. Mobile device then checks the authenticity of the cloud server by matching the decrypted cloud server MD with mobile device MD. If it matches, then the cloud server is authenticated (Alizadeh, et al., 2016).

Cloud-ready biometric uses user handwriting as an authentication factor to access cloud server. The user inputs password manually using touch screen and sends the encrypted image to the cloud server. The cloud server then decrypts the image and starts to check the validity of the handwriting and the password itself. The fingerprint is captured by a mobile device and sent to the cloud server as a plain text to perform fingerprint recognition. If it is recognized, then the user is accepted (Alizadeh, Abolfazli, Zamani, & Baharun, 2016).

Fuzzy vault, digital signature and zero-knowledge combination (FDZ) provides entity authentication if the mobile user wants to connect to the cloud server. To authenticate mobile user, fuzzy password system is represented where the mobile user needs to select correct images among the seven images provided. If the user selects five correct images, then the user is granted access to the server resources. Diffie-Hellman key exchange is used to encrypt and secure the channel between the mobile user and the cloud server. Digital signatures are used for verification in an entity authentication protocol (Alizadeh, Abolfazli, Zamani, & Baharun, 2016).

QR code-based protocol uses QR code which is a 2-dimensional/form matrix. It uses fuzzy password system just like FDZ to authenticate mobile user. The image, ID and password of the mobile user are converted to QR code. The user information is converted into three different versions of QR code and stores each QR code in a circulation loop method. It is then used as authentication certificate via its strength such as high data integration, error correction capability and compressibility. It prevents impersonation attack and Diffie-Hellman key exchange generates secure channel randomly (Alizadeh, Abolfazli, Zamani, & Baharun, 2016).

SeDiCi 2.0 protocol provides mutual authentication by allowing users not to disclose their passwords at each of the websites they visit. It's another form of zero knowledge proof. The user runs an authentication on the web browser to prove that given domain is under their control which acts as Third Trusted Party (TTP). User can login to the system if the name of the service is on the trusted list. The protocol implements the plugin-based to allow the application to bypass built-in browser policy (Alizadeh, Abolfazli, Zamani, & Baharun, 2016).

NemoAuth is an authentication method based on mnemonic multimodal approach. It's a combination of dynamic knowledge and biometric based approaches. It predefines and trains user's signature profile. The signature comprise of a set of mnemonic and atomic actions. It utilizes available mobile device sensors to measure and extract biometric features of a mobile device user. Atomic actions varies depending on the type of mobile device sensors available. Mnemonic image helps to simplify the remembering of password for users. Users can select a

signature profile to use a different authentication method on a different period (Alizadeh, Abolfazli, Zamani, & Baharun, 2016).

According to (Khan, Kiah, Khan, & Madani, 2013), the proposed authentication framework implements TrustCube infrastructure and implicit authentication method that translates user's behavior into scores. The probabilistic authentication scores are computed using statistical model and then assigned to user device based on user's behavior. Policies are implemented based on client device request which helps in the authentication process in the authentication engine.

Even though the above frameworks have their strengths but they also have their weaknesses. The authentication methods proposed are not mainly focusing on the protecting the personal information of mobile user. The methods that uses biometric approach mostly do not encrypt the data when sending to the cloud server. Therefore, the user information is exposed. Other methods uses probability values to identity the owner of the mobile device which is not accurate since the values varies from time to time. Authentication methods that allows many algorithms to run on mobile device does not abide the rules of MCC in terms of offloading intensive computation. Furthermore, using several authentication factors hinders the efficiency of mobile computing. Most of the authentication frameworks are not lightweight. Therefore, it becomes difficult to implement such authentication methods. Mutual authentication tends to be an important feature in authentication mechanism. Fuzzy password system used in some authentication methods contains limited password space. Thus, it does not provide secure authentication. Lack of tokens usage is some of the methods leads to complex methods and high computation power. Therefore, the proposed system will solve the weakness of the discussed authentication frameworks.

Khalil (Khalil, Khreishah, & Azeem, 2014) has an initiative called consolidated identity management system that alleviates the problem of illegitimacy of user access. As known that if the mobile devices stolen, then the stored credential can be used to sign in without checking the legitimacy of the connection. Thus, the idea in the consolidated identity architecture is to break up the identity string into two parts: the first part is called session commit value, while the second one is called encrypted session commit value. The client initiates the two parts and sends the session commit value directly to the server. The second part, encrypted session value, is sent by the client to IDM who will send it to the server. Once the server receives the two parts successfully, it will approve the connection.

