International Journal For Technological Research In Engineering
Volume 4, Issue 10, June-2017 *ISSN (Online): 2347 - 4718*

SECURE AND STRONG MOBILE AUTHENTICATION

Prasanna B T[1], Meghana B Ramesh[2]
[1]Assistant Professor, [2]M.Tech 2nd Semester
Department of Computer Science & Engineering, Sri Jayachamarajendra College of Engineering(SJCE),
JSS S&T University Campus, Mysore, Karnataka, India

ABSTRACT: Mobile cloud computing has dual benefits that include cloud computing and mobile computing. In mobile cloud computing data storage and data processing take place outside the mobile device. As a result, there is a high chance of security attack. The security even becomes more critical when mobile phone is initiating the handoff during handover management process. Thus, the attacker may easily get access to our sensitive data. Due to this the malicious user may see or modify our data. In order to overcome this problem, we need to store our data in clouds in a secured manner. So we propose a secure and strong authentication (SSA) process that stores the key at different cloud servers. This process provides strong authentication. Green cloud is used to validate the process. The results confirm that our proposed SSA protects the mobile cloud computing from malicious activities. Along with SSA we proposed protocols that are leverage trusted authority entities and the "elastic" virtualized nature of the cloud computing model to provide energy-efficient key management mechanisms and policy-driven data protection techniques that support the secure interaction of the mobile client with an assortment of cloud software and storage services. This ensures that mobile cloud is protected from unauthorized access and malicious activities.

Keywords: Cloud Computing, Authentication algorithm, security mechanism

I. INTRODUCTION

Over the years technology has covered every aspect of human life and has transformed into a utility. Aim of technology is to facilitate humans as much as possible so it's moving towards integrating real life critical areas such as education, health, finance and many others with emerging technologies. Now-a-days the usage of mobile devices increases rapidly, about 95% of the people are using mobile devices particularly smart phones. Smartphone's are overpowering the IT world by rising as a prerequisite for other technologies. Emerging technology paradigms such as Cloud computing, web data services, online banking and many others are revamping them as compatibility to Smartphone's. The smart phones provide several utilities such as short message service (SMS), multimedia message service (MMS) and videos. The mobile phone provides the mobility support to move from one network to another network rapidly as this feature meets our needs. On the other hand, it provides the platform for adversary to attack on our sensitive data. Furthermore, there is high possibility to slow down the service. Generally, the mobile computing architecture consists of "Radio Sub System (RSS), Network Sub System (NSS) and Operating Sub System (OSS)". In RSS all the mobile stations are connected to base station sub system. In NSS all the base station controllers are connected to mobile controller. The operations Support Systems (OSS) consists of an authentication center that controls the NSS and RSS. There are several possible attacks expected on the mobile cloud computing such as privacy, integrity, and authentication. In mobile computing the large data is stored in the cloud servers. Here, we have to consider three factors: The mobile cloud should securely store the data, the data should be transferred correctly and finally data should be received by the correct user. Handling these three factors is cumbersome process. Thus, transferring data from the cloud to mobile devices face the problem of malicious users that can exploit the confidential and sensitive data.

Limiting the access of adversary to mobile cloud computing, there is need of strong authentication process. Here, we propose methods to authenticate the mobile client also for data security. The Authentication method is secure strong authentication process based on One-time password (OTP). In this process, the user information is forwarded to cloud owner prior to accessing the mobile cloud that helps to identify the authenticity of the user. If the mobile cloud user is authenticated then OTP is sent for accessing the cloud otherwise, the access is denied. In our proposed approach, we store the key on different mobile clouds, so it is difficult to the attacker to break the key. This process secures the data in the cloud servers by providing strong authentication. For the security of data we proposed protocols leverage trusted authority entities and the "elastic" virtualized nature of the cloud computing model to provide energy-efficient key management mechanisms and policy-driven data protection techniques that support the secure interaction of the mobile client with an assortment of cloud software and storage services. Our proposed approach protects the mobile devices against authentication attacks.

II. PROPOSED METHODS

In order to keep the data more secure, we designed the methodologies.

