International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 24 (2017) pp. 15693-15698
© Research India Publications. http://www.ripublication.com

An Enhanced CTES Design for Authentication and Authorization to Cloud Services and Resources

Ashok Kumar J[1] and Gopinath Ganapathy[2]

*[1,2]School of Computer Science, Engineering and Applications Bharathidasan University,
Tiruchirapalli -620023, TamilNadu India.*

[1]Orcid Id: 0000-0002-4969-8257

Abstract

This paper elaborates the framework model for authenticating and authorization of cloud services and resources in the cloud environment. The Collaborative Trust Enhanced Security (CTES) model uses the traditional Kerberos protocol for authentication and extend it to provide the access policy for each node in the cloud environment. It can be enhanced in term of its performance by altering the authentication and eliminating the access control mechanism, which is used in agent system, as it increases the complexity of the trust level for each client authentication. So access control policy enforced with the agent system, is not required and not advisable, if it is imposed directly with modified characteristics of Kerberos protocol itself. GnuPG based public key and private key works based on the framework of "web of trust" model to manage the different trust relationship of user and establish the authenticity of binding between the user and public key in the system node. In this paper, the framework of CTES model is analyzed to bring out the issues faced with agent system and then the access control policy for each user to be imposed to suit with the Kerberos protocol itself. This paper also describes by extending the CTES model with web based applications in order to provide end to end security environment for Cloud resources and services.

Keywords: Kerberos, CTES, GnuPG, end to end security

INTRODUCTION

In a cloud environment, the server is responsible to provide the service to client in an organized manner and client utilizes those service to perform the desired task which is assigned specifically for the user. Establishing the Trust for user, is one of the most sensitive and important issue for maintaining the privacy of user information and so a strong authentication mechanism shall be used for client server interactions [1]. The user utilizes the facility of "pay as per use" of cloud computing to save cost of setup and maintenance of resources and so the risk related to data privacy can be managed by cryptography techniques. In order to access, services of cloud provider and data stored on cloud, it is necessary to have a valid authentication scheme such as Kerberos that can prove the identity of the user. The GnuPG identifies only the legitimate user gets the access to data. The CTES model represents a collaborative trust approach where all the users are trustworthy and work under the mutual authentication and authorization to users with their dynamic nature of joining and leaving the distributed systems. Trusted users that have been authenticated are often authorized to access the cloud resources based on their access privileges [2].

KERBEROS

Kerberos authentication system is a protocol applied in the authentication mechanism for the cloud environment. The KDC (key distribution centre) provides two services, one is AS (Authentication service), and the other is TGS (Ticket granting service)(Fig 1). The user's goal is to obtain services provided by Application servers (e.g., File Server, Email Server, Web Server, etc). Kerberos provides Ticket Granting Ticket (TGT) and Service ticket. TGT is used to validate its ticket at KDC, whereas Service ticket is to be validated at Application Server. Once the Kerberos grants the ticket, the user does not need to login every time to communicate with KDC and they will get service tickets for accessing different Application Servers[3]. The ticket TGT transmitted from client to server and then it is verified usually for each successful decryption of packet, will always be a burden to server and consume more computation power to balance the load of the server. An attacker can request many tickets by sending unencrypted message to Authentication Server and leads to guess password, since authentication is not required at AS. Kerberos does not provide the authorization policy for the user after they successfully authenticated in the client system.

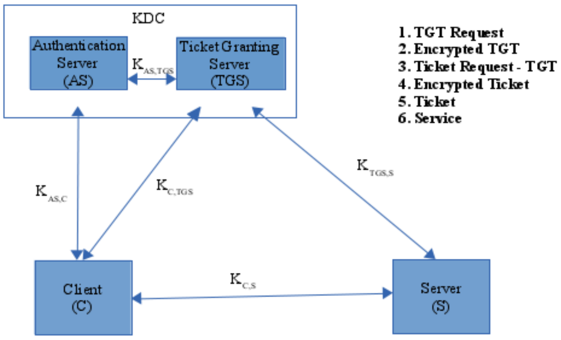

Figure 1: Traditional Kerberos Protocol

Pretty Good Privacy (PGP)

