The Distributed Authentication Login Scheme

Joonseok Park

Inha University, School of Computer Science and Engineering,
Inchon, Republic of Korea

Summary
In distributed system environment, security of system is an important issue whose basis is authentication protocol. Authentication protocol requires a authentication server which will decide login of users, and the server will be the main target of attacks. In this paper, we present a distributed authentication protocol model, whose goals are to avoid one point of attack and to increase availability by replication without loosing degree of security.

Key words:
Authentication protocols, Kerberos, Network Security, Distributed System.

1. Introduction

In recent years, many organizations have shifted their computing facilities from central main frames to servers accessed from personal computer via inter- or intra-network. In this situation, the importance of secure communication in distributed system is universally acknowledged. For this reason, much effort has been invested into providing security services in a various network and operating system environment. Kerberos is one of the best-known efforts. Kerberos is the authentication and key distribution system that was developed at the Massachusetts Institute of Technology (MIT) as a part of Athena project [7], which was targeting LAN environment at first. As time goes on, there has been tension to move on global network system. As other system modifies to fit in the new environment, Kerberos also has made its evolution. However, it inherently keeps unsolvable problems which are single point of attack and single point of failure. Single point of failure problem can be solved by replicating server, but single point of attack has not solved by any system. Above problems become more venerable as network environment scale up.

In this paper we describe a new network security service model, which can solve above two problems. This model is implemented from basis of Kerberos, however, it is more secure and more reliable system than conventional Kerberos. New model (distributed authentication server) is adapting mechanism of RAID storage system to avoid single point of failure problem – one node keeps checksum of data. At the same time, we partition secure data share, not by granularity of entity, but by bit level, we can avoid single point of attack problem. In this manner, breaking one node of distributed authentication server does not expose all secure data in the network.

The contribution of this paper is captured as follows.

- We extend Kerberos Authentication scheme to enhance its securities.
- We provide new distributed authentication model which is fault-tolerant.

This paper is consists of 5 chapters. We present Kerberos authentication scheme briefly and some other related researches in chapter 2. Then, we will describe our new authentication model and system architecture in chapter 3 and numerical analysis in chapter 4, respectively. Chapter 5 is conclusions.

2. Background

In this chapter, we first describe Kerberos system, which is used as our base system. We list some technical problems of Kerberos which are the motivating issues of this paper. Then we introduce some other contemporary authentication scheme as related works.

2.1 Kerberos Authentication Scheme

Kerberos authentication and key distribution protocols are based on Needham and Schroeder protocol [5]. This is the most widely accepted authentication scheme nowadays. It uses single session key (symmetric key) to encryption and decryption. However, there are some changes to support according to the needs of computing and network environment. The examples are (1) the use of timestamps, (2) ticket granting service, and (3) inter-realm authentication.

The message exchanges are consists of three parts as shown in fig 1, the AS (authentication server) exchange -(1,2), the TGS (ticket granting server) exchange -(3,4), and the Application Server exchange -(5,6). With the TGT (ticket granting ticket) in second step (as shown in Fig1(a)) of communication – part of $T_{c,tgs}$ –Kerberos

Manuscript received October 5, 2007
Manuscript revised October 20, 2007

supports SSO(single-sign-on) services. The details Kerberos protocol was described by J.T.Kohl et al [9].

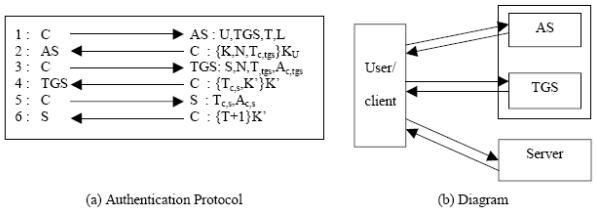

Fig. 1 Kerberos Authentication protocol

To be adapted to new network environment, recent version of Kerberos keeps its improvements. It provides the "realm-name identifier" in the Principal identifier to facilitate inter realm traverse and accesses. It improved modularity in encryption and supports individual network address support in the ticket exchange. When plain text is encrypted, the corresponding tag will be sent together. With the tag, recipient can recognize which encryption technique was used for encryption[5].

S.M Bellovin and M.Merritt pointed limitations of Kerberos system in their paper [8]. They listed remaining and some newly found problems in their draft version. The main concerns of it are the weakness against possible attack. Kerberos is weak in password guessing and dictionary attack and it is exposed to "verifiable" attack. Since Kerberos authentication system is stateless, it cannot verify fake users.

They recommended handshake authentication for basic login protocol, authentication of user to Kerberos server for initial exchange (two-party authentication with client and AS), and strong checksum to avoid Cut-and-Paste attack. They also recommended omitting protocol extensions that are not related to basic authentication.