Secure and Strong Authentication algorithm

The Authentication method used is a secure and strong authentication process. For strong authentication first we have to make a strong key, for the strong key we use SSA algorithm to encrypt the data. The cloud owner only has the idea about the encryption algorithm. After encryption, the key is stored on different cloud servers by splitting, so when the attacker wants to break the key it is very difficult to get each piece of the key from the different cloud servers. Even if the attacker gets all the pieces of the key then it is not capable to understand the pattern of the keys. By following this strong authentication process, we can easily secure our sensitive data from the malicious users in the mobile cloud computing. In this strong authentication process when user wants toget access to the data, he/she has to get the key from the owner. The process to get the key is explained in algorithm.

Algorithm: Secure and strong authentication process

1. Initialize: (U_i= user, O_i = owner, A_s= authentication server, C_s = cloud server)
2. User requests => owner
3. Owner sends => authentication server
4. Authentication server gives token => user
5. User maps token to cloud server
6. Cloud server access
7. Key 1 =>C_{s1}
8. Key 2 =>C_{s2}
9. Key 3 =>C_{s3}
10. Get access to the resources

Firstly, the user requests the cloud owner to use the data present in the cloud server. Then the cloud owner collects the user data and sends that to the authentication server that checks whether the user is legitimate or not and sends token to the user, if the user is legitimate. Then the user sends the token to the cloud server the cloud server checks the information and generates key for the user, if the user is not the correct person then it sends information to the owner. Then owner knows that some attacker is trying to get access to his sensitive data and stops the process. When the key from the user matches to the cloud server then it provides access to the user to use the resources. The complete secure authentication process is depicted in Figure 1.

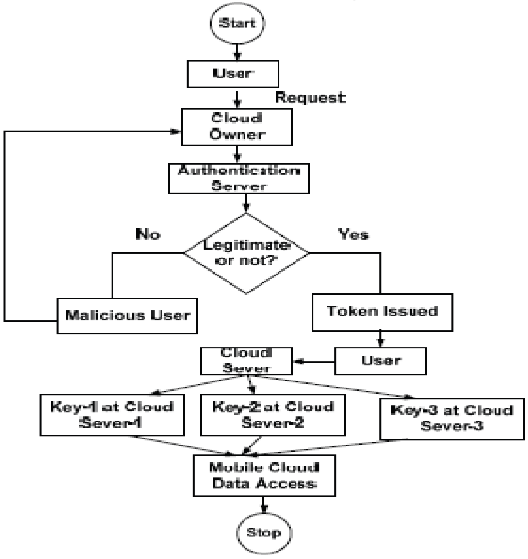

Fig1. Secure and Strong authentication process for mobile computing

In strong authentication algorithm when user U_i requests the owner O_i for the resources, then the owner sends the information of the user to the authentication server A_s. As authentication server checks whether the user is legitimate or not and generates token for the user. Through the token, the user gets access to the cloud server. After getting the access to the cloud, the user gets the data whatever requires. This is strong enough process to break the security.

Policy-driven security mechanisms.

This is used to provide the security to the data when it is transferred from cloud to client.

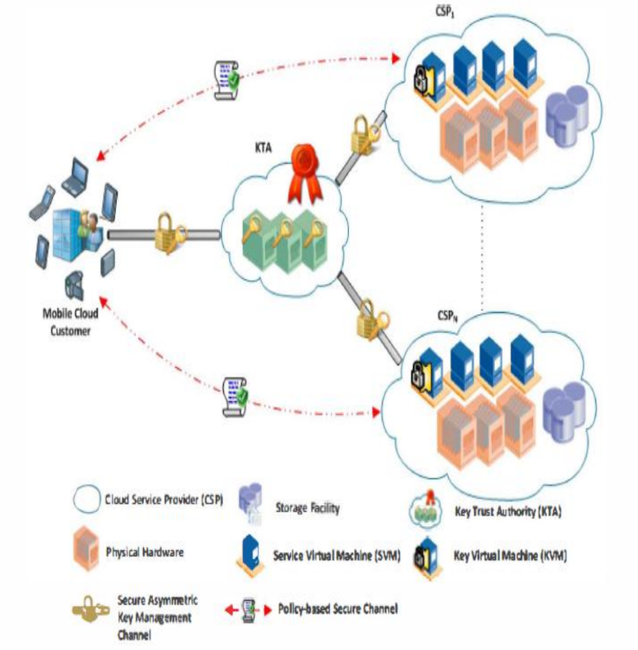
CSP$_1$
KTA
Mobile Cloud Customer
CSP$_N$
Cloud Service Provider (CSP)
Storage Facility
Key Trust Authority (KTA)
Physical Hardware
Service Virtual Machine (SVM)
Key Virtual Machine (KVM)
Secure Asymmetric Key Management Channel
Policy-based Secure Channel

