*Oriental Journal of Computer Science & Technology* Vol. **1**(2), 131-136 (2008)

# A secure protocol for authentication of passwords using delayed password disclosure

**V. SRIKANTH, T. PAVAN KUMAR, A. SUSHMA and K. RAJANI**

Department of IST, KLCE, Vaddeswaram (India).

(Received: November 10, 2008; Accepted: December 25, 2008)

**ABSTRACT**

Phishing is a type of attack in which the attackers lure the users to submit their personal information like passwords, credit card details etc. Passwords and security are a perennial problem we all face when using the Internets. There are various solutions, but none of them are perfect. Even the strongest encryption technologies available for common use could be broken given enough computer power or just simple luck on the part of a hacker. But, that doesn't stop computer scientists from trying to come up with new ways to make us electronically safe.

So, an authentication protocol namely, Delayed Password Disclosure is designed. The protocol's goal is aimed at reducing the effectiveness of phishing/spoofing attacks that are becoming increasingly problematic for Internet users. It does provide a user with the tools necessary to recognize an ongoing phishing attack, and prevent the disclosure of his/her entire password, providing graceful security degradation.

**Key words:** doppelganger; password authenticated key exchange; PAKE; phishing; secure user interfaces.

## INTRODUCTION

Phishing is a form of online fraud that is doing increasing damage to the financial industry. In its traditional form, a victim is sent a fraudulent email message directing the victim to fraudulent website, normally hosted on a hacked machine, with some urgent call to action. The purpose of the call to action is typically to incite users to follow a link, by suggesting that they will receive some form of reward for following the link, suffer a penalty for failing to follow, or some combination of the two. The website that the victims are directed to is designed to mimic the appearance of a legitimate site, such as an online bank, vendor or payment system. The goal here is to continue the confidence game, initiated in the call to action, so that the victim remains convinced that she is interacting with the legitimate site. The fraudulent website will request from the victim a number of credentials and other information of interest, such as personally identifying information.

### Current authentication practices

In current web-based login protocols[1], a person logs in to a service provider by sending his user identity and password to the server in question, which then looks up the corresponding record in its database, and performs a comparison to determine whether the password is valid.

The password is typically not stored in plaintext, but rather, a "salted one-way function[2,3]" of the password is stored. This means that if somebody gains access to the database of the service provider, they will not be able to obtain plaintext passwords. However, the password itself is generally sent prior to have the salted one-way function applied. So, it's a bit dangerous if some one eavesdrops it. From the initial times, many

authentication schemes have been proposed such as the pake[4,5]. In this the following procedure takes place. Imagine a solution in which a user must always when presented with a secure login window-press some combination of keys that moves the computation into a "safe state" in which only very restricted authentication functionality and its user interface are made available. This would shift the problem to that of securing the operating system, and allow the secure use of standard mutual authentication techniques. But in the absence of this mechanism delayed password disclosure comes to rescue.

**Doppelganger attacks**

In general in mutual authentication methods[6], the user would enter his user name and password, and the software performing the mutual authentication would connect to the other party, send the user name to this, and then perform a comparison of the password the user entered and the password the other party has stored. If the comparison succeeds, then the user gains access to the service; if not, then the log-in is aborted on both ends. But the problem with this is if a phisher can create a window that looks like the window during a mutual authentication session, but which is not, then he can dupe the user to enter his password into a form that will cause it to be sent over to the attacker. This is referred to as the doppelganger window attack.

**There are two types of doppelganger attacks-**

1. Offline doppelganger attacks
2. Online doppelganger attacks

**Offline doppelganger attacks**

The attacker has one or more accounts with the target site. The attacker is permitted to communicate with the target site a polynomial number of sessions in order to learn the behavior of the target site, and collect other information necessary to duplicate the appearance of the target site. Once this process is completed, the attacker constructs a doppelganger site[7], and tries to cause the user to enter her credentials (associated with the target site) as input to the doppelganger site. The attacker may later connect to the target site in order to attempt to impersonate the victim.

