ISSN 2395-1621

Cloud-based Secure and Privacy Improved Authentication Framework

[#1]Abdurrahman Saeed Noman, [#2]Dr. Mohamed A. M. Ibrahim

[1]abdulrahmansn86@gmail.com
[2]sabri196612@gmail.com

[#1]Information Technology & Engineering, Aden University, Yemen,
Msc. ITE.
[#2]Faculty of Engineering & IT, Taiz University, Yemen,
Head of IT & Management Msc Program.

ABSTRACT

Cloud computing is a new found service that has a rapid growth in IT industry during recent years. Despite the several advantages of this technology there are some issues such as data security and users privacy, authentication, and access control that affect the reliability of cloud computing models. The cloud consumer outsources their sensitive data and personal information to cloud provider's servers which is not within the same trusted domain of data-owner. So, there are several methods and mechanisms as well as ideas are proposed and presented to achieve fine grained security in cloud computing. In this paper we have proposed a scheme to achieve fine grained security with combined approach of public key cryptography and Kerberos in cloud computing. The proposed scheme provides authentication, confidentiality, integrity, non-repudiation, and privacy features to Cloud Service Providers and Cloud Users.

Keywords: Cloud computing, security, privacy, authentication, non-repudiation , Certificate, Kerberos.

ARTICLE INFO

Article History

Received: 11th June 2016

Received in revised form :

11th June 2016

Accepted: 24th June 2016

Published online :

5th July 2016

I. INTRODUCTION

Cloud computing is a large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet [1]. Cloud computing is the rapid growing Internet based technology that allows computer resources to be shared on an on-demand basis. In cloud computing end-user don't aware about where data is stored and how their data is being processed. They only access data, process and finally store them in the cloud. They can access data at anytime, anywhere if they are having Internet connection. This technology is highly scalable, flexible and distributed in nature [5]. Cloud computing is an Internet-based computing solution where shared resources/services are provided like electricity distributed on the electrical grid [2]. Cloud computing is new concept of computing technology, which currently comes in picture because it have various advantages from current technology. it converted every computing current technology as a service [4]. it provide basically three type of services. Software as a Service (SaaS) in which the cloud service provider provides applications and software over a network. Google Docs, Facebook, Gmail, Yahoo[3]. Platform as a Service (PaaS) provides application or development platform in which user can create their own application that will run on the cloud, example of PaaS are Microsoft's Azure, Google's Application Engine (app engine), Yahoo Pig [2]. Third type of cloud service is Infrastructure as a service (IaaS), the whole loud infrastructure, including servers, routers, hardware based load balancing, firewalls, storage and other network equipment is provided by the IaaS provider i.e. Amazon S3, Amazon EC2 [3]. Cloud computing can be deployed as public cloud, private cloud, hybrid cloud and community cloud. Public clouds are publicly available and can serve multiple tenants. Examples of public cloud are: Google App Engine, Microsoft Windows Azure, IBM Smart Cloud and Amazon [2] while private cloud is typically a tailored environment with dedicated virtualized resources for particular organization. Examples of private clouds are Eucalyptus, Ubuntu Enterprise Cloud–UEC, Amazon VPC (Virtual Private Cloud), vmware Cloud Infrastructure Suite, and Microsoft ECI data-centre. Similarly, community cloud is tailored for a particular group of customers Google Apps for Government, Microsoft Government Community Cloud are the example of community cloud [2]. Hybrid

cloud is composed of multiple clouds including public and private cloud like Windows Azure (capable of Hybrid Cloud), vmware v Cloud (Hybrid cloud Services).

Paper Organization

The rest of this paper is organized as follows. Section 2 discusses the literature review . Section 3 discusses the proposed model and its components. Section 4 provides a description of our proposed scheme in detail. Section 5 provides security analysis for our design. Section 6 conclusion our research. Section 7 it points out future work.

II. LITERATURE REVIEW

In recent year and according to the rapid growth of cloud computing technology, several user authentication models were proposed or designed by many researchers or enterprises.

Lee et al [6] have proposed public key and mobile out of band based authentication for cloud computing. However, the scheme transmits data (e.g. ID, PW, and PKI) in a plaintext form which can be easily intercepted by the adversaries. In addition, their scheme does not care about data confidentiality, data integrity, user privacy and users are not allowed to change their password. As result, their scheme is not fit for real time cloud computing.

