INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

AUTHENTICATION USING PASSWORD-ONLY AND KEY EXCHANGE FOR MULTI SERVER ENVIRONMENT

SHABANA T PIRJADE, PROF. P. K. DESHMUKH, SEEMA SHABADI

Rajarshi Shahu College of Engineering, Tathawade.

Accepted Date:

27/02/2013

Publish Date:

01/04/2013

Keywords

Password authentication,

Guessing attacks,

Multi-Server

Environment.

Corresponding Author

Mrs. Shabana T Pirjade

Abstract

Typical protocols for password-based authentication assume a single server which stores all the information (e.g., the password) necessary to authenticate a user. Unfortunately, an inherent limitation of this approach (assuming low-entropy passwords are used) is that the user's password is exposed if this server is ever compromised. To address this issue, a number of schemes have been proposed in which a user's password information is shared among multiple servers, and these servers cooperate in a threshold manner when the user wants to authenticate. The Multi Server Environment is quite promising for password based authentication, well suited for the setting of federated enterprises. However, none of the existing Multi-server password based authentication schemes enables a user to use the same password over multiple service servers, which is deemed an important feature of the Multi-server model. In this paper, we propose a new scheme, enabling this prominent functionality. Our proposed scheme is password-only and key Exchange, and slightly more efficient than the latest two-server password based authentication scheme.

I. INTRODUCTION

With the rapid growth of Internet technologies, the system providing resources to be accessed over the network often consists of many different servers around the world. The distribution of the remote system hardware in different places makes the user access the resources more efficiently and conveniently. In practice, the Multi-server model is as shown in Figure1 .Password based authentication is the most commonly used entity authentication technique, due to the fact that no secure storage is required, and a user only needs to memorize his password and then can authenticate anywhere, anytime. Protocols designed and proven secure for the case when clients use cryptographic secrets are generally insecure when passwords are used instead; this is so because these protocols are typically not resistant to off-line dictionary attacks in which an eavesdropping adversary derives information about the password from observed transcripts of login sessions. The dictionary attacks can occur *online* or *off-line*: in an on-line attack, the attacker repeatedly picks a password from the dictionary and login with it to the server by impersonating a legitimate user. Specifically, the two-server model comprises a front-end server, called Service Server (SA), and a back-end server, called Control Server (SB).

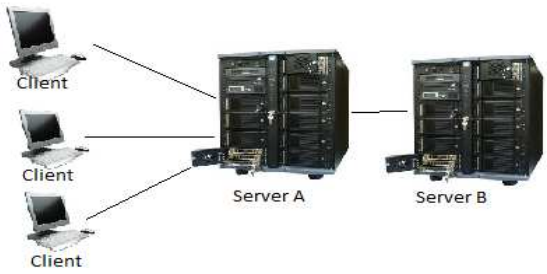

Figure 1. Multi-Server Model

The SA is the actual one providing certain services to users, and it is thus the one users communicate with; the SB stays behind the scene, and its sole responsibility is to help secret-share user passwords so as to avoid single point of failure, as well as to assist the SA for user authentication; users do not contact the SB during authentication, and they are even not necessarily aware of the existence of the SB. Key agreement protocols (for two parties) are a fundamental building block for ensuring secure communications between two parties over an insecure network. Generally, key agreement protocols allow two communicating parties who never met in advance to establish a common secret key via public communication. The agreed secret key, which is usually called a session key, can then be used to create a confidential or integrity-protected communication channel between the parties. We are motivated to propose a new scheme that enables a user to use the same password over multiple service servers in the Multi-server model. Our scheme is password-only, and efficient, achieving slightly better efficiency than the latest two-server password based authentication scheme in [12], which is the most round efficient of its kind in the literature. The rest of the paper is organized as follows. Section 2 presents related works Our new scheme is presented in Section 3, In Section 4 Security analysis of our protocol is done and Section 5 concludes this work

II. BACKGROUND AND RELATED WORKS

In recent years, much attention has focused on designing password based authenticated key exchange protocols which can resist offline dictionary attack by an intruder. To solve this problem, a new kind of authentication structure called the multiple server authentications was proposed. In such schemes, the capability of verifying a password is split between two or more servers, thus securing the system from intruders. In these multiple server authentication system, the two-server authentication protocol is the simplest and the most acceptable to users. Some of the authentication systems based on two server concept are discussed below. Mukesh et al.’s [4] proposed a robust finger print based two-server authentication and key exchange system; this is the first biometric two server authentication scheme. In this scheme, the user’s password is replaced by the random string generated by fingerprint template; the user need not memorize it. In Brainard et al.’s [2] two-server password system in which one server (called Blue Server or Service Server, SS for short) exposes itself to users and the other (called Red Server or Control Server, CS for short) is hidden from the public. While this two-server setting is very interesting, it is not a password-only system; both servers need to have public keys to protect the communication channel from users to servers. This setting makes it difficult to fully enjoy the benefits of a password system. Subsequently, Katz et al. Proposed a two-server password-only authenticated key exchange, the authors claimed that this is the first provably-secure two-server protocol for the important password-only setting (in which the user need remember only a password, and not the server's public keys), and is the first two-server protocol (in any setting) with a proof of security in the standard model.

