Distributed Authentication in the Cloud Computing Environment

Yanzhu Liu, Zhi Li(✉), and Yuxia Sun

School of Computer and Communication Engineering,
University of Science and Technology, Beijing 100083, China
liuyanzhu213@163.com

Abstract. The cloud computing is considered as the next-generation of IT technology that can provide various elastic and scalable IT services in pay-as-you-go. This technology has been used by worldwide companies to improve their business performance. Therefore, authentication of both clients and services is a significant issue for the trust and security of the cloud computing. At present, to protect the security of the cloud, many ideas have been proposed. In this paper, we present DSA protocol concentrated on authentication of clients. It is an improved protocol based on Kerberos, which prevents password guessing attack by using dynamic session key. We also solved the problem of service availability by introducing two additional messages in the scheme.

Keywords: Cloud computing · Distributed authentication · Password guessing attack · Kerberos · Ticket

1 Introduction

With the development of Internet technology, the number of Internet users and data has increased dramatically. The existing computer systems and network resources cannot meet users' requirements. Integration and optimization of the resource have become the inevitable trend of the future development of network. Conveniently, the concept of cloud computing is proposed, the lives of people quietly entered the era of cloud [1–3]. In cloud computing, all kinds of information are within reach and resources service can be used on-demand, anywhere, at any time [4].

Cloud computing became popular in 2007, to which the first entry in the English Wikipedia from March 3, 2007 attests, which, again significantly, contained a reference to utility computing [5]. Cloud computing is a new computing model based on distributed system, parallel computing and grid computing [6]. It is a new sharing infrastructure, which provides users with data storage, and network services in a large distributed environment [7]. Based on the technology of Internet and distributed computing, by integrating computing, storage and bandwidth resources into a resource pool, cloud computing provides users services in a dynamic and on-demand way [8, 9]. This new computing model has brought a dramatic change for the IT industry.

According to IDC's report, by 2016, 40 percent of enterprises will make proof of independent security testing a precondition for using any type of cloud service. At year-end 2016, more than 50 percent of Global 1,000 companies will have stored

G. Wang et al. (Eds.): ICA3PP 2015 Workshops, LNCS 9532, pp. 810–818, 2015.
DOI: 10.1007/978-3-319-27161-3_74

customer-sensitive data in the public cloud [10]. Due to the huge advantage of cloud computing, majority of companies have a great enthusiasm on cloud computing services. However, one after another accidents not only cause irreparable loss for users, but also hinder the development of cloud computing industry. The first is emergence of Amazon cloud computing server that interrupts the services [11]. Soon, Google leaked the users' personal information [12]. Then, Sony PlayStation Service network was hacked, about 77 million users' personal information were stolen [13]. Right now, with more and more personal and corporate information being stored in the cloud, users may have worried about the safety of personal information. Hence, the security has become an important issue in the field of cloud computing.

To ensure the security of resources and services has become the main goal, while the core of security mechanism is the authentication. Authentication protocol can ensure a real and secure communications, prevent the identity of the participants, and also prevent illegal tampering and other malicious attacks. At present, many companies use Kerberos to authenticate users, who need to pass the Kerberos authentication for each application. However, there are some limitations in the Kerberos authentication, and the authentication security has room for improvement. In view of the potential threat of session key in the process of client and application server communication, this paper proposes an extensible authentication model DSA. By adding a nonvolatile memory to store key chain in the client and the application server, it can guarantee to avoid password guessing attack to some degree.

2 Related Work

Kerberos is an authentication mechanism that can be used to authenticate user in the cloud computing environment. By using Kerberos authentication protocol, a user can authenticate itself to multiple application servers with the tickets distributed by Kerberos authentication center during a certain period. Many schemes have been proposed to prevent vulnerabilities and threats in Kerberos authentication protocol. Figure 1 shows the basic Kerberos architecture.