Fig 2. System Model

The system model assumed in this work consists of the following entities: a mobile cloud client consuming a set of services provided by one or more cloud service providers (CSPs). The service provider supports an Infrastructure as a Service (IaaS) cloud computing model where the different services provided execute in a virtual container on top of the physical commodity hardware. Each service is provided by a dedicated service virtual machine (SVM) on the CSP side. We leverage the "elastic" property of cloud computing to seamlessly allocate and release key management virtual machines (KVMs) to facilitate efficient key agreement mechanisms with the mobile customer. The system model also relies on a trusted key authority (KTA) that represents the mobile client main interface with the KVMs on the provider side. The KTA is trusted by both the cloud customer and providers and it is mainly responsible of providing the client with the necessary attributes and policies to secure its network connections using multi-level symmetric-key security channels instead of resource-intensive public-key channels. The system model is illustrated in Fig 2.

Two types of security channels are employed in the proposed protocol:

(1) public-key based channels linking the cloud customer with the KTA and the KTA with the KVMs of the various cloud providers supporting the client services.

(2) symmetric-key policy-based security channels that secure the actual client network communication with the virtualized services. The public-key channels secure the key management protocols to provide the client with a set of

symmetric key security associations to support efficient policy-based interaction with the CSPs.

System Design

In this section we present the different security protocols executed by the communicating entities comprising the mobile cloud computing model we described in the previous section. The system design consists of four security protocols: the first three are supportive protocols that manage the key agreement and policy exchange mechanisms. The fourth protocol is the main protocol that governs the policy-based secure interaction between the mobile client and the virtualized services in the cloud.

The Security Parameters Collection Protocol

The Security Parameters Collection protocol is periodically executed between the KTA and the different CSPs registered in the proposed policy-based security solution. The main point of contact on the CSP side is the KVM entity which is responsible, when contacted by the KTA, of:

1. Querying the different SVMs for the IDs of the customers registered in each SVM published service and for the service security policies (SSPs) that should govern the data confidentiality and integrity mechanisms between the mobile client and the SVM.
2. Generating a set of multi-level symmetric cryptographic keys for performing the policy- based security mechanisms between the mobile client and each SVM.
3. Updating each SVM with the different client keys generated in step 2.
4. Sending the registered client's SSPs and ciphering keys over a secure SSL channel to the KTA.

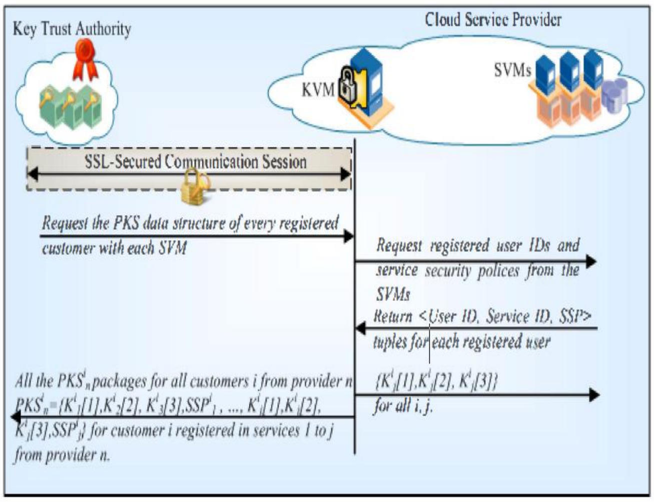
Key Trust Authority
Cloud Service Provider
KVM
SVMs
SSL-Secured Communication Session
Request the PKS data structure of every registered customer with each SVM
Request registered user IDs and service security polices from the SVMs
Return <User ID, Service ID, SSP> tuples for each registered user
All the PKS^i_n packages for all customers i from provider n $PKS^i_n = \{K^i_1[1], K^i_2[2], K^i_3[3], SSP^i_1, \ldots, K^i_j[1], K^i_j[2], K^i_j[3], SSP^i_j\}$ for customer i registered in services 1 to j from provider n.
$\{K^i_j[1], K^i_j[2], K^i_j[3]\}$ for all i, j.

