Efficient One Time Password Authentication Schme for Distributed Environment

Aye Aye, Than Naing Soe
University of Computer Studies (Mandalay)
ayeayeucsm@gmail.com

Abstract

In today's world of distributed environment, password authentication is very important to protect customer's sensitive data over Internet. The mainly two types of password are static password and dynamic password. Static passwords are highly susceptible to cracking, because passwords used will get cached on the hard drives. One Time Password (dynamic password) is a solution to solve the problems of static password. One Time Password is a password which changes every time the user logs in. Even if the attacker gets the password, it is useless to the attacker. This system is based on Lamport's one time password scheme. Unlike Lamport's scheme, new password initialization will depend on life-time and number of login (n) decided by Register Center. Diffie-Hellman Key Exchange Protocol will be used in this system for password initialization and the key exchange process of user's password. This system will be developed in distributed environment with the aim of the users and servers' mutual authentication.

Keywords: One Time Password, Diffie-Hellman Key Exchange, mutual authentication, Register Center, distributed environment

1. Introduction

A one-time password (OTP) is a means of proving the identity of a user in which a password is only valid for a single authentication session or for a short time period. In existing OTP systems, the user's password are encrypted using shared secret key with server or are hashed before it sends to server for registration. In this case, if the secret key is compromised, lost or stolen, then an adversary may be able to get the user's password to access with server. To solve this problem, Diffie-Hellman key exchange will be used for initial password registration process. Moreover, the trusted register center RC will be used to get granted ticket for clients to connect with the server. This system will be designed in distributed environment. The paper is organized as follow. The related work of my research is presented in Section 2. In section 3, the password authentication system and weak points of traditional password authentication are pointed out. One time password and the security analysis of this scheme are discussed in section 4. In section 5, Diffie-Hellman key exchange protocol and the security of this protocol are explained. Kerberos distributed authentication scheme and the analysis of security are described in section 6. The proposed system is presented in section 7 with step by step phases. In section 8, security features of proposed system are analyzed. Concluding remarks and future work are described in section 9.

2. Related Work

Kenneth G. Paterson1 and Douglas Stebila [1] showed a general technique for building a secure one-time-PAKE protocol from any secure PAKE and allow for the secure use of pseudo randomly generated and time-dependent passwords. Keith A. Watson [2] explained the detail analysis of one time password and the ways to be secure one time password system. Mijin Kim, Byunghee Lee, Seungjoo Kim, and Dongho Won [3] resolved the security flaws found in Kuo-Lee's one time password scheme. DineiFlorˆ encio and Cormac Herley [4] implemented URRSA service. The user pre-encrypts his password using an assigned set of keys and these encryptions are sent as one-time passwords to URRSA decrypts before forwarding to the Login server. Minkyu Kim [5] discussed about Kerberos V and Public Key Kerberos Secuirty.

3. Password Authentication

As the numbers of people who use the internet are dynamically increasing, it necessitates various techniques to protect their privilege. The promising technology to solve this problem is password authentication. Password authentication is one of the simplest and the most convenient authentication mechanisms over insecure networks. Password authentication allows the legal users to use the resources of the remote systems. Password authentication is more frequently required in areas such as computer networks, wireless networks, remote login, operation systems, and database management systems.

3.1. Traditional Password Authentication

A simplest authentication approach is to store and maintain a password table including users' IDs and PWs in the remote server. When the users login, the remote server searches in the password table to check whether or not the submitted ID and PW match with those stored in the password table. If the ID and PW match the

corresponding pair stored in the server's password table, the user will be granted to access the server's facilities. Since the user's password is stored in plain-text form in password table, this approach is vulnerable to the revelation of the passwords. Moreover, an intruder can impersonate a legal user by stealing the user's ID and PW from the password table. Therefore, traditional password authentication system is vulnerable to stolen-verifi er attack.