PGP was developed by Philip R. Zimmermann [4]. GnuPG (GNU Privacy Guard), is an open source compatible encryption system based on OpenPGP. PGP/GnuPG encryption, uses combination of public key cryptography, data compression, hashing and symmetric-key cryptography. It is used in several security constraints such as confidentiality, integrity and authentication for electronic mail and file storage applications etc., [5]. GnuPG creates the digital signature for the given data to verify the authenticity of the sender. Sender sends the hash digest along with the given data to the receiver. Then receiver uses the sender's public key to verify the digital signature. If it matches the digital signature, it will be confirmed that it is from the expected sender. The author analyzed and reviewed the secure transmission of electronic medical records from one point to another (such as from hospital to other hospital, or from hospital to a patient) with OpenPGP and S/MIME security protocols and added that some technical problems with regards to Transport Layer Security (TLS)[6]. In case of scattered companies, TLS does not scale to support a large community of partners and may be a reasonable option when dealing with a small number of partners and communication with different customers will not be easily able to manage the infrastructure[6].

This paper explains about the existing CTES framework model and analyzed limitations or issues in CTES framework and then proposes the modified CTES framework with improved Kerberos protocol and with end to end security environment. Finally, it discuss about the benefits of the proposed framework over CTES model.

CTES Framework Model

The CTES model represents a collaborative trust approach where all the users are trustworthy and work under the mutual authentication and authorization to users with their dynamic nature of joining and leaving the cloud environment. The model is based on hybrid approach that consists of four components Coordinators (SuperHost and Agent), KDC Key Distribution Center (AS- Authentication Server and TGS-Ticket-Granting Server), Server and Client. The coordinators and KDC are the reliable controller systems around which all the activities are scattered. There exists a mutual communication between the controller systems. SuperHost maintains records of the clients and the servers of the cloud environment, monitors the life time of the clients and servers and also controls the other controller systems. Agent manages the registry of service provided by servers, monitors clients behavior, handles client requests for services and balances the load of servers. The KDC is responsible for authenticating and authorizing the client when it requires a service. Client is a registered node of the cloud environment that can access the services and resources. The server creates a service and publishes its interface and access information to the service registry maintained by Agent[1].

Comparison of CTES model with Kerberos

Kerberos uses 6 messages for authenticating the client to the server, whereas the CTES model works on 12 messages. Kerberos uses 2 messages each for authenticating the identity of client and obtaining TGT; for obtaining SGT; and for accessing service, whereas CTES architecture uses 4 messages for authenticating the identity of client and obtaining TGT, 2 messages for obtaining SGT, 4 for obtaining reference of server from Agent and 2 for accessing service from server. Agent is involved for providing the reference of server to the client on the basis of the requested service. Before transferring message m8 to client, Agent analyzes the behavior of client in the system along with sending intimation to SuperHost to update the trust level of client and enforce the access control policy to the client system, which is not required and not advisable, if it is imposed directly with modified characteristics of Kerberos protocol itself. The following issues are analyzed deeply in the CTES model for providing better solutions under the section “Limitations in CTES model” and then suggested some modifications to the Kerberos protocol itself to enhance the CTES model framework in cloud environment under the section “Suggestions to modify the traditional Kerberos” and analyzed end to end security environment in the section “Suggestions for end to end security environments”.

Limitations in CTES model

The following limitations or issues are analyzed for CTES model along with traditional Kerberos.

1. Kerberos authentication database at KDC is not maintaining the user ID and password of all the users in the cloud environment whereas SuperHost accomplishes this task. SuperHost maintains the record of all clients who may access services and provide authentication for each client through Kerberos. It requires more number of authenticate messages and degrades the authentication performance.
2. Instead of providing the ID and IP address of Server system to access a particular application service, TGS provides the reference of Agent system to client and client will communicate with application server through agent system. This can be modified to impose access policy for client to access directly a particular application server with Kerberos itself.
3. Client requests Agent instead of SuperHost, to get the list of services for which client is authorized and the list of servers for a service is obtained using

messages m7, m8, m9 and m10. This information can be imposed directly from SuperHost to client without referring the Agent System.

4. The Web services shall comply the requirements of the CTES model which is most useful for web based deployment of Applications. So the existing solution for CTES model shall be extended to provide an umbrella for web based applications as well.

5. Applications can be deployed independently in the system by providing end to end security environment to enhance the CTES model.

Suggestions to modify the traditional Kerberos

Based on the paper Ashok Kumar J et al. [7] points out the below mentioned modification which are embed into traditional Kerberos protocol to resolve those issues, is analyzed in the section "Limitations in CTES model-1,2,3".

1) A pre-authentication technique shall be introduced for AS and TGS in order to protect the unauthorized access to KDC database.