Another direction in identity management systems is to generate dynamic identities for the users to access the cloud resources. For example, Leandro (Leandro, et al, 2012) uses Shibboleth as an access control without the need for a trusted third party. Although, this direction gives promising results in the authorization part, the authentication is still a problematic. According to (Khalil, Khreishah, & Azeem, 2014), these techniques are unable to cope with the mobility challenges. Moreover, Xiao (Xiao, 2013) argued that the current security techniques are insufficient in the mobile cloud computing field. This is due to that in case of the credential of a user has been stolen, this means that the whole cloud is at risk

PROPOSED FRAMEWORK

In this section, we will give details about the proposed coherent IDM framework. The coherent framework includes modules on both the mobile user side, IDM side, and cloud server side. Figure 1 depicts the proposed framework whereby the implicit authentication based on user behavior will be implemented in the mobile user side. The implicit authentication module gathers the information on user's behavior to predict the legitimacy of the user. Based on the implicit authentication module, the mobile device creates the authentication information to be sent to homomorphic signature module. In this case, the authentication information is a

record of legitimate user and mobile device description rather than the traditional username and password. The homomorphic signature module, in the mobile device, encrypts this information to generate the homomorphic signature that to be sent to IDM server. The benefit of homomorphic signature is to encrypt the data so nobody can disclose the data except the user him/herself. IDM server manipulate and verify the user's identity without decrypting the homomorphic signature sent by the user. This is one of the strength point of our proposed framework.

Mobile user sends the homomorphic signature including the requested service details to the IDM server to be verified using a list of full signature of clients in database. If it matches, IDM generates and distributes token into two parts. The first token will be distributed to the mobile device and the second part is sent to the cloud server. The mobile device will then send the received token to the cloud server. Cloud server will merge the two tokens and verify with the IDM. If it matches, then the result will be sent to the cloud server and the mobile device/user is granted access to cloud server services. To explain this well, assume that there is a VIP event. The organizer invite special people to attend this event. For more security, a list of invitees is placed in the reception counter at the venue gate. In this case, the invitee should show the invitation card to the recipient who will also check the invitee list. The invitee can't enter the venue unless he/she has invitation card and his/her name is in the list. Otherwise, he/she can't enter. Similarly, our framework will verify the user's identity based on the homomorphic signature. IDM server will send two verification tokens, one to the user (invitation card) and one to server (list of invitees). The cloud server should compare the verification token sent by the user (invitation card) with that sent by IDM server (invitee list). If the two tokens are matched, the user grant access to the cloud services. Otherwise, the access is considered unauthorized and will be blocked. Homomorphic signature allows secure and encrypted authentication information to be sent to IDM server. Furthermore, it allows only partial information to be sent in binaries. Therefore, full authentication information cannot be accessed by an attacker.