3.2. One Time Password (OTP)

One Time Password is suggested by American scientist Leslie Lamport [1][4]. It is valid for only one session and useless to attackers even if the password is stolen. OTP reduces the vulnerability of the hacker sniffing network traffic, impersonation and modification. At each login process, the server verifies that the hash of what the user presents is same with the previously used password. Since each of the passwords is used only once it is of no use to an attacker. A further advantage is that the server needs to store only the previously used password, no password sequences. Thus even the database at the server contains nothing useful to the attacker.

4. Lamport's One Time Password

The third approach of one-time password devised by Leslie Lamport, the user and the system agree upon an original password, P_0 and a counter, n. The system calculates $h^n(P_0)$, where h^n means applying a hash function n times. In other words, $h^n(x)=h(h^{n-1}(x))$ $h^{n-1}(x)= h(h^{n-2}(x))$ … $h^2(x)= h(h(x))$ $h^1(x)= h(x)$. The system stores the identity of Alice, the value of n, and the value of $h^n(P0)$. Figure 1 shows how the user accesses the system the first time [8]. Figure 1 shows the design of Lamport's One Time Password Scheme.

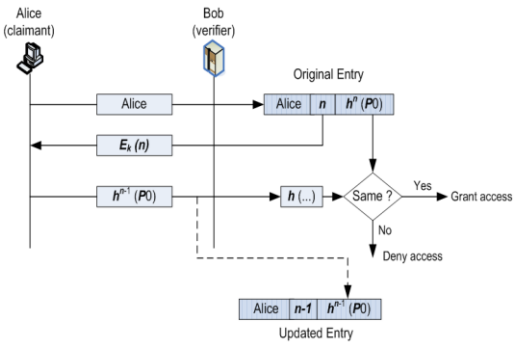


Figure 1. Lamport's One Time Password Authentication Scheme

When the system receives the response of the user, it applies the hash function to the value received to see if it matches the value stored in the entry. If there is a match, access is granted; otherwise it is denied. The system then decrements the value of n in the entry and replaces the old value of the password $h^n(P0)$with the new value $h^{n-1}(P_0)$. When the value becomes 0, the user can no longer access the system; everything must be set up again. For this reason, the value of n is normally chosen as a large number such as 1000.

4.1. Security Analysis of Lamport's One Time Password

In this section, the security analysis of Lamport's One Time Password authentication scheme will be discussed [2] [3]. Though one-time password offers stronger security than fixed password and prevents users from impersonation attack, it remains many security problems. It is insecure just using only the login number (n). It is useless computation if the number of authentication is uncertain (decision for number of login). It is vulnerable in key exchange of initial password (P_0). It is uncertainty of the decision of number of login (n).

5. Diffie-Hellman Key Exchange Protocol

The Diffe-Hellman algorithm, introduced by Whitfeld Diffe and Martin Hellman, was the first system to utilize "public-key" or "asymmetric" cryptographic keys. This algorithm overcomes the secret key exchange problem of "private-key" or "symmetric" key systems because asymmetric key management is much easier. The purpose of the algorithm is to enable two users to securely exchange a key that can then be used for subsequent encryption of messages. The algorithm itself is limited to the exchange of secret values. Figure 2 shows the detail processing of Diffie-Hellman key exchange protocol [6].

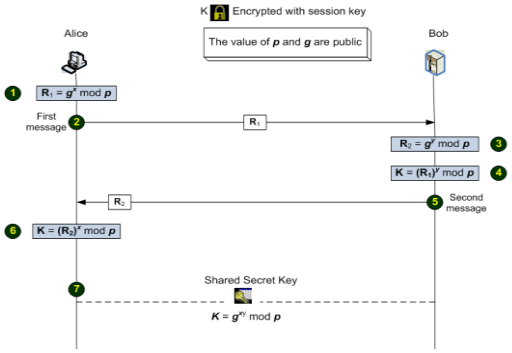


Figure 2. Diffie-Hellman key exchange protocol

5.1. Security Analysis of Diffie-Hellman Key Exchange Protocol

The advantages and disadvantages of Diffie-Hellman key exchange protocol are discussed in this section. It can solve the key exchange process of symmetric key system and overcome the slow connection for any sort of bulk encryption. Moreover, it can not only get security against eavesdroppers but also prevent man in

the middle attack. However, it complicates for managing public and private keys. It may be time complexity problem for generating of strong key which is large enough to resist brute-force attack.