A novel privacy enhanced anonymous authentication and access control scheme have been proposed to secure the interactions between mobile users and services in Pervasive Computing Environments (PCEs) with optional context, blind signature and hash chain, into a highly flexible and lightweight authentication and key establishment protocol [7]. It provides explicit mutual authentication and allows multiple current sessions between a user and a service, while allowing the user to anonymously interact with the service. The requirements of privacy and security for (PCEs) are realised and analyzed that existing privacy-preserving access control schemes do not fully satisfy these requirements so proposed two approaches to enhance privacy against malicious, and to enhance security [8].An authentication and authorization protocol for anonymous communication in the cloud is proposed [9]. The protocol is an extension of existing standards making it easy to integrate and compatible with existing standards. New password authentication schemes that support the Diffie–Hellman key agreement protocol over insecure networks are proposed [11]. A method of implementing two factor authentication using mobile phones is also proposed [12].The proposed method guarantees that authenticating to services, such as online banking or ATM machines, is done in a very secure manner. An authentication based on sending one time password to registered mobile number is proposed [13]. The SMS system doesn't guarantee to deliver the token at real time.

As we can see from above literature, we exist security issues in cloud computing have many security flaws and the data can still be intercepted by the malicious persons. This paper address most of the security concerns of cloud computing such as authentication, confidentiality, integrity, non-repudiation, and privacy to create and improve strong framework for cloud computing.

III. PROPOSED MODEL

The security and privacy in cloud computing will be important criterion for large-scale distributed computing. To balance the requirements for privacy against malicious insiders, with the security requirements for authentication, and data confidentiality, integrity, non-repudiation, we adopt an efficient hybrid approach that combines both public key based on the certificate, and symmetric key based on the Kerberos KDC.

Components of our proposed model

A. Trust Third Party (TTP)

The TTP is central authority that is invoked in exceptional circumstances (e.g., certification, dispute resolution, anonymity revocation).

Central Authority is a trusted third party that issues digital certificates Digital certificates allow entities to securely share their public key with individuals with whom they have no established relationship. Digital certificates are copies of the entity's public key that are digitally signed by a CA.

Central Authority is arbiter entity that verify the digital signature to solve disputes between parties.

B. Kerberos (KDC)

Cloud computing is open distributed servers environment, in which users wish to access services on servers distributed throughout the network We would like for servers to be able to restrict access to authorized users, and authenticate trusted requests for service, and prevent malicious persons and threats, Rather than building in elaborate authentication protocols at each server, Kerberos provides a centralized authentication server[14]. Kerberos is an authentication protocol for network security based on Symmetric cryptography. It provides mutual authentication and message integrity as well as data confidentiality. It uses secret key cryptography, which proves identity of communicating parties over network, and also prevents eavesdropping or replay attacks [15]. Kerberos performs secure verification of users and services based on the concept of a trusted third party (KDC) [16].

Components of the Kerberos (Servers)

The Kerberos authentication system consists three servers i.e. Authentication Server (AS), Ticket Granting Servers (TGS) and real server (CSP) that provides services to others [3].

Authentication Server (AS)

It is the KDC in the Kerberos. We suppose that all users share and store their certificates in a centralized database at authentication server (AS). And the AS shares a unique secret key with each server, then each user registered with AS and is granted a user identity and password and keep these credentials in its database of every individuals. AS

verifies the identity of the user (A), and issues a session key to be used between user and TGS.

Ticket Granting Servers (TGS)

It provides service access ticket to user (A) who has already been authenticated, and it issues a ticket for the real server (B). It also provides the session key K_{AB} between user (A) and real server (B).

Real Server (CSP)

It is the final executor of service, that provides services to the users.

IV. THE HYBRID APPROACH TO AUTHENTICATION USING PUBLIC-KEY CERTIFICATE AND KERBEROS

Since Kerberos does not support non repudiation, this weakness of Kerberos [3]. but in our proposed, we use hybrid system that can support public key cryptography and Digital Signatures depends on certificate , and use central TTP to verify the signature and prevent repudiation, and as well to solve dispute between user and CSP, so the our proposed can deploy successfully with Kerberos.