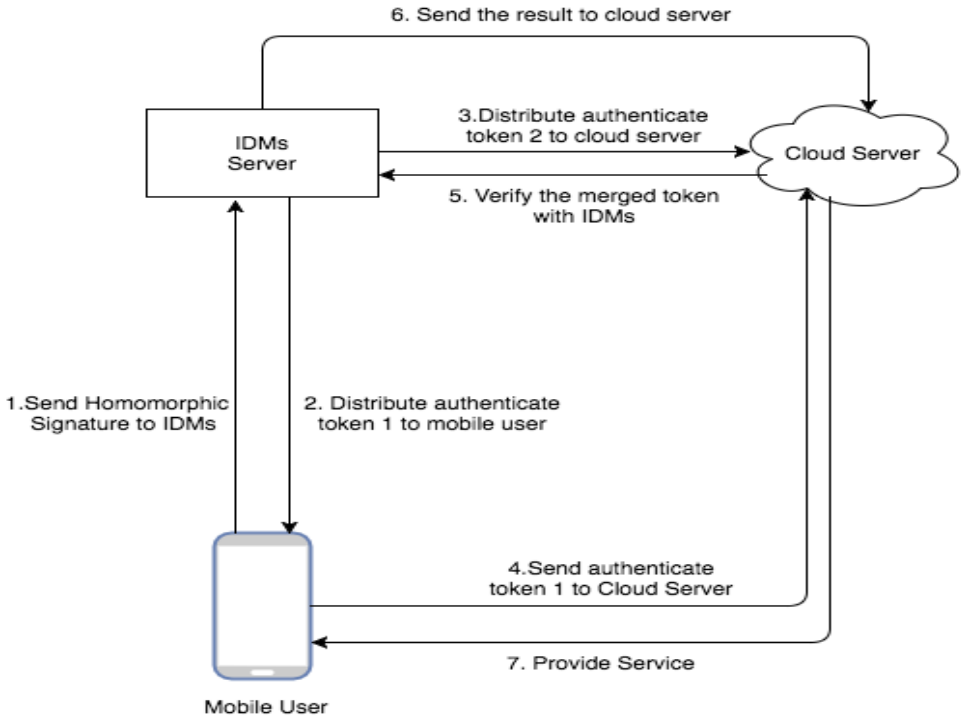

Figure 1. Proposed Framework.

DISCUSSION ON THE POTENTIAL CHALLENGES

The challenges of the proposed framework are discussed in this section. They are also the main and most important parts of the framework.

Predict the user's behavior

In the literature to authenticate a user implicitly based on user's behavior, monitoring of user's behavior is conducted. The daily activities of the user are recorded. Data collected are: location, accelerometer measurements, motion, communication, application usage, application installation and smart device usage (Fan, Li, Wu, & Sun, 2014). Based on (Shi, Niu, Jakobsson, & Chow, 2011) framework, the data collected is considered as past behavior. It is used to learn the user model from the past behavior. With the recent user behavior, the learnt user model can then compute the probability/authentication score that the mobile device is in the hands of a legitimate user. Authentication score makes authentication decision using threshold to accept or reject the user. If the authentication score is below the threshold, then the user must explicitly authenticate by entering password or passcode. In the proposed framework, we will predict the mobile device users based on the user's typing behavior. We will build a prediction model using machine learning classifier based on Support Vector Machine (SVM). It will provide continuous authentication. Data is collected using accelerometer, gyroscope and orientation sensors under certain time constraints. The data/captured signals will then be normalized using mean, linear and spline normalization to remove constant values. With the help of time-based feature extraction, the features becomes user profile and data point is computed in every T seconds when the user is typing. These data points will help to find a hyperplane that optimally classifies between authorized and unauthorized classes (Gascon, Uellenbeck, Wolf, & Rieck, 2014).

Homomorphic signature

Homomorphic signature is a cryptographic primitive where each of signatures is computed on each message in message space. It contains a complex process of polynomial-time algorithms. Based on new homomorphic signature proposed (Wang, Sun, & Chen, 2014), user profile is encrypted to a homomorphic signature. The partial signature will be vector of binary numbers that is sent to the IDM server to verify. To generate a homomorphic signature, it should compute the following algorithms:

- **Setup** (1^{λ}, n): Takes as input a security parameter λ and a maximum authentication features of size n, *o*utputs a public key *pk* and a secret key *sk*.
- **Sign**(sk,v_i):Takes a secret key *sk*, authentication feature v_i, outputs a signature $\tilde{\sigma}_i=(\sigma_i,s_i,t)$,where s_i denotes a random number and t denotes a period of validity.
- **Verify** (*pk*, *score*, $\tilde{\sigma}$): Takes a public key *pk*, authentication information $\subset \{v_1 ... vn\}$, and a signature $\tilde{\sigma}= (\sigma, s, t)$, outputs either 0 (reject) or 1 (accept).
- **Eval**(pk,(c_1,...,c_n),$\tilde{\sigma}$):Takes a public key *pk*, a vector (c_1,...,c_n) corresponding to authentication information $\subset$ **v** (if *vi* ∈ *authentication information* then *ci* = 1, otherwise *ci* = 0), and a row of signatures (full signature) $\tilde{\sigma} = (\tilde{\sigma}_1,...,\tilde{\sigma}_n)$, outputs a partial signature $\tilde{\sigma}$ on *authentication information.*

Token generation and matching

IDM accepts the homomorphic signature from mobile device and verifies using a list of full signatures. If the homomorphic signature is valid, then the IDM server generates an authentication token and splits into two tokens. Merging of the two tokens is done in the cloud server which is later used to match with tokens available in the IDM server (Ahmad, Hassan, & Aziz, 2014).

CONCLUSION

This paper presents proposed framework that works in mobile cloud computing environment. The framework comprises of mobile device, IDM server and cloud server. Mobile device monitors the behavior of the user which is used as login credentials after being encrypted (Homomorphic signature). The signature is sent to the IDM server to verify. If it is successful, then authentication token is generated and sent to the mobile device and cloud server. Then the cloud server matches the two tokens. If it matches, then user has an authorized access to the cloud server. With implicit authentication as a primary method of authentication, the proposed framework can protect the identity of the mobile user from adversaries and also resolve unauthorized access to the cloud server.