Al-Janabi et al. [14] implemented public-key cryptography extension specifications to the traditional Kerberos standard which incorporated public-key infrastructure (PKI) into the scope of underlying systems trusted by Kerberos. In [15], a model of Kerberos Protocol Version 4 was verified to find problems with respect to the replay attack. The presence of Intruder in the system was considered and the possible replay attack between various entities was also found out. Dua et al. [16] used triple password scheme to prevent replay attack and password guessing attack. In their research, Authentication Server stored three passwords. Authentication Server sends two passwords which were encrypted with the secret key shared between Authentication server and Ticket Granting server to Ticket Granting Server. Similarly Ticket Granting Server sends one password to Application Server. Meanwhile service granting ticket was transferred to users by encrypting it with the password that TGS had just received from AS which help to prevent replay attack. In [17], the process of Kerberos authentication protocol was analyzed. The dynamic password was used to improve the encryption

security during the process of interaction between the client and Kerberos key distribution center. By using Diffie-Hellman key, algorithm passwords were securely exchanged. Du et al. [18] presented to use dynamic password and one-time public key to improve the Kerberos protocol. The security of the session key and the password were considered. It made the protocol to improve the aspects of the resisting password guessing attack and replay attack. For mobile agent environment, Kandil and Atwan [19] introduced novel efficient and light security framework based on Kerberos system. By using 2-layer software that accomplishes the work of the hardware component, the framework could reduce the usual overhead resulting inside the Kerberos system.

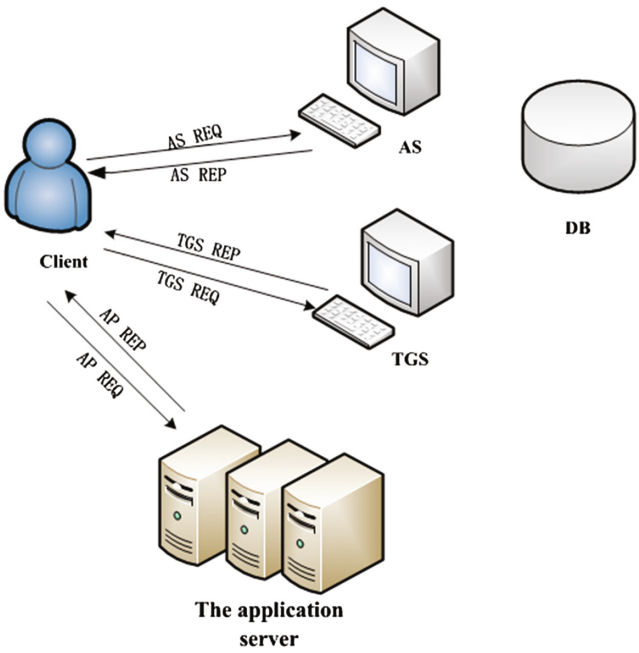

Fig. 1. Basic Kerberos architecture

In this paper, a formal model of DSA using a non-volatile memory is presented. The full schema of protocol dialogue is presented in next section and formal modeling of its operation will be modeled in future sections.

3 The Design of DSA Protocol

In this part, we propose a distributed service authentication (DSA) protocol in cloud. The protocol could realize authentication before client apply for services in the cloud.

3.1 System Model

The model is divided into three parts: client, application server and authentication server.

1. **The client.** The client has two main functions. Firstly, it listens the registration from client. Secondly, client can send requests to KDC, receive the feedback information and store them.
2. **The application server.** The main function of the application server is to monitor the request service from client and provide the service to client. Agent A would provide client with index of servers.
3. **The authentication server.** The authentication server has two key functions. First, it could monitor the requests of AS and TGS, meanwhile monitor the registration information from the client. Second, it could modify the information for registered client and authorize the client. The core algorithm of authentication service is AS authentication algorithm and TGS authentication algorithm.

3.2 The Procession of Authentication

The DSA model involves the following terms:

1. Client, can be the running processes or the ordinary users.
2. Server, application server. An entity provides service to the users.
3. TGS (Ticket Granting Server), issuing a ticket to the server. The users can show their identity by using the ticket to the application server.
4. AS (Authentication Server), an authentication server. Issue ticket to the users, by which the users can use to show their identity to TGS.
5. TGT (Ticket with Granting Ticket), client uses TGT issued by AS to prove its identity to TGS.
6. ST (Service Ticket), client use ST which is issued by TGS to prove its identity.

Now we will introduce DSA protocol from four aspects: request the ticket-granting ticket *TGT*, request the service-granting ticket *ST*, request the service index from agent and request the service from application server.

The procession of authentication of DSA protocol is shown in Fig. 2. The relevant symbols are shown in Table 1.