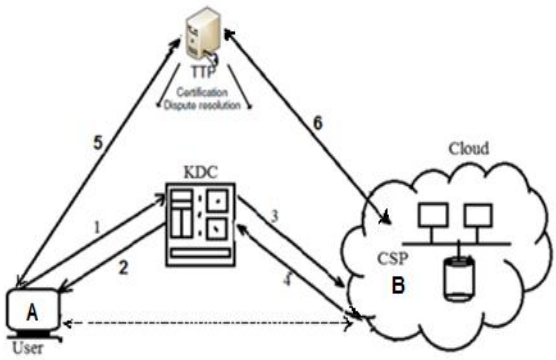

Figure 1: Proposed Model for Authentication for cloud

As shown in figure 1 the basic idea of the proposed model is as follows.

Step-1. User register his identity to Kerberos (KDC).

Step-2. KDC provides ticket to user to communicate with CSP.

Step-3. KDC also send a ticket and user identity to CSP, now CSP stores these credentials for future use.

Step-4. CSP acknowledge to KDC about user's credentials storage.

Step-5. User encrypts his data, and put his digital signature before sending to cloud, then user sends his message to TTP.

Step-6. TTP decrypts the signature and checks the user signature to validate the message, then TTP send the message to B.

Working of Kerberos in hybrid approach

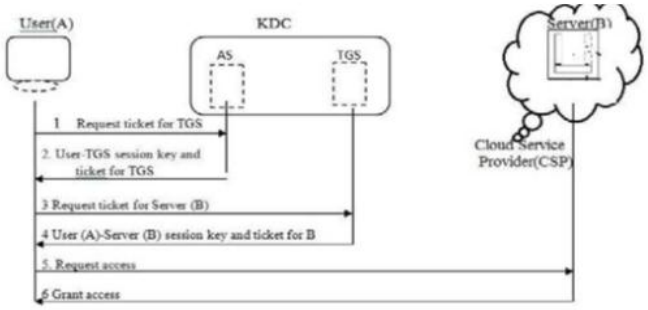

Figure 2. Kerberos Authentication of Cloud Service Provider.

As shown in figure 2, the basic steps of the Kerberos scheme is as follows.

Step 1. A $\longrightarrow$ AS : $ID_A \| ID_{TGS}$

- The user A, sends his request for service to AS. Where:
  ID_A = identifier of user on A ,
  ID_{TGS} = identifier of Ticket Granting Servers,
  AD_A = network address of A.

Step 2. AS $\longrightarrow$ A : $E(K_{A\text{-}AS}, [K_{A\text{-}TGS} \| ID_{TGS} \| Ticket_{tgs}])$.
, $Ticket_{tgs} = E(K_{AS\text{-}TGS}, [K_{A\text{-}TGS} \| AD_A \| ID_A \| ID_{TGS}])$

- The AS sends a message encrypted with User's (A) permanent symmetric key, $K_{A\text{-}AS}$.
- The message consists two items: a session key $K_{A\text{-}TGS}$ that is used by user A to contact the TGS and a ticket for the TGS that is encrypted with the TGS symmetric key $K_{AS\text{-}TGS}$.
- The user types his symmetric password correctly then the appropriate algorithm together creates $K_{A\text{-}AS}$, because he does not know the $K_{A\text{-}AS}$.
- The password is destroyed immediately, it is not send to the network and it does not stay in the terminal. It is used for a moment to create key, $K_{A\text{-}AS}$. Process now uses $K_{A\text{-}AS}$ to decrypt the message sent. $K_{A\text{-}TGS}$ and the ticket are extracted.

Step 3. A $\longrightarrow$ TGS : $ID_B \| Ticket_{tgs} \| Authenticator_A$.
$Ticket_{tgs} = E(K_{AS\text{-}TGS}, [K_{A\text{-}TGS} \| AD_A \| ID_A \| ID_{TGS}])$
$Authenticator_A = E(K_{A\text{-}TGS}, [ID_A \| RN_1])$

- User (A) now sends three items to the TGS.
  - The first is the ticket received from AS,
  - The second is the name of the real server (B) (i.e. Cloud Service Provider),
  - The third is Authenticator of user A, that encrypted by $K_{A\text{-}TGS}$. And contain random numbers (RN_1) is instead of timestamp to prevent replay attack.
  - We replace timestamp by random numbers, because the time doesn't

synchronize in the distributed network environment.