2.2 Related Researches

The Sesame authentication system is an extension of Kerberos [6]. European Computer Manufacturer Association (ECMA) develops this protocol. Sesame user can communicate securely with Kerberos because it is compatible. The current version is version 4. There are two basic differences with Kerberos [16]. One is PAS (privilege attribute server) in authentication server component, and the other one is the authentication scheme which is using public key encryption.
Sesame need 8 message exchange steps. Six of them are identical with those of Kerberos. The additional two steps are related to PAS (privilege attribute certificate) that is responsible for “Role-Based access control”. SESAME enforces a role based access control scheme by using PAS. Other difference is public key encryption. When client sends login message to AS it sends client’s signature and public key. Then AS returns TGT (which is identical to that of Kerberos) with AS’s signature with its private key. With TGT client consults PAS before communicating application server and get access capability. Since AS authenticate client with their signature – encrypted with private key – it’s stronger than Kerberos against password guessing and dictionary attack.
By the way, there is extra cost to get principals public key. Client needs to get its public key and private key form CA (certificate Authority) which requires off-line message exchange.
KryptoKnight is the protocol used in IBM netSP[3]. The most unique feature in this protocol is that it provides secure service in any type of network environment, providing scalability and flexibility, which are noble requirements for current, inter network environment [11]. The KK protocol supports authentication (mutual or one-way), Key distribution, integrity protection (authentication of contents and origin of message), and secures end-session (authenticated session termination).KK solves most of limitations of Kerberos. It provides two-party authentication for first login, provide message integrity using MAC [14].

KK uses single basic protocol. However, KK protocol decides actual protocol to execute in the system depending on authentication type and key distribution sequence condition. Since KK software consists of separate modules organized in modularized layers and layered structure, it can provide this flexibility [11].
This structure also provides scalability. Depending on environment, KK authentication chooses proper message exchange scenario. For example, 2 party key distribution needs 3 message exchanges; however, the "A-B-KDC pull scenario" needs 6 messages exchange, which is combination of basic message exchange.

One outstanding factor in authentication part is use of MAC. The MAC is a cryptograghical checksum. This is one way hash function using secret key, which is share only by two communicating nodes. It is appended to the message at the source at a time when the message is assumed or known to be correct. The receiver authenticates that message by recomputing the MAC with shared secret key. By MAC receiver can check the authentication of message and message sender - MAC expression provides integrity of the data. The details of requirement of hash function and algorithm is explained in [13], [14].

SPX (Sphinx) is developed by DEC in the late of 80s. Although it attracted much attention throughout the network security community, it was aborted after version 2.4 in 1992. It uses both RSA public key encryption scheme and DES secret key encryption scheme [2]. Hence, it increases overall network security. In SPX, public key certificates are created by a trusted and highly secure CA. To reduce the impact of limited availability, the CA generates certificates that are stored in and distributed by online certificate distribution center. The public key encryption scheme is used to distribute a secret key in secure way.

SPX uses different credentials depending on whether a principal is acting as claimant or verifier. (client and server respectively in normal condition) Claimant credentials are valid for relatively time and verifier credentials are valid typically for long term period.
There are two protocols in SPX. One is credential initialization protocol and other is authentication exchange. The former is for initialization of client when user login, and the later for getting public key of communication destination and corresponding mutual authentication. The advantage and disadvantage of this system exactly holds those of public key encryption scheme.

Until now, we introduced different authentication schemes as motivations. Including these systems, most of network system has common problems related to security as follows:

- Single point of attack
- System cannot notice whether its security is broken or not until user notice a problem

In the next chapter we introduce distributed authentication scheme, which addresses above two non-trivial and important issues.

3. Distributed Authentication Server

Because of these reasons, if the server who manages secret data (for example, Authentication Server in Kerberos) lost its secrecy, before administrator or user notices the security failure, all data are exposed to attacker. To neutralize this problem, we introduce distributed secret managing model.

3.1 Overview

The goal of this model is to distribute secret data to several servers, instead keeping all data in one server, which is possible attack point. Secret data in the authentication system is pair of user or server id and corresponding secret data (usually password). We need to be careful to distribute secret data regard to following matters as important requirements.

- The distribution of secret data should not need a control manager.
- The node data independence - One exposed node of distributed data should not be clue to other node.

For the first issue, the central control manager will be a new attack point. Hence, we confront identical problem after distribution. If the second requirement, the node data independence, does not meet, then the success to open one node recursively exposes secrecy of dependent node that is not our aim.

To meet above requirements, we choose all distributed nodes keep user or server id, and distributed secret data in a bit by bit manner. For example, assume that secret data and nodes are 64 bits and 4 respectively, each node should keep 16 bit of secret data exclusively. If we need to generate one secret key, 4 secret data should be get together to build full 64 bit data.