Table 1. Key notations

Notation	Explanation	Notation	Explanation
ID_C	Identity of client C	$K_{C,A}$	Key between client C and agent A
T_{S1}	Timestamp	AD_A	Network address of agent A
P_C	Password of client C	TGT	Ticket generated by AS
TL_C	Trust level of client C	ST	Ticket generated by TGS

(*Continued*)

Table 1. (*Continued*)

Notation	Explanation	Notation	Explanation
$K_{C,AS}$	Password between client C and AS	$K_{A,TGS}$	Session key between TGS and agent A
AD_C	Network address of client C	T_{S4}	Timestamp
AD_{TGS}	Network address of Ticket server	T_{S5}	Timestamp in $Auth_{C2}$
$K_{TGS,AS}$	Key between TGS and AS	TGT	A ticket provided by AS
$Lifetime_1$	Survival time of message m1	S_{List}	A service lists provided by agent A.
$Lifetime_3$	Survival time of ST	T_{S6}	Timestamp
$Lifetime_2$	Survival time of TGT	T_{S7}	Timestamp
K_i	Session key between application server S and client C.	S_c	The service selected by client C.
T_{S2}	Timestamp	SC_{List}	A server lists selected by client C.
T_{S3}	The generation time of $Auth_{C1}$	T_{S8}	Timestamp
$Auth_{C1}$	Client generate identification code to verify TGT	seq	Record the rank of message between server S and client C.
$K_{C,TGS}$	Session key between TGS and client C	R_{ES}	Service response from application server S.

1. Request the ticket-granting ticket TGT

   ①m1: C→AS=[$ID_C \parallel T_{S1} \parallel Lifetime_1$];
   ②m2: AS→SH=[ID_C];
   ③m3: SH→AS=[$P_C \parallel AD_C \parallel TL_C$];
   ④m4: AS → C=E ($K_{C,AS} \|[Ticket_{C,TGS} \parallel AD_{TGS} \parallel T_{S2} \parallel Lifetime_2 \parallel TGT]$);

   $$TGT = \mathrm{E}\ (K_{TGS,AS}\|[\,Ticket_{C,TGS}\,||\,ID_C\,||\,AD_C\,||\,AD_{TGS}\,||\,T_{S2}\,||\,Lifetime_2\,||\,TL_C\,])$$

In this part, the client C requests a ticket-granting ticket by sending its identity and password to the AS, indicating a request to use the TGS service. $K_{C,TGS}$ is the session key between client C and TGS.

2. Request the service-granting ticket ST

   ①m5: C→TGS=[$TGT\|Auth_{C1}$];
   ②m6: TGS→C=E ($Ticket_{C,TGS} \|[K_{C,A}\|AD_A\|T_{S4}\|ST]$)
   $Auth_{C1}$=E ($Ticket_{C,TGS} \|[ID_C\|AD_C\|T_{S3}]$),

   $$ST = E\ (K_{A,TGS}\|[\,K_{C,A}\,||\,ID_C\,||\,AD_C\,||\,AD_A\,||\,T_{S4}\,||\,Lifetime_3\,||\,TL_C\,]).$$

Before the client C accesses to the service of server S, the first check is to have a service-granting ticket ST. If not, the client C should send a request message m5 to TGS for authorization.

3. Request the service index from agent A

①m7: C→A=[$ST \| Auth_{C2}$];
②m8: A→C=E ($K_{C,A} \| [T_{S6} \| S_{List}]$);
③m9: C→A=E ($K_{C,A} \| [S_c \| T_{S7}]$);
④m10: A→C=E($K_{C,A} \| [T_{S8} \| SC_{List}]$);

$$Auth_{C2} = E\left(K_{C,A} \| [ID_C \| AD_C \| T_{S5} \| Lifetime_3]\right).$$

Before the client C requests a service from application server, it must obtain a list of authorized services available from the agent A at first. Then, agent A decrypts the message m9 and sends back to the service index to the client C, according to the information selected by the client.

4. Request the service from application server

①m11: C→S=E ($K_i \| [ID_C \| S_c]$);
②m12: S→C=E ($K_{i+1} \| [R_{ES}]$).