6. Kerberos

Kerberos [7] is an authentication service developed as part of Project Athena at MIT. Kerberos is valuable for the authentication of distributed environment in which users at workstations wish to access services on servers distributed throughout the network. Kerberos supports for servers to be able to restrict access to authorized users and to be able to authenticate requests for service because an opponent can pretend to be a user.

6.1. Security Analysis of Kerberos

Although Kerberos solves some of the problems of authentication in an open network environment, problems remain. The benefits and shortcomings of Kerberos will be stand out [5].
It is no cleartext passwords on the network and has no client passwords on servers. Then, it can minimize password exposure on workstation. The shortcomings of Kerberos also exist. It uses symmetric key system which may causes secret key exchange problem for generation of Ticket, encryption and decryption process The initial request for registration is plaintext. It may appear lifetime too short or too long problem. It may be process overloading for using Ticket granting server (tgs). It has limited security advantages (it can't resist against spoofing attack and guessing attack)

7. Efficient One Time Password Authentication Scheme

This system intends for client/server authentication over distributed environment. This scheme consists of a trusted register center (RC) for generating a ticket to client in the registration phase. The ticket is encrypted with the server's public key not only to prevent modification attack but also to get the confidentiality property. Then, the client makes registration to server by sending ticket with his identity. The server stores client's identity, the number n and the n^{th} hash of user password. When the user logs in to server, the client computes $n\text{-}1^{th}$ hash of user password and sends it to server. The server makes hashing of the client's password and compares the result with the value stored in database. If the result is equal, the server will allow the user as authenticated user. Otherwise, the server will deny the user's request. Then, the server will update the number of login (n) in database. For granted user, the server replies the authenticator to user for authentication. The overview of this system will be explained in Figure 3.

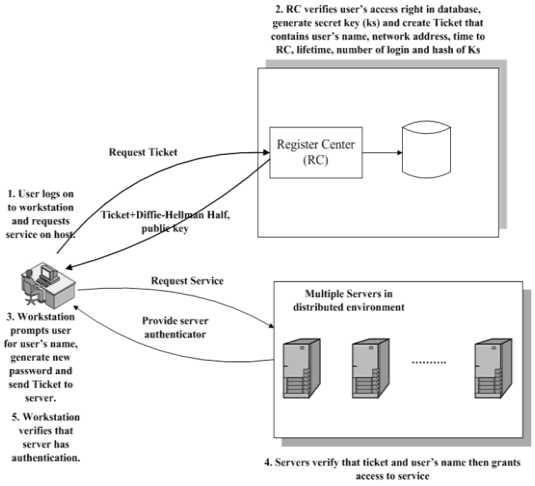


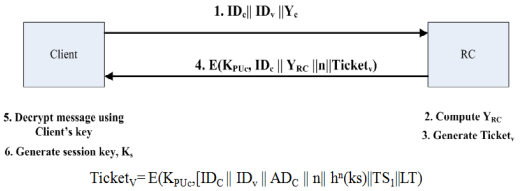
Figure 3. Overview of System

7.1. Phase of Implementation Aspect

There are three mainly phases in this system such as (1) requesting ticket from RC phase (2) registration to server phase and (3) login to server phase. At each step, the authentication gains are considered in this system.

7.1.1. Phase of Requesting Ticket to Register Center (RC)

Every user who wants to register to server needs to request ticket from register center. The user sends his/her identity, the requested server's identity and Diffie-Hellman public key to RC. Then, RC generate secret key and send Ticket to user using the user's public key.

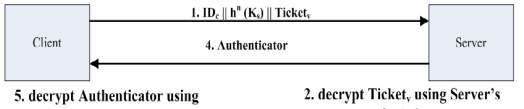


$Ticket_V = E(K_{PUc}, [ID_C \| ID_v \| AD_C \| n \| h^n(ks) \| TS_1 \| LT)$

Figure 4. Requesting Ticket to Register Center

7.1.2. Phase of Registration to Server

When the user gets the response from RC, he/she sends Ticket to server to make registration. When the message from client arrives at server, the server decrypts Ticket and verifies the user. If the user is authorized, the server stores user's identity, number of login (n) and n^{th} hash of password in database. The server returns the authenticator to user with the aim of user can authenticate the server.