2) The authorization access for client shall be enforced directly to the client, after the successful authentication of user.

3) Each client shall generate the GnuPG public key and private key by itself in the secure manner and stored securely in the client system itself. The issues of key distribution and key management in Kerberos are solved by distributing keys using public key algorithm.

4) The timestamp mechanism of traditional Kerberos protocol shall be replaced with the sequence number mechanism.

5) One Time Password (OTP) shall be used to overcome the password guessing attack. So password guessing attack is not possible because private key is independent of user's password.

6) Sequence number shall be used to overcome the replay attack. So replay attack is not possible because the sequence number is replaced instead of timestamp.

7) The client private key shall be available in the client system itself, so that it is client's responsibility to decrypt the authorized information about the user. It is more secure than the traditional authentication and authorized information should be enforced into the client system directly for the user.

8) Digital Signature is not present in traditional Kerberos, So the public key encryption shall be used to support Digital Signature.

Suggestions for end to end security environments

The author analyzed the email security protocols and standards, in order to adopt for secure transmission of electronic medical records from one point to another (such as from hospital to other hospital, or from hospital to a patient)[6]. Transport Layer Security (TLS) is used to secure a wide number of Internet applications. There are some technical problems with regards to TLS. TLS has become very popular for securing email communications between strategic partners due to its flexibility to set up between point-to-point and within known organizations. In case of scattered companies, TLS does not scale to support a large community of partners. Therefore, in case of health organizations where multiple hospitals may need to exchange e-mails, it's not advisable to adopt TLS for securing the data. Hence the OpenPGP provides better security in end to end form while compared with TLS and it follows the security procedures of confidentiality, non repudiation, authentication and integrity getting initiated at client end and applicable upto Application Server in the Cloud Environment.

Proposed modification for CTES Framework Model

In this paper, a novel design model is proposed for authentication and authorization of cloud services and resources, by modifying the CTES model. Based on the analysis of CTES model dealt in the section "Limitations in CTES model", altering the authentication and eliminating the access control policy with agent system provides better solution, as it increases the complexity of the trust level in each client authentication. CTES model uses the traditional Kerberos protocol for authentication and extend it to provide the access policy for each node through agent system in the cloud environment. This access control policy is enforced with the agent system which is not required and not advisable, if it is imposed directly with modified characteristics of Kerberos protocol itself as suggested in the section "Suggestions to modify the traditional Kerberos". This proposed approach for the framework of CTES model will be modified based on client authentication, access control policy and end to end security environment to work efficiently and will improve the performance with inbuilt access policy of Kerberos protocol.

The picture (Fig 2) describes the top level view for the proposed framework of interaction in between the SuperHost and KDC systems. SuperHost can be extended into "n" number of systems in the group of clustered system to manage the registered Application server and client, which is to be proposed for CTES model. The KDC system is placed infornt of the SuperHost to authenticate the users for the application services and to provide the access control rights for accessing the cloud resources in the cloud environment. The changes in each user profile will be replicated in one way direction that is from the SuperHost to the KDC system.

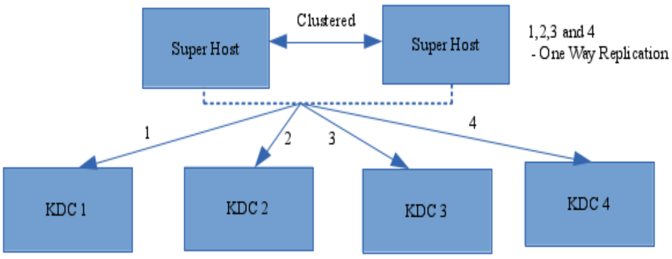

Figure 2: Top level view of new CTES model

Client authentication in the modified CTES framework

Client can access the services of the cloud environment as mentioned in below steps to authenticate with Kerberos and then to verify the capability list in the SuperHost. The interacting messages (Fig 3) used among the different servers are:

Step 1: The user login into the system node with his username and password. This authentication is based on Google Authenticater and the two way authentication is enabled with kerberos for centralized authentication.

Step 2: The message MSG1 is sent to the Authentication Server for authentication.

Step 3: The messages MSG1 is verified and MSG2 is used to check any new access policy is defined by the cloud service provider at SuperHost.

Step 4: If there is any change in the access policy for the authenticating user then it will be intimated through MSG3. This will intimate "null" if there is no change in access policy.

Step 5: The MSG4 contains the message for access policy for valid user (If valid MSG1) to Client. Otherwise the client will not be authenticated in the Kerberos Server.