When client C requests the service, it sends the message m11 to the corresponding application server S, which will provide the client with the corresponding service. In massage m11 and m12, we use K_i as the session key between client C and Server S. When client C sends seq = i request message with K_i encrypted to server S and server S uses K_i to decrypt it, then server S uses K_{i+1} to encrypt the message and the client C uses K_{i+1} to decrypt the response message. In the way, K_{i+1} will be a new session key

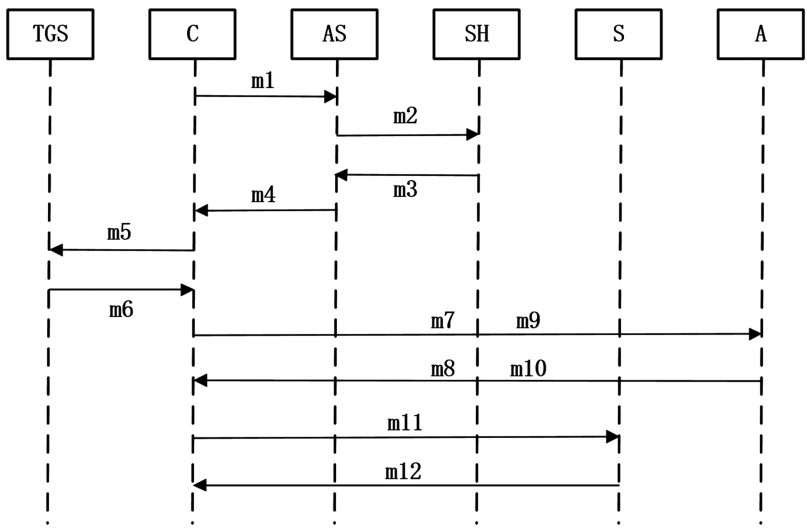

Fig. 2. Authentication process of DSA

in next request. In this paper, we apply the dynamic session key to increase the difficulty of password guessing attack and improve the security of the protocol.

4 Performance Analysis

In this part, we will compare the performance of DSA with Kerberos protocol and analyzed its security performance. The result shows that DSA authentication protocol can increase the ability to resist password guessing attack.

1. Service availability analysis
   The tickets of DSA protocol is an improvement of Kerberos. Kerberos need 6 messages and DSA protocol has 12 messages. In the six additional messages, two of them are used to access the service, these two messages solved the problem of service availability in Kerberos protocol.
2. Efficiency analysis
   Kerberos use a database to store all passwords and identities of users. Different from Kerberos, we use a non-volatile file to store the key in this scheme. Therefore, the DSA is more secure and efficient.

   DSA protocol uses a ticket data structure in the process of authentication. It can safely send the result of authentication and session key to the application server and can be reused in its lifetime. At the same time, it reduces the using frequency of password and the workload of the AS server. Therefore, DSA protocol could reduce overhead and improve the efficiency of authentication.
3. Security analysis
   For the security of DSA protocol, we mainly analyze password guessing attack. Since Kerberos cannot resist password guessing attack, DSA protocol uses a monitor to solve the problem. In DSA protocol, AS does not get the client's password and position information from the client indirectly, but from client's monitor which is responsible for validation. Aiming at the potential threat in the session key between client C and server S, we add a non-volatile memory to store key chain between client C and server S. The key chain K_i is used to take the place of $Ticket_{C,S}$ to encrypt the message. We use K_i as the session key between client C and Server S. When client C sends seq = i request message with K_i encrypted to server S and server S uses K_i to decrypt it, then server S uses K_{i+1} to encrypt the message and the client C uses K_{i+1} to decrypt the response message. In the way, K_{i+1} will be a new session key in next request. We use the dynamic session key in the scheme to increase the difficulty of password guessing attacks and improve the security of the protocol.

We also compared the performance of DSA with Kerberos (versions 4 and 5) in other aspects. The results are shown in Table 2 as follows.

Table 2. Performance comparison

Items	Kerberos V4	Kerberos V5	DSA
Single sign-on	Yes	Yes	Yes
Resistance of password guessing attack	No	No	Yes
Database used for storing key	Yes	Yes	No
Service registration	No	No	Yes
Symmetric key	Yes	Yes	Yes
Dependence of operating system	Yes	Yes	No
Mutual authentication between entities	No	Yes	Yes
Access authorization	No	No	Yes
Resistance of denial of service attack	No	No	No

5 Conclusion

In cloud computing environment, security is essential in all aspects of fields. Authentication and authorization is the first step for users to enjoy the service of cloud. Kerberos provides a third party authentication, by which client can authenticate itself to multiple servers using its password. However, it is not feasible when facing password guessing attack. With respect to the problems existing in Kerberos, this paper puts forward a distributed authentication (DSA) protocol in the cloud. The system could realize authentication when the client apply for services in the cloud. We apply a dynamic session key into the protocol, which increases the difficulty of the password guessing attack and improves the security of the protocol. Then we also compared DSA with Kerberos in efficiency and security. The result shows DSA protocol improved the performance of Kerberos. In protocol of DSA, the application server is available and the capacity of resistance password guessing attack is enhanced. In the future, we will further study the resistance in replay attack of Kerberos.