Fig 3. SCP Protocol

The Security Parameters Collection protocol steps are illustrated in Figure 2. In Figure 2:- K_j^i[l], K_j^i[2] &K_j^i[3] are the cryptographic symmetric keys for supporting the policy-based encryption and Message Authentication Code (MAC) operations on customer i's data delivered by the SVM service j. K_j^i[l] is the ciphering key with which the encryption algorithm executes the maximum number of rounds on a plaintext data block and thus provides the highest security strength. K_j^i[l] is typically a 256-bit key in today's symmetric key standards. K_j^i[2] is the medium strength encryption key and is typically 192-bits in size. K_j^i[3] is the lowest strength cryptographic key which provides suitable security with the best encryption throughput. This key is practically 128-bits in size.

- SSP_j^i is the service security policy specifying the degree and range of encryption and MAC operations on customer j's data delivered by the SVM service j.
- PKS_n^i is the data structure that encapsulates the different service encryption keys and SSPs for customer i provided by the nth CSP (a CSP may provide multiple services per client via a set of SVM infrastructure virtual machines. The PKS package of each client is securely transferred to the KTA via an SSL secure session. The KTA executes the Security Parameters Collection protocol periodically and the period is determined based on the security requirements of the CSP services provided. The more sensitive the service is the more frequent this phase is updated to refresh the client cryptographic keys and policies and thus to elevate the system security level.

It is worth mentioning here that the PKS client packages are not always pulled from the KVM upon the KTA request. In many cases the KVM pushes these packages to the KTA to support the registration of new customers or to declare the service revocation of existing ones. Note that, for scalability reasons, the KTA sites may be replicated to enhance the quality of service and to avoid single protocol points of failure.

The PKS Retrieval Protocol

The PKS Retrieval protocol is executed by the mobile client for retrieving the PKS packages of its different registered services from the KTA. The network communication in this phase is secured using a single SSL session with the KTA. This session effectively aggregates all the public-key sessions that were necessary in the traditional security approach to agree on client-service symmetric keys. The session aggregation concept results in major energy savings on the mobile client especially when the mobile client consumes a relatively large set of services in the cloud(s).

The SSP Update Protocol

The SSP Update protocol allows the client to update the default SSP retrieved from the KTA for better supporting its performance and security requirements. To achieve this, the client sends the updated SSP to the respective KVM which in effect studies it and decides whether to accept or reject this update based on the service security requirements and the compliance of the new SSP with them. The client is explicitly notified of the KVM decision. To secure the policy communication in this protocol, the client adds a MAC to the policy specification to secure its integrity.

The Policy-based Secure Communication Protocol

The Policy-based Secure Communication protocol is the main pillar in providing the essential security services needed for protecting the mobile cloud data all the way from the SVM to the mobile device. The security services supported in this protocol are controlled and configured based on the security specifications present in the SSP and enforced using the cryptographic keying material retrieved in the Security Parameters Collection protocol. The SSP specifies the security behavior and operation of the mobile cloud application. The main purpose is to identify which data needs to be secured and to what degree. In general terms, the SSP is comprised of a set of conditions and rules over which the policy can be applied and a set of actions (encryption and MAC operations) that should be executed in the event of satisfying the conditions. The policy enforcement point is present in the respective SVM on the CSP side. A specialized policy enforcement engine inspects the cloud data retrieved from the cloud storage (which is the main data mine for the service) and applies the SSP encryption and MAC rules on this data before it is being transferred to the mobile client. After the client receives the policy-protected data, it applies the inverse operations to decrypt it and MAC verifies its integrity.

Utilizing a customizable security policy in the proposed protocol enhances to a great degree the scalability, flexibility, and customization of the security system. Moreover it facilitates the process of managing and administering the different elements composing the security architecture.