Step 6: Client receives the MSG4, But it can not be read by the client. It is because the other valid part of access policy is required. So it requests the other part with MSG5 from TGS server.

Step 7: TGS server verifies the MSG5 and provides the another part of access policy with MSG6. If the MSG5 is not valid, then the connection is closed.

Step 8: Client receives the MSG6 from TGS server and validate this MSG6 with MSG4. If it matches then the access policy will be available to client. Thus the client authenticate the data with GnuPG private key. This private key is available in the client system itself and so it is more secure than any other methodology. This approach provides the way to end to end security environment.

Step 9: Then the client has the capability to work on Single-Sign-On with Application Server. For authenticating with Application Server, the client does not provide the password for that Application Server, it simply provide the token and get authorized to Application Server. This is applicable to all Cloud resource as well. The token MSG7 is sent to the Application Server.

Step 10: The Application Server verifies the token with TGS server for MSG8 and MSG9. If it matches, then the Application Server or any cloud resources will be available to provide the service to client.

Step 11: Client receives the MSG10 for authorization access with the Application Server or any Cloud resources.

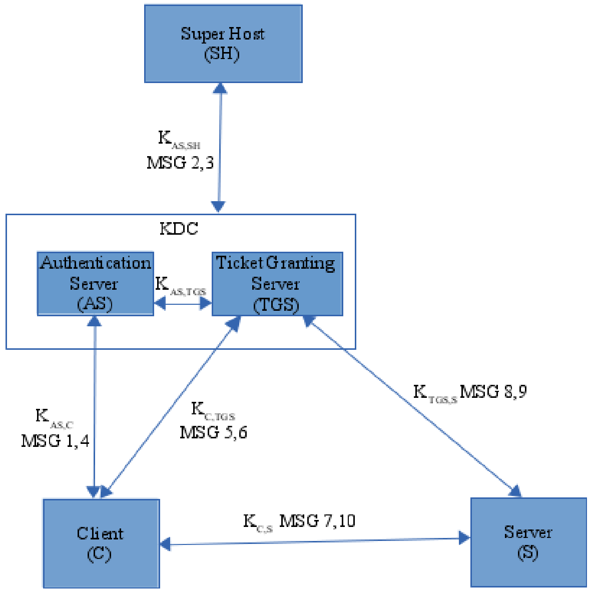

Figure 3: CTES model for Client Authentication and Authorizations

Access Control Policy in the modified CTES framework

Cloud computing offers a wide range of network access that is convenient and has configurable resources for network storage servers. It also enable the businesses to utilize cloud infrastructure to virtually synchronize and access the data. The PGP and the Kerberos combination are used for double process authentication purposes by integrating the features of both to provide the two-way authentication with the Google Authenticator [9]. The PGP user access can be established by the Google Authenticator. This two-way authentication occurs in the PAM (pluggable authentication module) with the combinations of PGP and Kerberos. Whenever the user login into system node, GnuPG user process is started and Kerberos authentication is performed automatically, then the access policy from cloud service provider is downloaded in to client program through Kerberos server and this change the behavior of the user policies with the SELinux. In this, the user asks for access to the file, then the Selinux policy is enforced to access the data or for computing the data for the user. Thus the cloud service provider access policy will be enforced directly with the use of Kerberos and enabled with SELinux.

This approach enables the CTES model more effective with the use of SELinux and GnuPG to process the data

End to End security environment in the modified CTES framework

This proposed design provides security in end to end form with the security procedures of confidentiality, non repudiation, authentication and integrity getting initiated at client end and applicable upto Application server. This end to end security procedures are applicable for web based applications [8] and for independent applications which are used to communicate from client to server or vice versa. It means the target Application server is only responsible to encrypt or decrypt the data. So the intermediate nodes do not have the capability to access those data. Thus GnuPG confirms that the data is arrived securely from sender by verifying with digital signature. In case sender sends data which contained malware, then the vaccine program of receiver cannot protect the malware, because of that the pattern of malware is also encrypted, so it is not exposed to the vaccine program [10]. Some times, there is a need to access the data by vaccine program or any other intermediate system to check the threats or malware in the data, for that the client, Application server and intermediate server shall use all their public key for encryption and for decryption, thus any one can decrypt with their private key. The virtual group key can be created with the use of GnuPG and will be used to encrypt with public key from client system or in intermediaries or with Application Server and finally this data can be decrypted with any one of the private key user. These three systems should be trusted with Kerberos and the access policy can be enforced by the cloud service provider. Thus the end to end security procedure is dependent with GnuPG user process in the system node and access policy from Kerberos change the behavior of the user policies with the SELinux.