References

1. Armbrust, M., Fox, A., Griffith, R., et al.: A view of cloud computing. Commun. ACM **53**(4), 50–58 (2010)
2. Marinos, A., Briscoe, G.: Community cloud computing. In: Jaatun, M.G., Zhao, G., Rong, C. (eds.) Cloud Computing. LNCS, vol. 5931, pp. 472–484. Springer, Heidelberg (2009)
3. Velte, T., Velte, A., Elsenpeter, R.: Cloud Computing, a Practical Approach. McGraw-Hill, Inc., New York (2009)
4. Qian, L., Luo, Z., Du, Y., Guo, L.: Cloud computing: an overview. In: Jaatun, M.G., Zhao, G., Rong, C. (eds.) Cloud Computing. LNCS, vol. 5931, pp. 626–631. Springer, Heidelberg (2009)
5. Boss, G., Malladi, P., Quan, D., et al.: Cloud computing. IBM white paper, vol. 1 (2007)
6. Wei, L., Zhu, H., Cao, Z., et al.: Security and privacy for storage and computation in cloud computing. Inf. Sci. **258**, 371–386 (2014)
7. Feng, D.G., Zhang, M., Zhang, Y., et al.: Study on cloud computing security. J. Softw. **22** (1), 71–83 (2011)

8. So, K.: Cloud computing security issues and challenges. Int. J. Comput. Netw. **3**(5), 1–9 (2011)
9. Sabahi, F.: Cloud computing security threats and responses. In: 2011 IEEE 3rd International Conference on Communication Software and Networks (ICCSN), pp. 245–249. IEEE (2011)
10. Talon, C., Insights, I.D.C.E.: The impact of cloud computing on the development of intelligent buildings. CABA Intell. Integr. Build. Counc. (IIBC), pp. 1–29 (2013)
11. Zunnurhain, K., Vrbsky, S.: Security attacks and solutions in clouds. In: Proceedings of the 1st International Conference on Cloud Computing, pp. 145–156 (2010)
12. Malandrino, D., Petta, A., Scarano, V., et al.: Privacy awareness about information leakage: who knows what about me?. In: Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society, pp. 279–284. ACM (2013)
13. Quinn, B., Arthur, C.: PlayStation network hackers access data of 77 million users. Guardian (2011)
14. Al-Janabi, S.T.F., Rasheed, M.A.: Public-key cryptography enabled Kerberos authentication. In: Developments in E-systems Engineering (DeSE), pp. 209–214. IEEE (2011)
15. Mundra, P., Shukla, S., Sharma, M., et al.: Modeling and verification of Kerberos protocol using symbolic model verifier. In: 2011 International Conference on Communication Systems and Network Technologies (CSNT), pp. 651–654. IEEE (2011)
16. Dua, G., Gautam, N., Sharma, D., et al.: Replay attack prevention in Kerberos authentication protocol using triple password. Int. J. Comput. Netw. Commun. **5**(2), 59 (2013)
17. Wang, C., Feng, C.: Security analysis and improvement for Kerberos based on dynamic password and Diffie-Hellman algorithm. In: 2013 Fourth International Conference on Emerging Intelligent Data and Web Technologies (EIDWT), pp. 256–260. IEEE (2013)
18. Du, Y., Ning, H., Yang, P., et al.: Improvement of Kerberos protocol based on dynamic password and "One-time public key". In: 2014 10th International Conference on Natural Computation (ICNC), pp. 1020–1025. IEEE (2014)
19. Kandil, H., Atwan, A.: Mobile agents' authentication using a proposed light Kerberos system. In: 2014 9th International Conference on Informatics and Systems (INFOS), pp. CNs-39–CNs-45. IEEE (2014)