ACKNOWLEDGMENTS

This research is supported in part by UTARRF Fund IPSR/RMC/UTARRF/2015-C2/S07 from Universiti Tunku Abdul Rahman, Malaysia.

REFERENCES

Ahmad, A., Hassan, M. M., & Aziz, A. (2014). A Multi-Token Authorization Strategy for Secure Mobile Cloud Computing. *International Conference on Mobile Cloud Computing, Services, and Engineering*, 136-141.

Alizadeh, M., Abolfazli, S., Zamani, M., & Baharun, S. (2016). Authentication in mobile cloud computing: A survey. *Journal of Network and Computer Applications, 62*, 59-80.

D. Hardt, E. (2012). *The OAuth 2.0 Authorization Framework*. Retrieved December 04, 2016, from https://tools.ietf.org/html/rfc6749#section-1.2

Dharmale, P. N., & Ramteke, P. L. (2015). Mobile Cloud Computing. *International Journal of Science and Research, 4*(1), 2072-2075.

Fan, Y., Li, H., Wu, Q., & Sun, W. (2014). Mining and Predicting Smart Device User Behavior. *Network Research Workshop, 38*, 65-77.

Gartner. (n.d.). *Identity and Access Management (IAM)*. Retrieved December 04, 2016, from http://blogs.gartner.com/it-glossary/identity-and-access-management-iam/

Gascon, H., Uellenbeck, S., Wolf, h., & Rieck, K. (2014). Continuous Authentication on Mobile Devices by Analysis of Typing Motion Behavior. 1-12.

Hassan, Q. F. (2011). Demystifying Cloud computing. *CrossTalk*(2011), 16-21.

Hursti, J. (1997). *Single Sign-On*. Retrieved December 4, 2016, from http://www.tml.tkk.fi/Opinnot/Tik-110.501/1997/single_sign-on.html

Khalil, I., Khreishah, A., & Azeem, M. (2014). Consolidated Identity Management System for secure mobile. *Computer Networks, I*(65), 99-110.

Khan, A. N., Kiah, M. M., Khan, S. U., & Madani, S. A. (2013). Towards secure mobile cloud computing: A survey. *Future Generation Computer Systems, I*(29), 1278-1299.

Khana, A. N., & M.L. Mat Kiahamee U. Khanb, S. A. (n.d.).

Kiani, K. (n.d.). *Application*. Retrieved December 04, 2016, from https://www.sans.org/reading-room/whitepapers/application/attacksoauth-secure-oauth-implementation-33644

Leandro, M. A., Nascimento, T. J., dos Santos, D. R., Westphall, C. M., & Westphall, C. B. (2012). Multi-tenancy authorization system with federated identity for cloud-based environments using shibboleth. In *Proceedings of the Eleventh International Conference on Networks* (pp. 88-93).

OAuth. (n.d.). *OAuth*. Retrieved December 04, 2016, from https://oauth.net/

OpenID. (2016). *What is OpenID*. Retrieved December 04, 2016, from http://openid.net/get-an-openid/what-is-openid/

Rouse, M. (n.d.). *Identity management (ID management)*. Retrieved December 04, 2016, from http://searchsecurity.techtarget.com/definition/identity-management-ID-management

Sandhu, R. S., & Samati, P. (1994). Access control: Principles and Practices. *IEEE Communications Magazine*, *32*(9), 40-48.

Shi, E., Niu, Y., Jakobsson, M., & Chow, R. (2011). Implicit Authentication through Learning User Behavior. *Information security*, 99-113.

Sujithra, M., & Padmavathi, G. (2012). Mobile Device Security:A Survey on Mobile Device Threats, Vulnerabilities and their Defensive Mechanism. *International Journal of Computer Applications*, *56*(14), 24-29.

Wang, Z., Sun, G., & Chen, D. (2014). A new definition of homomorphic signature for identity management in mobile cloud computing. *Journal of Computer and System Sciences*, *I*(80), 546-553.

What is SSO. (n.d.). *What is SSO (Single Sign On)?* Retrieved December 04, 2016, from https://auth0.com/docs/sso

Xiao, Z., & Xiao, Y. (2013). Security and privacy in cloud computing. *IEEE Communications Surveys & Tutorials*, *15*(2), 843-859.