Ensuring data storage security in cloud computing with effect of kerberos

Mehdi Hojabri
Email:{Hozhabri64@gmail.com}

Abstract

Cloud computing as we know is envisioned of the next-generation technology of IT industry. In other method is Internet based technology where the users can subscribe high quality of services from data and software that resides in the remote servers. This make many advantage and drawback for the users to create and store data in the cloud servers thereby utilizing fewer resource in client syste and the other hand management of the data and software may not be fully trustworthy and accurate which possesses many security challenges. So the security is important aspect of service quality. In this article,we concentrate on cloud data storage security by the implementation of kerberos authentication service.We define the kerberos for create the ticket and granting ticket for each user.so to make the more focus on user we made more secure.

Key words*:kerberos service, cloud service provider ,authentication service.*

1. Introduction

As cloud computing continues to thrive and as more and more enterprises penetrate the cloud, security becomes a further pressing issue. Several trends are opening up the era of cloud computing,which is an internet base development and use of computer technology.the ever cheaper and more powerful processors,together with the software as a service SaaS computing architecture ,are transforming data center into pools of computing service on a huge scale. In this paper i survey the Kerberos (a.k.a.: Cerberus) effect in cloud computing server. Kerberos uses strong encryption and a complex ticket-granting algorithm to authenticate users on a network. Also of interest to many of users, Kerberos has the ability to distribute "session keys" to allow encrypted data streams over an IP network.each user for connecting to the cloud

at the first shoud make the profile and user ID. After that it must get the password.and also the information of all participating user such as User ID, hashed password will save in the large Data Base for more secure. All user are register with the kerberos server. In this method each user want connect to the cloud server at the first time he or she logs on to workstation and:

A. send the Request for ticket granting ticket to the As.
B. As verifies user's access right in data base,create ticket-granting ticket and session key. Result are encrypted using key derived from user password.
C. user will send the request cloud service granting ticket to TGS.
D. the TGS will send the Ticket+session key to the user.(it execute one per type of service).
E. Workstation sends ticket and authenticator to cloud server provider.
F. server verifies ticket and authenticator match,then grant access to service.
G. In this paper i have try to assume each user for connecting and utilize the cloud server must create the profile and apply some private information for more secure.

2. Problem statement

A representation network architecture for cloud data storage with effect of kerberos is illustrated in Figure 1.

six different network entities can be identified as follows:

2.1. user:user,who have data to be stored in the cloud and rely on the cloud for data computation,consist of both individual consumer and organization. and want access to cloud server for doing job with effect of kerberos service.

2.2. Cloud service provider (CSP): Cloud service providers offer cloud solutions, like Google Apps, that are delivered electronically over the internet. Unlike a managed service provider, cloud service providers do not sell or install hardware – everything they offer is stored online and accessible securely from anywhere. There are many advantages to working with a cloud service provider like Cloud Sherpas when switching from your old email and collaboration software.

2.3. Kerberos operation: Kerberos is an authentication protocol for trusted hosts on untrusted networks. The Kerberos protocol is designed to provide reliable authentication over open and insecure network where communicate's between

the hosts belonging to it may be intercepted. The following requirement for kerberos is: Secure-Reliable-Transparent-Scalable

2.4. Authentication service AS: an authentication service that know the password of all user and stores these in a centralized database. in addition,the AS shares a unique secret key with each server.

2.5. Tickets granting service (TGS): TGS provide and issue tickets to user who have been authentication to AS.

2.6. Data Base: The kerberos server must have the user ID(UID) and hashed password of all participating user in the data base.All user are register with the kerberos server. It make more security in cloud server.

3. Ensuring cloud data storage with effect of kerberos

In cloud data storage system,users store their data in the cloud and for accessing must refer to cloud server provider. Thus the correctness of the user being refer to the distributed cloud server must be guaranteed. The data stored in the cloud may be frequently updated with user including: insertion , deletion , modification,appending,reorderi ng,etc.To ensure this updating is under correctness user is important. So in this paper we introduce one model based on kerberos. In this model each user for gain the cloud server must be register in the data base and after added to the data base it can get some qualification and after that get the cloud server.and...