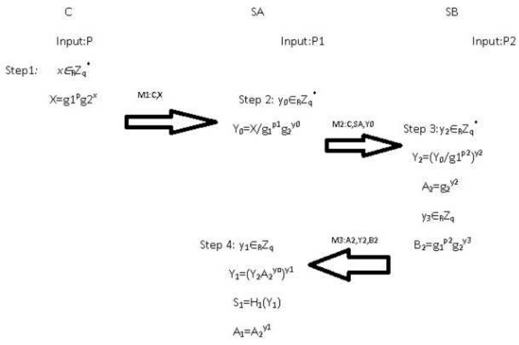

Figure 2. Authentication and Key Exchange Protocol

Recently, a practical two server architecture was proposed by Yang et al. and an efficient password-only two-server authenticated key exchange system, this scheme is a password-only variant of the one introduced by Brainard et al.'s. None of the existing two-server password based authentication schemes enables a user to use the same password over multiple service servers, which is deemed an important feature of the two-server model. Yanjiang Yang proposed a new scheme, enabling this prominent functionality.

III. THE PROPOSED PROTOCOL

The proposed protocol is divided into three phases: the system initialization, the client registration and the authenticated key exchange phases. We describe each phase as follows:

A. Initialization Phase

Let *G* be a multiplicative group with order *q*, where *q* is a big prime. Let $H_0(.):\{0,1\}^* \rightarrow G$, and $H_1(.),H_2(.),H_3(.):\{0,1\}^* \rightarrow \{0,1\}^s$ be cryptographic hash functions, where *s* is a system parameter.

B. Registration Phase

Each Client needs to register in advance his password to every service server he wants to communicate with, as well as the control server SB. To register to a particular service server SA and SB, a Client C (supposing his password is P) generates two random password shares P1;*P*2 such that *P* = *p1* +p2 (mod *q*). C then registers P1 to SA and P2 to SB through some out-of-band channels. Note that SB needs to associate P2 not only with C, but also with that particular SA, since we allow C to register with different service servers using the same password P.

C. Authenticated Key Exchange Phase

Suppose a Client C has password *P*, and the corresponding shares at a service server SA and the control server SB are *P*1 and P2, respectively.

The authentication and key exchange protocol is depicted in Figure 2, which ends up allowing C and SA to authenticate each other and establish a common session key *K*. Note that in the protocol, *g*1*; g*2 are generated in real time by each party as follows. C generates *g*1 = *H*0(*C; SA;* 1), *g*2 = *H*0(C*; SB;* 2) using his own identity and that of the service server he wants to communicate with; following the same format, SA generates *g*1 and *g*2 with its identity and that of the Requesting user; SB generates *g*1 and *g*2 using the identity of the requesting SA and the user. Intuitively, the protocol can be viewed to be composed of two parts: one part are the messages associated with *X; Y*0*;A*2*; Y*2*;A*1*; S*1 and *S*2, which enable C and SA for mutual authentication and key exchange; the other part are the messages associated with *B*2*;B*1*;B*0*;A*0 and *S*3, which enable SB to authenticate C and SA. There are seven steps for authentication and key exchange

Step 1. The Client C randomly chooses an integer x from Zq*, Compute $X=g_1^p g_2^x$ and then send a request message

M1={C, X} to SA

Step 2. On receiving M1, the Server SA randomly chooses an integer y_0 from Zq* and computes $Y_0=X/g_1^{p1}g_2^{y0}$ and then send a message M2={C, SA, Y_0} to the SB

Step 3. After receiving M2 the server SB randomly chooses y2 from Zq* and Computes $Y_2=(Y_0/g_1^{p2})^{y2}$

$A_2=g_2^{y2}$

And then chooses y_3 from Zq* and computes $B_2=g_1^{p2}g_2^{y3}$ and then send a message

M3= {A_2, Y2, B2} to SS

Step 4.After receiving M3 SA chooses randomly an integer y_1 from Zq* and computes

$Y_1=(Y_2A_2^{y0})^{y1}$

$S_1=H_1(Y_1)$

$A_1=A_2^{y1}$ and then chooses y_4 from Zq* and computes $B_1=B_2g_1^{p1}g_2^{y4}$ and then send a message

$M4=\{A_1,S_1,B_1\}$ to Client C.