$Ticket_V = E(K_{PUc}, [ID_C \| ID_v \| AD_C \| n \| h^n(ks) \| TS_1 \| LT)$

$Authenticator = E(K_{PRv}, [ID_C \| ID_v \| TS_2)$

Figure 5. Registration to server

7.1.3. Phase of Login to Server

The user logs in to server using his/her identity and n-1th hash of user's password. Server make hashing the incoming message and compares with the value stored in database. If the values are equal, the user is granted as authorized user. Then the server will update the value in database.

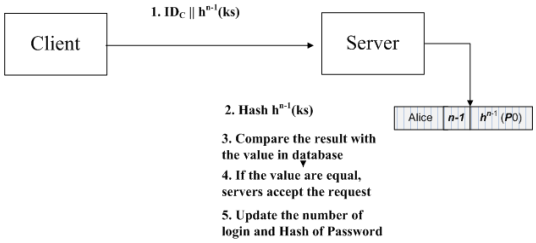


Figure 6. Login to server

8. Security Analyses of Proposed Scheme

In this section, the security features of the proposed scheme are analyzed.

(a) Mutual authentication: The server can authenticate the users by comparing the value in ticket and users' information. The client can also authenticate the server because of usage of authenticator.

(b) Preventing modification attack: In the communication flow between user and register, ticket is encrypted with public key of server to prevent the modification of attackers. Hash function is also used between the interaction between user and server.

(c) Preventing replay attack: Timestamp and lifetime are used in ticket to avoid the replay attack. Similarly, the login number (n) is also used.

(d) Preventing stolen-verifier attack: Since the value in database is valid for only one session, it may be useless for eavesdroppers.

(e) Preventing bruce-force attack: As hash function is used for password authentication, it is resistance to bruce-force attack.

9. Conclusion

In this paper, a secure and efficient one-time password authentication scheme is proposed. Although Lamport's one-time password scheme is secure, it still remains many security problems. Moreover, Diffie-Hellman key exchange protocol consumes time complexity and suits in single-server authentication. Therefore, this system uses Diffie-Hellman only for secure password registration. Although Kerberos uses static password authentication system, this system uses One Time Password authentication system. Kerberos causes computation and network overloading for the usage of Ticket granting server (tgs) to reduce this overhead. The decision of initial password is not obvious in One Time Password but this system uses Diffie-Hellman key exchange protocol for security intention. The transmission of registration password over network is insecure in One Time Password, but this system uses Diffie-Hellman key for the security of password registration. While large and small n attack in One Time Password still remains security problem, this system uses lifetime and Register Center for mutual authentication. This system can be extended in infrastructure layer of cloud computing environment intended for the security of servers.

References

[1] Kenneth G. Paterson1 and Douglas Stebila, One-Time-Password-Authenticated Key Exchange, September 4, 2009

[2] Keith A. Watson, One-time Password Systems, October 13, 2011

[3] Mijin Kim, Byunghee Lee, Seungjoo Kim, and Dongho Won, Weaknesses and Improvements of a One-time Password Authentication Scheme, Vol. 2, No. 4, December, 2009

[4] Dinei Florencio and Cormac Herley, One-Time Password Access to Any Server Without Changing the Server, Vol. 2, No. 4, December, 2009

[5] Minkyu Kim, A Survey of Kerberos V and Public-Key Kerberos Security, http://www.cse.wustl.edu/~jain/cse571-09/ftp/kerb5/index.html

[6] Keith Palmgren, Diffe-Hellman Key Exchange: A Non-mathematician's explanation, ISSA Journal | October 2006

[7] William Stallings, Cryptography and Network Security Principles and Practices, Fourth Edition, November 16, 2005

[8] Behrouz A. Forouzan, Cryptography and Network Security, International Edition 2008