3.2 System Architecture and Protocol

Distributing secret data in bit-by-bit method holds the second requirement we put in overview. However, each of data in node does not mean anything unless a certain node or controller collects all of corresponding data. The biggest problem of distributing data in this way is to collect data with keeping the first requirement - The distribution of secret data without control manager. To achieve that, we made distributed nodes collect a secret data with changing the node in role (of collecting) dynamically. For collecting message exchange we use public key encryption. For the convenience of explanation, we'll explain our model applying to Kerberos.

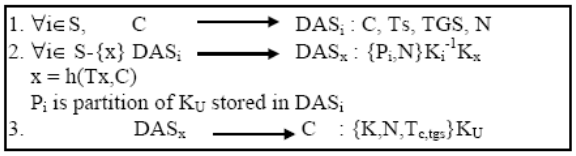

Fig. 2 Message exchange in Distributed Authentication System.

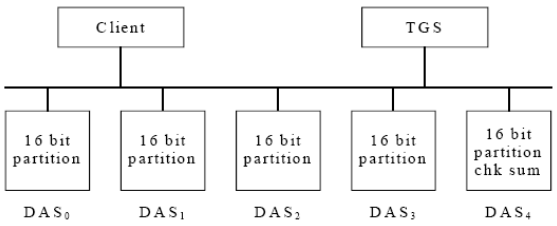

Fig. 3 Schemes for 64 bit secure data distribution

Our objective model consists of n distributed authentication servers (DAS) that have m bit secret data (password in Kerberos). Each DAS node keeps [m/n] bit of data exclusively. All DAS nodes know public key of each node.

Fig 2 shows how this works. When a user starts login, the client should multicasts to all DAS node-message contents. After receiving message, each of DAS nodes searches for data entry indexed by user id. At the same time, each node decides which node will collect all data based on requester id and timestamp in messages. To decide all nodes use publicly known hash function. Even when an attacker gets this function, it cannot get results, because the result of this function is dependent upon requester's timestamp.

After deciding the main collecting node for a secret key, each DAS node sends messages that are encrypted with its private key and target's public key. The target nodes collect all messages and finally extract corresponding password from encrypted message. Remaining service is same as Kerberos (2) - (6) $_2$ S = {0, 1, 2,...., n} where AS consists of n distributed nodes. DAS_n : store checksum, N : nounce for mutual Authentication in fig1 (a). In this way, security breakage of single DAS node exposes subset of messages, which was determined to be collected at that node. Since the decision is dynamic depends on timestamps, we can neutralize the risk. With n DAS node, one node failure makes all system down. To resolve this disadvantage we can use spirit of RAID (level 4).

As in example in the fig3, we use 5 nodes instead of 4, to collect 4 encrypted data. Four nodes will keep its partial secret key and one extra node keeps a checksum. It is helpful in two ways. The first benefit is failure-tolerance. In case one node's failure does not cause whole system failure. The second one is that with the checksum, system can detect active attack after attacker broke one of node's securities.

4. Analysis

The benefits of proposed authentication scheme are obvious - we can make more secure system. There are three advantages in the aspects of security.

- First, it can avoid single point of attack.
- Second, it provides better availability.
- Third, it can avoid whole exposure of secret data, unless attacker breaks all DAS nodes.

As mentioned at then end of chapter 2, the system cannot detect whether secrecy is broken or not. Only administrator or end user can detect the problems. In conventional scheme, since all secure data in single server, when user detects a problem, attacker already had gotten all secured information. However, in this model, exposure of one DAS node, only expose subset of whole messages (1/N). Hence, there is more possibility to detect security failure before loosing all secure data.

With these advantages, there is some trade-off for cost. We have to pay the cost for multiple authentication servers, more messages to login, and network support for multi-cast. However, there are enough advantages to make use of it for secure system.

4.1 Networks

This model use checksum node to support more availability. There is small possibility of checksum error. However, the possibility of checksum error in real situation is negligible. (As in fig4, checksum error occurs in a $1/2_{16}$) However, In this system, there is heavy load in the network because of multicast messages between node.
Also, failure of one node causes some failure of messages, which should be collected at that node. I remained this problem unsolved, but it's not serious problem at all. Since decision that which DAS node collects messages between node is based on Timestamp, later retry might be serviced at the other node.
We use level 4 RAID not to complicate other authentication message exchange protocol. That makes bottleneck in checksum node. We will not consider this point in this paper, however, it is possible to use level 5 RAID if we use more complicated hash function control.

4.2 Efficiency

Because of multicast and public key encryption in messages between DAS servers, efficiency of this model may not good. However, this system is definitely better than conventional Kerberos in the aspects of security. Moreover, we can improve efficiency with "Group key

agreement scheme"[16], in the situation which pubic key encryption is not important requirement.