Benefits of the Proposed Framework over CTES

The proposed CTES framework with inbuilt access control policy has the following benefits over the existing CTES model.

1. Kerberos authentication database at KDC maintains the user ID and public key certificates of all the users in the cloud environment. Superhost maintains the record of all clients and Application Servers and provides the access service to the authenticated client with Kerberos protocol.
2. With the use of ID and IP address of Application Server, client is granted directly to access the Application Server.
3. Client requests SuperHost to get the list of services for which client is authorized to communicate with the services and resources in the Cloud Environment. The authentication work load of each client is balanced with in the client system itself and minimized the server workload to enhance the performance of CTES model.
4. The Web services should comply with GnuPG based implementation of Web Application Servers. Thus the client should use GnuPG service to encrypt or decrypt the data for sending or receiving it. By using these end to end security communication, the intermediate nodes does not have the capability to access those data. Thus it is more effective than any other methodology.
5. This proposed design provides security in end to end form with the security procedures of confidentiality, non repudiation, authentication and integrity getting initiated at client end and applicable till Application server. Thus it enhances the CTES model by providing end to end security environment. This end to end security procedure is dependent with GnuPG user process in the system node and access policy from Kerberos change the behavior of the user policies with the SELinux.

CONCLUSION

The limitations or issues with CTES model is briefly explained in this paper along with the messages involved in the process of user authentication and authorization for Cloud Services and resources. A Modified Kerberos style authentication and authorization has been proposed through a collaborative approach to maintain the trust, each node is required to register itself to SuperHost which is a trustworthy system with highest trust level. Registering of a node does not mean that it is authorized to obtain all the services and resources in the Cloud Environment. A client has to authenticate itself with KDC to access the list of services and resource based on the access control policy which is enforced by the cloud service provider through Kerberos Protocol. A SuperHost system keeps the track of all clients and regulates their uncontrolled behavior. The end to end security procedure is dependent with GnuPG user process in the system node and access policy from Kerberos change the behavior of the user policies with the SELinux. Moreover the client private key is available in the client system itself and so the only responsibility of client is to decrypt the data. It is more secure than the traditional authentication and access control information about the authenticating user will be enforced into the client system directly.

REFERENCES

[1] Sanjeev kumar pippal, Aruna kumari & Dharmender singh kushwaha. (2011). CTES based Secure approach for Authentication and Authorization of Resource and Service in Clouds. *International Conference on Computer & Communication Technology (ICCCT),* 444-449.

[2] Aruna kumari, Shakti mishra & Kushwaha. (2010). A New Collaborative Trust Enhanced Security Model for Distributed System. *International Journal of Computer Applications*, 1(26), 117-123.

[3] William stallings (2006). *Cryptography and network security principles and practices*. (4th ed ed.). Pearson Prentice Hall.

[4] Kamarudin shafinah & Mohammad mohd ikram (2011). File Security based on Pretty Good Privacy (PGP) Conce. *Computer and Information Science*, 4(4), 10-28.

[5] Michael louie loria. (2014). *Pretty Good Privacy*. Retrieved 21 August, 2017, from http://slidedeck.io/michaellouieloria/pgp

[6] Mohamed S. Nabi, M. L. Mat Kiah & A. A. Zaidan (2013). Suitability of Adopting S/MIME and OpenPGP Email Messages Protocol to Secure Electronic Medical Records. *Second International Conference on Future Generation Communication Technologies*, 93-97.

[7] Ashok Kumar J & Gopinath Ganapathy. (2017). A Modified Approach for Kerberos Authentication Protocol with Secret Image by using Visual Cryptography. *International Journal of Applied Engineering Research*, 12(21), 11218-11223.

[8] Rakesh shukla, Hari om prakash & R phanibhusan. (2016). Open PGP based Secure Web Email. *3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 735-738.

[9] Qassim bani hani & Julius p ditcher. (2017). Stand-Out Segmentation Access Control for Cloud Outsourced Data. *IEEE International Conference on Edge Computing*, 210-215.

[10] Young sic jeong & shin gak kang. (2013). E-mail encryption methods and lawful interception methods of it. *15th International Conference on Advanced Communications Technology (ICACT)* , 29-32.

[11]