Step 5.After receiving message M4 Client checks whether $S_1 \overset{?}{=} H_1(A_1^{x})$

$S_2=H_1(A_1^{x},0)$

$K=H_2(A_1^{x},C,SS)$

and then randomly select x* from Zq* and computes $B_0=(B_1/g_1^{p})^{x*}$

$A_0=g_2^{x}*$

and send message $M5=\{S_2,B_0,A_0\}$ to SS.

Step 6.On receiving M5 from Client,SA checks whether $S_2 \overset{?}{=} H_1(Y_1,0)$

and computes $K=H_2(Y_1,C,SS)$

$S_3=H_3\ (B_0/A_0^{y4})$ and send message $M6=\{A_0,S_3\}$ to CS

Step 7.After receiving M6 from SA, SB checks

$S_3 \overset{?}{=} H_3(A_0^{y3})$ and SB authenticates SA and C.

IV. SECURITY ANALYSIS

A. Brute force attacks

In case of any Brute force attacks, it may be either Dictionary attack or exhaustive search, this method works. For example consider a scenario, if an intruder wants to crack the password, then he is going to try all the possibilities starting from integer '1'. Here the valid password is "1234567" and the random number is" 3". All the single and two digit tries are easily rejected by the system since the random number is 3. The front-end server is compromised after he entered 123.But the backend server is not compromised still and his next try will be '124'. But in this case the front end server is decompromised and the user is blocked.Because, once the front end server is compromised,again it should not be decompromised. In this case we can easily identify the intruder due to the rollback of front end server from compromised to decompromised state. Hence our two server system is effective against exhaustive search or brute force attacks.

B. Strengthening Condition

Once the front end server is compromised in first attempt, the backend should be compromised within five attempts. Probably only the legitimate user can compromise the server in first attempt. Incase due to some careless, he may enter the wrong password for backend server. However the five attempts are too much for that legitimate user. The next condition is once the front end server is compromised it should not be decompromised

again. So from the above mentioned strengthening conditions it is clear that the proposed system has good security against hacking

V. CONCLUSION

In this paper, we present an efficient and secure authentication scheme for multi-server environment. We demonstrate that our scheme can satisfy all of the essential requirements. Our scheme does not only manage the secret key associated with the users but also achieve user's anonymity. Moreover, our scheme only uses hashing functions to implement mutual verification and session key agreement. our protocol is secure against passive and active attacks in case that one of the two servers is compromised.

VI. REFERENCES

1. M. Bellare, D. Pointcheval, and P. Rogaway. Authenticated key exchange secure against dictionary attacks.In B. Preneel,editor, *EUROCRYPT 2000*, pages 139–155. Springer-Verlag LNCS 1807, 2000.

2. B. Kaliski and M. Szydlo J. Brainard, A. Juels.Nightingale: “A new two-server approach for authentication with short secrets”, in Proceedings of the 12th USENIX Workshop on Security, pages 1-2. IEEE Computer Society, 2003.

3. C. Boyd and A. Mathuria. *Protocols for Authentication and Key Establishment*. Springer-Verlag, Berlin, 2003.

4.]Mukesh, R.Damodaram, A.Subbiah Bharathi,V. “A robust fingerprint based two server authentication and key exchange system”, 3rd International Conference on Communication Systems Software and

Middleware And Workshops, 2008 Bangalore, pp. 167-174.

5. D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In J. Kilian, editor, *CRYPTO 2001*, pages 213–229. Springer-Verlag LNCS 2139, 2001.

6. S.D. Galbraith. Super singular curves in cryptography. In C. Boyd, editor, *ASIACRYPT 2001*, pages 495–513. Springer-Verlag LNCS 2248, 2001.

7. Yang, Y., Bao, F and Deng, R. H (2005) A new architecture for authentication and key exchange using password for federated enterprise. Proc. SEC'05, pp. 95-111.

8. Yang, Y., Deng, R. H and Bao, F. (2006) A practical password based two-server authentication and key exchange system. IEEE Trans. Dependable and Secure Computing, 3(2), 105-114.

9. Yang, Y., Deng, R. H. and Bao, F. (2006) Fortifying password authentication in integrated healthcare delivery systems. Proc. ASIACCS'06, pp. 255-265.[10] Yi, X., Tso, R. and Okamoto, E. (2009) ID-based group password authenticated key exchange. Proc. IWSEC'09, pp. 192-211.

10. S. Halevi, H. Krawczyk: Public-key Cryptography and Password Protocols. In: Proc. ACM. Computer and Communication Security, CCS'98, pp. 122-131, 1998.

11. H. Jin, D. Wong, Y. Xu: An Efficient Password-Only Two-Server Authenticated Key Exchange System. In: Proc. International Conference on Information and Communications Security, ICICS'07, LNCS 4861, pp. 44-56, 2007.