We applied our model to a Kerberos. We believe that it is possible to apply similar changes to other authentication protocol. For example, since sesame uses same authentication structure as Kerberos it is possible to apply. At the same time, we have to consider the cost of message exchange which is importance because of network latency.

5. Conclusions

In this paper, we presented distributed authentication model. Although we only applied our scheme to Kerberos, we believe that it is possible to apply similar changes to other authentication protocol [16], which requires safe key at the early stage of protocol. For example, since sesame uses same authentication structure as Kerberos it is possible to apply. It is based on Kerberos model, but much better than Kerberos in several aspects. By distributing secure data efficiently, it provides stronger security and better availability. It avoids single point of attack problem and single point of failure problem at the same time, both of which are important issues in large scaled network environment and later of which was never solved before. Attacking one of nodes exposes secrecy 1/N in randomized pattern which can distribute risk.

References

[1] Roger M. Needham and Michael D. Schroeder. "Using encryption for authentication in large networks of computers" in Communications of the ACM, 21(12):993-999, December 1978.

[2] J.Tardo and K.Alagappan, "SPX: Global Authentication Using Public Key Certificates" in Proceedings of the IEEE Symposium on Security and Privacy, IEEE Computer Society Press, LA, CA 1991, pp.232-244

[3] R.Molva, G.Tsudik, E.Van Herreweghen, and S.Zatti, "KryptoKnight Authentication and Key distribution System," in Proceedings of 1992 European Symposium on Research in Computer Security, Toulouse, France, November 1992, pp-155-174, http://www.isi.edu/~gts/paps/mtvz92.ps.gz

[4] B.Lampson, M.Abadi, M.Burrows and E.Wobber "Authentication in Distributed Systems:Theory and Practice" in ACM Trans. Computer Systems10, 4(Noc.1992), pp 265-310

[5] B. Clifford Neuman and Theodore Ts'o." Kerberos: An authentication service for computer networks." in IEEE Communications Magazine, pages 33-38, September 1994.

[6] P.V. McMahon "SESAME V2 public key and Authentication Extensions to KERBEROS" in proceedings of the Internet Society Symposium on Network and Distributed System Security, February 1995, pp. 114-131.

[7] S.P. Miller, B.C. Neuman, J.I.Schiller, and J.K.Salzer Kerberos Authentication and Authorization System Section E.2.1 of Project Athena Plan 27 Oct 1988 "ftp://athenadist. mit.edu/pub/kerberos/doc/techplan.PS"

[8] S.M Bellovin, M.Merritt, Limitations of the Kerberos Authentication System, ACM SIGCOMM Computer Communication Review, October 1990

[9] John T. Kohl, B. Clifford Neuman, and Theodore Y. T'so, The Evolution of the Kerberos Authentication System. In Distributed Open Systems, pages 78-94. IEEE Computer Society Press, 1994.

[10] P.Janson, G.Tsudik, M.Yung, "Scalability and Flexibility in Authentication Services : The KryptoKnight Approach" IEEE Infocom 1997 , March 1997. http://www.isi.edu/~gts/paps/hkt94.ps.gz

[11] F.Piessens, B.De Decker, and P.Janson, "Interconnecting Domains with Hetrogeneous Key Distribution and Authentication Protocols", in Proceedings of the IEEE Symposium on Security and Privacy, IEEE Computer Society Press, Los Alamitos, CA, 1993, pp 66-79

[12] W.Stallings Network And Internetwork Security Principles and Practice, Prentice Hall.1995

[13] G. Tsudik "Message Authentication with One-Way Hash Functions" ACM computer Communication Review vol 22,1992. pp29-38. http://www.isi.edu/~gts/paps/t92.ps.gz

[14] M. Steiner, G. Tsudik and M. Waidner "CLIQUES: A New Approach to Group Key Agreement" IEEE International Conference on Distributed Computing Systems (ICDCS'98) ,May 1998.

[15] M.Vandenwauer, R.Govaerts, J.Vandewalle "Overview of Authentication Protocols" procedings 31st annual IEEE Carnahan Conference on Security Technology pp 108-113 1997 http://www.cosic.east.kuleuven.ac.be/sesame/papers/carnahan.pdf

[16] Commercial Encryption Export Controls. 1998 September. http://www.bxa.doc.gov/Encryption/

Joonseok Park received the B.S.in Mathematics from Sogang University in 1997, M.S. degrees and Ph.D in Computer Science from University of Southern California in 2000 and 2004, respectively. During 2004-2006, he stayed in System On Chip Laboratory of Samsung Electronics, Inc. as a senior Engineer. He now full-time lecturer of Inha Univerisity, School of Compesuter Science and Engineering.