Without loss of generality we provide a description of a single client/service SSP. The policy typically consists of three main sections:

- General Identification Section: This section provides some identification information such as the type of policy whether it's the default policy provided by the SVM or a user defined policy updated by the client in the SSP Update Protocol. The service TD, CSP TD, and the customer TD are also identified in this section.
- Security Algorithms Section: This section mainly identifies the security algorithms that should be employed for ensuring the confidentiality and integrity of data flowing from the CSP services to the mobile client. Some possible algorithms are: the encryption, the MAC, and the data encoding algorithms.
- Protection Rules Section: This section controls the scope and level of the security operations to be applied on the cloud data before it leaves the SVM on the CSP side. Assuming that the cloud data is structured as a set of relational records composed of one or more fields (this is typically how data is stored in relational database management systems in the cloud storage facility). The scope determines what fields in the retrieved records are to be encrypted and MAC depending on a certain criteria related to the sensitivity of the field contents. For instance, any field content pattern matching a medical history record, a social security number, a credit card, or a banking account format should be encrypted. The level specifies the strength of encryption to be applied on the data field. Typically three encryption levels are provided depending on the strength of the encryption key used. Kji[1] provides the highest encryption strength, Kji[2] provides a medium encryption strength, while Kji[3] provides the lowest encryption strength. It should be noted that, in modern cryptography, as the key length increases, the security of the system as well as the processing requirements of the algorithm increase.

Not all the data fields are treated the same way by the policy enforcement engine. Each field is secured based on the protection rule it matches in the policy. The matching criterion is based on the field contents satisfying a particular regular expression. After the policy protection rules are applied on a certain data field, this field is tagged with the rule ID used to allow the policy enforcement engine on the mobile client to unambiguously decrypt the field and MAC verity its integrity as specified by the policy. The policy-protected data fields are added to the service reply and transferred to the mobile client. Upon receiving the protected data fields, the client extracts the protection rule tags attached to the fields to decipher them and verity their integrity as specified by the corresponding policy file.

III. CONCLUSION

This paper presented a set of policy-based security protocols and secure strong authentication algorithm for ensuring the confidentiality and integrity of enterprise data in mobile cloud computing. We proposed efficient key management techniques that suit the virtualized nature of the cloud computing service model to reduce the number of expensive asymmetric-key operations on the mobile cloud client. Moreover, we presented a secure policy-based communication protocol that efficiently protects the cloud service data based on content and sensitivity.

BIBILOGRAPHY

[1] Samad J, Loke SW, Reed K. Mobile Cloud Computing. Cloud Services, Networking, and Management. 2015:153-90.

[2] Nkosi MT, Mekuria F. Cloud computing for enhanced mobile health applications. In Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on 2010 Nov 30 (pp. 629-633).IEEE.

[3] Kumar K, Lu YH. Cloud computing for mobile users: Can offloading computation save energy? Computer. 2010 Apr 1(4):51-6.

[4] Zissis D, Lekkas D. Addressing cloud computing security issues. Future generation computer systems. 2012 Mar 31:28(3):583-92.

[5] Rizvi, Syed, Abdul Razaque, and Katie Cover. "Third-Party Auditor (TPA): A Potential Solution for Securing a Cloud Environment." Cyber Security and Cloud Computing (CSCloud), 2015 IEEE 2nd International Conference on. IEEE, 2015.

[6] Hashizume K, Rosado DG, Fernández-Medina E, Fernandez EB. An analysis of security issues for cloud computing. Journal of Internet Services and Applications. 2013 Dec 1:4(1):1-3.

[7] S.K.Sood, "A combined approach to ensure data security in cloud computing", in S.K. Sood/Journal

of Network and Computer Applications Vol.35, pp. 1831–1838, 2012.

[8] Ko SK, Lee JH, Kim SW. Mobile cloud computing security considerations. Journal of Security Engineering. 2012 Apr;9(2).

[9] Chetan S, Kumar G, Dinesh K, Mathew K, Abhimanyu MA. Cloud computing for mobile world available at chetan. ueuo. com. 2010.

[10] Rizvi, Syed, Abdul Razaque, and Katie Cover. "Cloud Data Integrity Using a Designated Public Verifier." High Performance Computing and Communications (HPCC), 2015 IEEE 7th International Symposium on Cyberspace Safety and Security (CSS), 2015 IEEE 12th International Conference on Embedded Software and Systems (ICESS), 2015 IEEE 17thInternational Conference on. IEEE, 2015.