In this scenario:

A. the client logs on the workstation and send the requests access a ticket-granting ticket on behalf of the user by sending its user's ID to the AS,together with TGS ID,indicating a request to use the TGS service.

B. The AS responds with a ticket that is encrypted with a key that is derived from user password.When this response arrives at the client,the client prompts the user for his or her password,generates the key,and attempt decrypt the incoming message.if the correct password is supplied,the ticket is successfully recovered. Because only the correct user should know the password,only the correct user can recover the ticket. Thus,we have used the password to obtain credentials from kerberos without having to transmit the password in plaintext. The ticket itself consist of

the ID and network address of the user,and the ID of the TGS. This corresponds to the first scenario.This is that this ticket can be used by the client to request multiple cloud service granting tickets.so the ticket granting ticket is to be reusable. However,we do not wish an opponent capture the ticket and waits until the user has logged off his or her workstation. The opponent either gain access to that work station or configure his workstation with the same network address as that of the victim.The ticket include a timestamp ,indicating the data and time for which the ticket was issued,and a lifetime, indicating the length of time for which the ticket is valid. Thus,the client know has a reusable ticket and need not bother the user for a password for each new service request.

A. The client request a service-granting ticket on behalf of the user. For this purpose,the client transmits a message to the TGS containing the user's ID,the ID of the desire cloud service,and the ticket-granting ticket.

B. The TGS decrypt the incoming ticket and verifies the success of the decryption by the presence of its ID. It check to make sure that the lifetime has not expired.Then it compares the user ID and network address with the incoming information to authenticate the user. If the uses is permitted access to V,the TGS issues a ticket to grant access to the requested cloud service provider. The service-granting provider ticket has the same structure as the ticket-granting ticket.Indeed,because the TGS is a server,we would expect that the same elements are needed to authenticate a client to the TGS and to authenticate a client to an application server. Again,the ticket contain a timestamp and lifetime. If the user wants access to the same cloud service at a later time,the client can simply use the previously acquired service-granting ticket and need not bother the user for a password. Note that the ticket is encrypted with a secret key(K_v) known only to the TGS and the server,preventing alteration. Finally,with a particular cloud service –granting ticket,the client can gain

access to the corresponding service with next step.

C. The user request access to cloud service on behalf of the user. For this purpose the client transmits a message to the server containing the user's ID and the cloud service granting ticket,The server authentication by using the contents of the ticket.

The table 1 shows how to implement this scenario .

(A)Authentication Service Exchange:to obtain ticket-granting Ticket (1) C ⟶ AS: ID_c ID_{tgs} TS_1 (2) AS ⟶ C: $Ek_c[k_{c,tgs}$ ID_{tgs} TS_2 $lifetime_2$ $Ticket_{tgs}$ $Ticket_{tgs}=Ek_{tgs}[k_{c,tgs}$ ID_c AD_c ID_{tgs} TS_2 $Lifetime_2]$
(B)Ticket-granting cloud service Exchange:to obtain cloud service-granting ticket (3) C ⟶ TGS:ID_v $Ticket_{tgs}$ $Authenticator_c$ (4) TGS ⟶ C:$Ek_{c,tgs}$ $[k_{c,v}$ ID_v TS_4 $Ticket_v]$ $Ticket_{tgs}=Ek_{tgs}[k_{c,tgs}$ ID_c AD_c ID_{tgs} TS_2 $Lifetime_2]$ $Ticket_v=Ek_v[k_{c,v}$ ID_c AD_c ID_v TS_4 $Lifetime_4]$ $Authenticator_c=EK_{c,tgs}[$ ID_c AD_c $TS_3]$
(C)client/Server Authentication Exchange:to obtain cloud service (5) C ⟶ K: $Ticket_v$ $Authenticator_c$ (6) K ⟶ C:$Ek_{c,v}[TS_5+1]$ (for mutual authentication) $Ticket_v=Ek_v$ $[k_{c,v}$ ID_c AD_c ID_v TS_4 $Lifetime_4]$ $Authenticator_c=Ek_{c,v}$ $[ID_c$ AD_c $TS_5]$