Step 4. TGS ⟶ A : $E(K_{A\text{-}TGS}, [K_{A\text{-}B}||ID_B||RN_1||Ticket_B])$.

$Ticket_B = E(K_{TGS\text{-}B}, [K_{A\text{-}B}||AD_A||ID_A||ID_B])$

- The TGS sends two tickets, each containing the session key between user(A) and real server(B). $K_{A\text{-}B}$,
- The ticket for user (A) is encrypted with $K_{A\text{-}TGS}$.
- The ticket for server (B) is encrypted with B's public key $K_{TGS\text{-}B}$.
- Note: Eve cannot extract $K_{A\text{-}B}$, because Eve does not know $K_{A\text{-}TGS}$.
- The user (A) will compare the random number RN_1 , with random number that itself sent to the TGS, If the two numbers are equal, then it can be confirmed that this message is a new message.

Step 5. A ⟶ B : $Ticket_B||Authenticator_A$.

$Ticket_B = E(K_{TGS\text{-}B}, [K_{A\text{-}B}||AD_A||ID_A||ID_B])$

$Authenticator_A = E(K_{A\text{-}B}, [ID_A||RN_2])$

- User (A) sends Server (B) ticket with the new random number to prevent replay attack, encrypted by $K_{A\text{-}B}$.

Step 6. B ⟶ A : $E(K_{A\text{-}B}, [RN_2 + 1])$. (for mutual authentication)

- Real server B confirms the receipt by adding 1 to the random number. The message is encrypted with $K_{A\text{-}B}$ and send to user (A).

After successful authentication by Kerberos the user (A) uses his private/public keys to achieve digital signature that provide message authentication and integrity, and use secrete session key to encryption process for confidentiality. The user A and Server B may be agree to change symmetric session key K_{ss} , during the session.

Authentication and Integrity

In authentication process, the sender (A), first calculates the message digest of the data, and put his digital signature which figure illustrates:

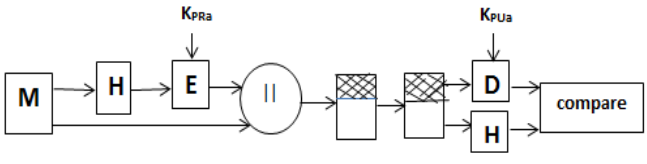

Figure 3. Authentication, and Signature

A ⟶ B : $M||E(K_{PRa}, H(M))$

- Provides authentication and signature, Only A has K_{PRa} to encrypt.
- Provides Integrity , using hash function H(M).

Step-1. User calculates the message digest of the message .

Step-2. And then, he encrypts this digest with his private key (put his digital signature).

Step-3: He concatenates the original message with encrypted message digest and sends to Cloud service provider CSP. All these three steps are perform by user and next following steps are perform by Cloud Service Provider as:

Step-4: After receiving the message from user, the CSP decrypts the digest with user public key and get the message digest.

Step-5: The CSP calculate the message digest of the message received using same hash function.

Step-6: If both digest comparison calculated same; it shows that the sender is authentic user, whose public key is available to CSP repository. Also calculated digest show that the integrity of the message is uniform.

Confidentiality, Authentication and Integrity

To provide confidentiality using several steps as follows

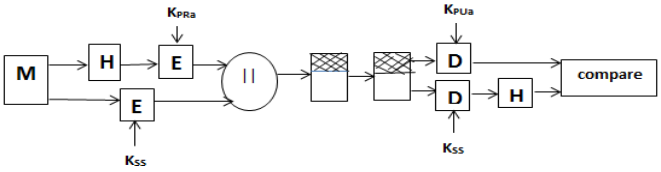

Figure 4. Confidentiality, Authentication, and Signature .

A ⟶ B : $E(K_{SS}, M)||E(K_{PRa}, H(M))$

- Provides confidentiality, only A and B share Kss
- Provides authentication and signature, Only A has K_{PRa} to encrypt.
- Provides Integrity , using hash function H(M)

Step-1. User calculates the message digest of the message.

