

A Zero Knowledge Password Proof Mutual Authentication Technique Against Real-Time Phishing Attacks

Mohsen Sharifi, Alireza Saberi, Mojtaba Vahidi, and Mohammad Zorufi

Computer Engineering Department, Iran University of Science and Technology
msharifi@iust.ac.ir,
{a_saberi, mojtavahideh, zorufi}@comp.iust.ac.ir

Abstract. Phishing attack is a kind of identity theft trying to steal confidential data. Existing approaches against phishing attacks cannot prevent real-time phishing attacks. This paper proposes an Anti-Phishing Authentication (APA) technique to detect and prevent real-time phishing attacks. It uses 2-way authentication and zero-knowledge password proof. Users are recommended to customize their user interfaces and thus defend themselves against spoofing. The proposed technique assumes the preexistence of a shared secret key between any two communicating partners, and ignores the existence of any malware at client sides.

1 Introduction

Phishing is a branch of internet crimes. In these attacks, users' sensitive information such as passwords and credit card details are captured. Attackers use social engineering in their attacks to masquerade themselves as legitimate servers [1].

In a phishing attack, the attacker spoofs a trusted website and then sends e-mail(s) to users. An encouraged user clicks on a link embedded in the email. By following the link, the unaware user is redirected to a fake website. Due to the similarity between this site and the trusted one, user may enter the phisher's desired information. In this stage, phisher has gained sufficient information and may forge user's identity or withdraw from victim's internet banking account [2, 3].

In traditional phishing attacks, phisher is only connected to the user and saves captured user sensitive information. After that, in a suitable time, phisher sends this information to forge user's identity and abuse the victim's resources. Almost all existing solutions are designed to fight against traditional phishing attacks.

In real-time phishing, phisher stands in the middle of communication between a user and a trusted web site. After getting user's information, phisher sends them to the trusted web site and replays the reply to the user. Detection of real-time phishing attacks is harder than detection of traditional ones. Nearly none of the existing methods are able to detect real-time phishing attacks. This paper proposes an Anti-Phishing Authentication (APA) technique for fighting against real-time phishing.

The paper is structured as follows. Our proposed technique and the way it deals with phishing are discussed in Sections 2. The implementation is presented in Section 3. Sections 4 and 5 are devoted to the evaluation of the proposed technique and conclusions.

2 The Proposed Method

In this section, our Anti-Phishing Authentication (APA) mechanism is proposed. A short secret key (e.g. password) is assumed between a user and a trusted website.

APA is based on SPEKE [8] which is a cryptographic method for password authentication key agreement. SPEKE uses passwords to resist against man in the middle attacks. It performs a 2-way authentication and resists on-line and off-line dictionary attacks [8]. SPEKE is used in APA with some modifications.

SPEKE has two phases. The two sides of communication are user and (web) server. First, user and server start to create a session key (k) using password. Then, each side authenticates itself to the opposite party using the session key.

APA only enhances the second phase of SPEKE. Table 1 describes the symbols used in subsequent tables. Table 2 explains the second phase of authentication process in APA. Using SPEKE, APA allows communicating parties to authenticate each other without revealing passwords, i.e. by zero knowledge password proof.

Table 1. Symbols used in authentication scenario [8]

Symbol	Description
R_U, C_U	Random numbers generated by user
R_{AS}, C_{AS}	Random number generated by server
$E_k(m)$	Symmetric encryption of message m using key k
$D_k(m)$	Symmetric decryption of message m using key k
$U \Rightarrow AS: m$	User sends message m to server
$AS \Rightarrow U: m$	Server sends message m to user
K	Session key
AS_{IP}	IP address of server
PH	Phisher

Table 2. The second phase of authentication in APA

#	Action	Description
1	$U \Rightarrow AS: E_k(C_U, AS_{IP})$	User generates random number C_U then encrypts C_U and IP address of opposite side. User encrypts message m using key k and sends it to opposite side.
2	$AS: D_k(E_k(C_U, AS_{IP}))$	Server decrypts user's message. If the IP inside the message does not match with its IP address, authentication fails.
3	$AS \Rightarrow U: E_k(C_U, C_{AS})$	Server generates the random number C_{AS} and encrypts C_U and C_{AS} . Server encrypts message m using key k and sends message to user.
4	$U: D_k(E_k(C_U, C_{AS}))$	User decrypts server's message to validate legitimacy of C_U .
5	$U \Rightarrow AS: E_k(C_{AS})$	User encrypts C_{AS} and sends it to server. From user point of view authentication finished successfully.
6	$AS: D_k(E_k(C_{AS}))$	Server decrypts user's message. Authentication is successful if the received C_{AS} matches the sent C_{AS} .

2.1 Prevention from Traditional Phishing Attacks

Users under traditional attacks conceive phishers as trusted servers and reveal their passwords. But by using APA, attackers are unaware of password and cannot generate session key and therefore are defeated in the first phase of authentication.

APA takes advantage of SPEKE by forcing phishers to generate a session key without knowing the password at the end of the first phase of SPEKE. This task entails lots of computational time (say in orders of months using a single computer) and the session key is valid only for the short period of this session; i.e. even if the session key can be generated, it is useless since the validity of the key must have been expired a long time ago. The time complexity of these attacks against SPEKE is noted in [8, 9].

2.2 Prevention from Real-Time Phishing Attacks

In real-time phishing attacks, phisher is located in the middle of communication between user and server, and replays the received information from one side to another. On-line (real-time) phishing attacks can be run only in two ways.