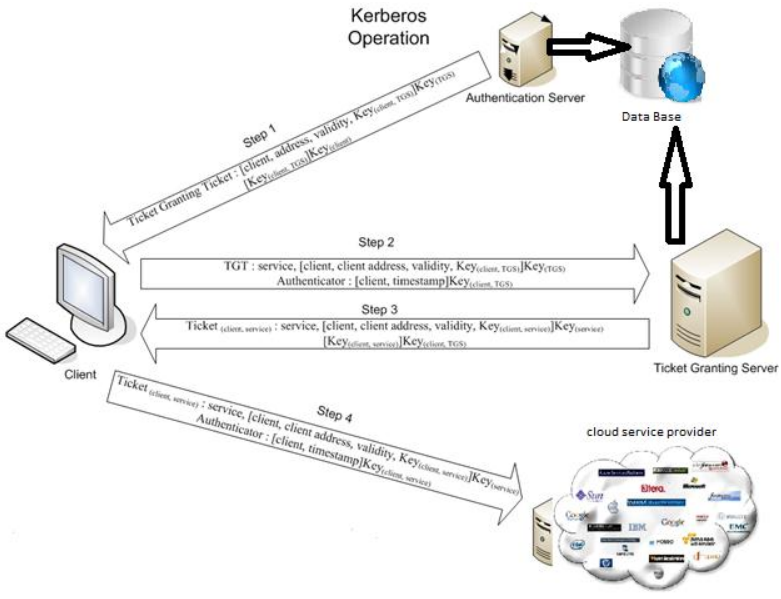

Fig.1:Cloud data storage architecture

4. Conclusion

In this paper, we investigated the problem of data security in cloud data storage, which is essentially a distributed storage system. To ensure the correctness of users' data in cloud data storage, and correctness of users who can access to the cloud server , we proposed an effective and flexible distributed scheme with explicit dynamic data support, including kerberos and authentication service. Kerberos provides a centralize authentication service whose function is to authenticate user to cloud server and cloud server to user. any user to access the cloud server first should make the profile and password.Then it can use the cloud server with gain the qualify. As we know the unique attribute of network is security.So for making more secure network we must make the way for controlling the cloud system and storing the information of user's. we would like for cloud servers to be able

to restrict access to authorized users and to be able authenticate request for service.As we know in an un protected network environment,any client can apply to any cloud server for service but kerberos operation with make use of DES,in a rather elaborate protocol,to provide the authenticaion service.In my opinion this model is novel model in era of cloud domain.

5. References

[1]MILL88 Miller,S,;Neuman,B.; Schiller,j.;and Saltzer,j."Kerberos Authentication and authorization System."Section E.2.1,Project Athena Technical plan,M.I.T.Project Athena,Cambridge,MA.27 October 1998.

[2]STE188 Steiner,j.:Neuman,C.; and Schiller,j."Kerberos:An Authentication Service for Open Networked Systems."Proceeding of the Winter 1988 USENIX Conference,February 1988.

[3]KOHL89 Kohl,j."The Use of Encryption in Kerberos forNetworkAuthentication."Proceed ing. Crypto 96, August 1996;published by Springer-Verlag.

[4]KOHL94 Kohl,j.;Neuman,B.;and Ts'o.T."The Evolution of the Kerberos Authentication Service."In Brazier.F.,andJohansen, D.Distributed Open Systems.Los Alamitos,CA:IEEE Computer Society Press,1994.Available at http://Web.mitedu/kerberos/www.papers,html.

[5] Amazon.com, "Amazon Web Services (AWS)," Online at http://aws.amazon.com, 2008.

[6] N. Gohring, "Amazon's S3 down for several hours," Online at http://www.pcworld.com/businesscenter/article/142549/amazons s3 down for several hours.html, 2008.

[7] A. Juels and J. Burton S. Kaliski, "PORs: Proofs of Retrievability for Large Files," Proc. of CCS '07, pp. 584–597, 2007.

[8] H. Shacham and B. Waters, "Compact Proofs of Retrievability," Proc.of Asiacrypt '08, Dec. 2008.

[9] K. D. Bowers, A. Juels, and A. Oprea, "Proofs of Retrievability: Theory and Implementation," Cryptology ePrint Archive, Report 2008/175,2008,http://eprint.iacr.org/ .