Step-2. After calculating digest he encrypts this digest with his private key (put his digital signature).

Step-3: User encrypts original message by secrete session key**.**

Step-4: He concatenates the secrete message with digital signature of digest, and sends to Cloud service provider

Step-5: After receiving the message from user, the CSP decrypts data by symmetric session key .

Step-6: Then the CSP decrypts the digest with user public key and get the message digest.

Step-7: The CSP calculate the message digest of the message received using same hash function.

Step-8: if both digest comparison calculated same; it shows that the sender is authentic user, Also calculated digest show that the integrity of the message is uniform.

Non Repudiation using digital signature

Non-repudiation is a property which using digital signature and prevents an individual or entity from denying having performed a particular action related to data through cryptographic methods.

Every signed message from a sender A to a receiver B goes first to trusted an arbiter TTP.

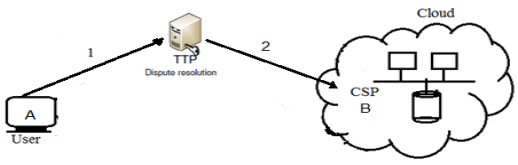

Figure 5. *Non-Repudiation*

A ⟶ TTP : ID_A||E(K_{SS}, M) || E(K_{PRa}, [ID_A||H(E(K_{SS}, M))])

- Provides confidentiality, only A and B share Kss
- Provides authentication and signature, Only A has K_{PRa}
- Provides Integrity , digest of secrete message.
- Provides non-repudiation, TTP verifies contents, and add his signature and a timestamp.
- The timestamp informs B that this message is timely and not a replay.

The trusted center TTP decrypts the signature and checks the hash value to validate the message. Then TTP transmits everything that it received from A, plus a timestamp, to B.

TTP ⟶ B : ID_A||E(K_{SS}, M) || E(K_{PRt},[E(K_{PRa} ,[ID_A||H(E(K_{SS}, M))]) **||T])**

B can store M and the signature. In case of dispute, B, who claims to have received M from A, sends the following message to TTP :

E(K_{PRt}, [E(K_{PRa} ,[ID_A || H(E(K_{SS}, M))] || T])

V. SECURITY ANALYSIS.

1. User Privacy :

no unauthorized entity, external or internal to the system can be able to trace the real identity of the user, or to link different sessions of the user, unless the user or system policy explicitly permit it. So the proposed scheme never transmits user private data in plaintext format. The messages are transmitted over a public channel. Clearly, these messages cannot be decoded easily to get ID, PW etc. Hence, the scheme provides user privacy.

2. *Session Key Agreement* :

In proposed scheme, the session key, K_{SS} is established between the user and the CSP after authentication process. Using this key they can communicate with each other for a particular session, This key is generated carefully, and it's different in every login session, so it cannot be breach easily and cannot be replayed after the session expires.

3. Identity Management :

In proposed scheme, the KDC and CSP store all the registered IDs in the database and checks availability of a unique ID in each new registration and provide certificate to manage the identity of user.

4. The replay attacks :

The random number was introduced into the Kerberos protocol to replace the timestamp in the traditional Kerberos protocol, which avoids the clock synchronization problem in the network. When the user (A) receives the response information sent by TGS, User will decrypt the response message with session key between A and TGS to get RN1. Compare RN1 with the random number that user sent to TGS. If they are the same, it means that the message is new, not a retransmitted one, which can prevent the replay attacks.

5. Mathematical Attack :

This attack will occur by determining p, q or f(n), where p,q are the large prime numbers, and n=p.q , F(n)=(p-1)(q-1), It could be prevented by using 2048 bits exponents in RSA. Also it could be prevented by increasing the value of digest *h*, the chance of successful mathematical attack would be decreased considerably.

6. *Security against Brute Force Attack:*

All possible combinations to guess the private key have been tried by the attacker during the brute force attack. In the original RSA, the probability of failure against this attack will be decreased considerably by choosing exponents larger than 2048 bits.