In the first way, phisher is located in the middle of user and server, and introduces himself/herself as a user to server and vice versa and starts authentication with both sides *simultaneously* and *separately*. As in traditional phishing, due to unawareness of password, attacker cannot generate a suitable session key. Therefore, user and server will detect the existence of attacker in the middle of communication and can stop the authentication.

In the second scenario, phisher is located in the middle of user and server and replays the messages between user and server without any modification. At the end of the first phase of APA authentication process, user and server generate a session key (k) without being aware of phisher’s existence in the middle of their communication.

In this scenario, phisher just replays packets between user and server without modification and as a result he/she would be unaware of password. It is impossible for attacker to generate a session key without information about password. As in the previous scenario, this is equal to finding a session key during the first phase of SPEKE without awareness of password. We know that session key generation in a limited period of session key validation is impossible. Although at the end of the first

Table 3. Prevention of real-time phishing in the second phase by APA

	Action	Description
1	$U \Rightarrow PH: E_K(C_U, PH_{IP})$	User generates the random number C_U , encrypts C_U and IP of the opposite side. User encrypts message m using key k and sends it to the opposite side.
2	$PH \Rightarrow AS: E_K(C_U, PH_{IP})$	Phisher does not know the key session so he cannot manipulate message. Phisher can only replay message to server.
3	$AS: D_K(E_K(C_U, PH_{IP}))$	Server decrypts user’s message. The IP inside the message (PH_{IP}) does not match with servers IP (AS_{IP}). From server point of view authentication fails.
4	wait	User does not receive any message so after a while, “Connection Time Out” occurs and from user point of view authentication fails.

phase, the attacker is hidden from user and server, but because of unawareness of session key, attacker cannot read the messages of the second phase or modify them.

Table 3 shows the steps that lead to detection of the second real-time phishing scenario by APA.

3 APA Implementation

APA toolbar implementation has two components: one to be located in server to do authentication, and one to be installed on user's web browser for authentication.

Users should enter the user name and password in the APA toolbar. To prevent forging the toolbar by attackers, a personalization facility is added to the toolbar in which user can put an image to make it harder to forge the toolbar. Authentication in server side is implemented as library functions. Programmer needs only to call the functions in the login page. So, APA can be employed in all current web based applications; by only changing the login form of web applications, one can use APA.

4 Evaluation

Since APA is based on password, it is required that a shared password exists between server and user. Furthermore, APA cannot defend against malware. But if phishers want to masquerade themselves as real servers to users, they have to overcome APA authentication mechanism that is equivalent to attacking SPEKE. Since no serious flaws have been reported against SPEKE yet [8], APA is secure too.

There exists a method [5] against real-time phishing attacks, which needs costly hardware token. APA implementation does not require any hardware. Despite proposed methods in [6, 7], APA does not need initial setting for each site. In addition, users are not required to specify their sensitive information initially such as in [4]. Despite location based approaches [4, 6, 7], users may employ any computer without limitation. APA does not require Certificate Authority or authentication centers. Since there is no need to send password, APA is also immune against eavesdropping. Attacker may not get password by sniffing network. APA also resists against DNS poisoning attacks known as Pharming. In this type of attack, user trusted internet address is redirected to phisher computer. Without dependency on SSL or Certificate Authority, APA can prevent from such attacks.

5 Conclusion

The increased number of phishing attacks and identity theft has increased demands for effective mechanisms to fight against them. Although various approaches have been introduced to counter phishing, most of them are not immune against real-time phishing or are expensive to use. In this paper, a method called APA was proposed. In addition to real-time and traditional phishing prevention, APA has simple implementation. It is possible to employ APA in all current sites and is inexpensive.

References

1. Chou, N., Ledesma, R., Teraguchi, Y., Mitchell, J.C.: Client-Side Defense against Web-Based Identity Theft. In: 11th Annual Network and Distributed System Security Symposium, San Diego, USA (February 2004)
2. Dhamija, R., Tygar, J.D., Hearst, M.: Why Phishing Works. In: CHI Conference on Human Factors in Computing Systems, Montreal, Canada (2006)
3. Kirda, E., Kruegel, C.: Protecting Users against Phishing Attacks with AntiPhish. In: 29th IEEE Annual International Computer Software and Applications Conference, UK (2005)
4. Anti-Phishing Working Group: Phishing Activity Trends Report (2005), http://antiphishing.org/reports/APWG_Phishing_Activity_Report_May_2005.pdf
5. Anti-Phishing Working Group: Phishing Activity Trends Report (2006), http://antiphishing.org/reports/apwg_report_May2006.pdf
6. Herzberg, A., Gbara, A.: TrustBar: Protecting Web Users from Spoofing and Phishing Attacks. Cryptology ePrint Archive, Report 2004/155 (2004), <http://www.cs.biu.ac.il/~herzbea/TrustBar/>
7. Yee, K., Sitaker, K.: Passpet: Convenient Password Management and Phishing Protection. In: Second symposium on Usable privacy and security, Pittsburgh, Pennsylvania, USA (2006)
8. Jablon, D.: Strong Password-Only Authenticated Key Exchange Computer Communication Rev. ACM SIGCOMM 26, 5–26 (1996)
9. Zhang, M.: Analysis of the SPEKE Password-Authenticated Key Exchange Protocol. Communications Letters 8(1), 63–65 (2004)