VI. CONCLUSION AND FUTURE WORK

In our proposed, we use public key certificates along with Kerberos based security in cloud to achieve fine grained security and to improve authentication scheme. Kerberos proves identity of users over networks and provides data integrity and secrecy. Kerberos performs secure verification of users and services based on the concept of a trusted third party (KDC). and we have kept our framework flexible enough to integrate with trusted third party TTP to prevent repudiation, and to solve dispute.

In the future work, we are focusing on the efficient data security methods to control auditing of dynamic data storage with fine grained data updates, when the verification is done by trusted third party .

REFERENCES

[1] Foster, I., Zhao, Y., (2008). *Cloud Computing and Grid Computing 360-Degree Compared*. In: Grid Computing Environments Workshop (2008)

[2] Patel, S.C., Umrao, L. S. & Singh, R. S. (2012). *Policy-Based Framework for Access Control in Cloud Computing*, International Conference on Recent Trends in Engineering & Technology (ICRTET2012) ISBN: 978-81-925922-0-6

[3] Subhash C. P, Ravi S. S., and Sumit J.(2015). *Secure and Privacy Enhanced Authentication Framework for Cloud Computing*. IEEE Sponsored Second International

Conference on Electronics and Communication (ICECS '2015)

[4] Sandeep Sax, G. S. , Shashank S., (2014). *Mutual Authentication Protocol Using Identity Based Shared Secret Key in Cloud Environments*. IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)

[5] Sanjeet K.N. , Subasish M. & Banshidhar M.(2012). *An Improved Mutual Authentication Framework for Cloud Computing.* International Journal of Computer Applications (0975 – 8887)

[6] S. Lee, I. Ong, H.T. Lim, H.J. Lee, *"Two factor authentication for cloud computing",* lnternational Journal of K1M1CS, vol 8, Pp. 427-432

[7] Kui Ren · Wenjing Lou "*Privacy-enhanced, Attack-resilient Access Control in Pervasive Computing Environments with Optional Context Authentication Capability*"

[8] Emmanouil Magkos, Panayiotis Kotzanikolaou, "*Achieving Privacy and Access Control in Pervasive Computing Environments*" SECURITY AND COMMUNICATION NETWORKS *Security Comm. Networks* 00: 1–12 (2010).

[9] Umer Khalida, Abdul Ghafoor, Misbah Irum, Muhammad Awais Shibli, "*Cloud based Secure and Privacy Enhanced Authentication & Authorization Protocol*" International Conference in Knowledge Based and Intelligent Information and Engineering Systems - KES2013. Procedia Computer Science 22 (2013) 680 – 688.

[10] H. Liping, S. Lei, Research on trust model of pki, in: *Intelligent Computation Technology and Automation (ICICTA),* 2011 International Conference on, Vol. 1, IEEE, 2011, pp. 232–235.

[11] Liao, I-E., Lee, C.-C. & Hwang, M.S. (2006). *A password authentication scheme over insecure networks*" Journal of Computer and System Sciences 72 (2006) 727–740,Elsevier

[12] Aloul, F., Zahidi, S. & El-Hajj, W. (2009). *Multi Factor Authentication Using Mobile Phones*", International Journal of Mathematics and Computer Science, 4(2009), no. 2, 65–80.

[13] Choudhury A. J., Kumar P., Sain M.,Hyotaek L. andHoon J., "*A Strong User Authentication Framework for Cloud Computing*", Services Computing Conference (APSCC), 2011 IEEE Asia-Pacific,2011.

[14] William Stallings.(2005). *Cryptography and Network Security Principles and Practices, Fourth Edition.***,** Pub Date: November 16, 2005 (ISBN : 0-13-187316-4)

[15] Srinivasa Rao Yarlagadda, Rupesh Shantamurty "*Kerberos authentication made easy on OpenVMS*", OpenVMS Technical Journal V18, http://h71000.www7.hp.com/openvms/products/kerberos/

[16] Gary C. Kessler, *An Overview of Cryptography* , Handbook on Local Area Networks 1999 edition ,short edition, May2 .2014.

[17] *Min Li , Xin Lv , Wei Song , Wenhuan Zhou, Rongzhi Qi, " A Novel Identity Authentication Scheme of Wireless Mesh Network Based on Improved Kerberos Protocol "* 2014 13th International Symposium on Distributed Computing and Applications to Business, Engineering and Science**.**