**Online doppelganger attacks**

In contrast to offline doppelganger attacks, here the phisher communicates both with the client and the bank at the same time. Suppose a user visits the doppelganger's site, the doppelganger then quickly visits the authentic site, and based on its appearance draws the same picture on the user's display as was shown to the attacker on his. All of the information sent by the user is available to the attacker as there are no security protocols enabled on the doppelganger site; similarly, all information sent by the authentic site becomes available to the attacker, as he is the apparent end-point of the communication as far as the authentic site is concerned.

**Delayed password disclosure (DPD)**

The DPD approach is a mutual authentication technique that augments password entry with an image sequence specific to the user and service provider. Each user learns to recognize their sequence of images and knows not to enter their password if the images are incorrect. This addresses the doppelganger-window attack. It permits a user interface that provides users with visual character-by-character feedback as they enter their passwords, allowing users to stop entering their password if they obtain feedback that they do not recognize – a sure sign of interacting with the wrong site.

The key idea behind DPD[8] is to augment the traditional username and password system with feedback images that are specific to the users password so that it's not duplicatable by the phisher unless he has access to the users password. Originally there are three reasonable time periods at which the server could provide such images. Before the user enters her username, after the user enters her username but before she enters her password and after the user enters her password. There are a few problems[9] but: they are providing an image as a feedback in any one of these periods does little to stop phishers. If an image is shown to the user before she enters her username then it's not specific to her and it can be easily be retrieved from the website by the phisher. If an image is shown after she enters her username but before she enters password then the image can be made specific to

the user. However on many sites the username is not considered as a secret and the phisher can easily trace it.

The third scheme too doesn't help much. The user can only be informed that she is phished. So, the best way would be to give the feedback while entering the password for each letter. This is what is done in DPD. In particular when a user registers with a server that is using the DPD protocol, he is provided with a series of photographs that correspond to each character in his password and he would be instructed that during any future log on attempts he will be provided these images during password entry. The user is also informed that if any site fails to provide any one of these images then it's a fraudulent site and he should stop entering the password further.

When a user attempts to log on with the DPD system, once he enters his username he begin entering his password. At the completion of entering each letter of the password, its corresponding image will be shown to the user. If it's the correct picture he may proceed and enters the next or his password, repeating the process until the password is fully entered. If any point he receives an image with which he is familiar, he can either stop entering the password completely or enter bogus suffix characters for the reminder of his password.

A phisher will not able to reproduce the feedback images because each image shown to the user is selected based on the output of a pseudo random function[10] that takes as input the user's name and currently entered prefix of the password.

The high level idea of DPD is the following. Imagine that the server has a database of easily distinguishable images. In particular, assume (for explanatory purposes only) that there are as many images as possible prefixes of passwords. Again for the sake of example, let's suppose the characters in a password are chosen from an alphabet of size 26. In delayed password

Disclosure[11], when the user enters the first character in her password she is returned one of the first 26 images in the database.

Specifically we think of the first character of her password as indexing into the image database via an oblivious-transfer protocol. When the user enters the second character of her password, she is returned one of the next $26^2$ images in the database which is indexed by the the first two characters of the password. This process continues for each character in the user's password. Once an image has been returned for each character of the user's password, and they are all the images she was expecting, then she can be relatively sure that she is interacting with the correct protocol.

**Working**

For instance let's think a client has entered his password. It will be converted into a binary form and then it's put in a secret form say some envelope like thing in which for every one(binary digit),the first row gets selected and for every zero(binary digit) the next row is selected. This whole thing is sent to the bank for instance. The bank then selects a set of random numbers and writes the digits in the correct places as sent by the client in red ink (invisible) and the remaining gaps are filled in green ink. This is sent back to the client. The client then opens it and can read the digits the bank has sent him (red ink ones). If confirmed then the corresponding image is displayed.

This process fig. 1 is explained lucidly in the diagram below.

In this way the delayed password disclosure ensures safety to some extent.

4. Security guarantees.

The following are some of the merits of this delayed password disclosure.

**Tricking the attacker**

- The attacker impersonates a bank to the user, and manages to make the user connect to the attacker using the proper DPD software.
- The attacker learns neither password nor images by performing these attacks. While it may not seem obvious at first that he learns no information about the images, this is due to the fact that the actual image is not sent

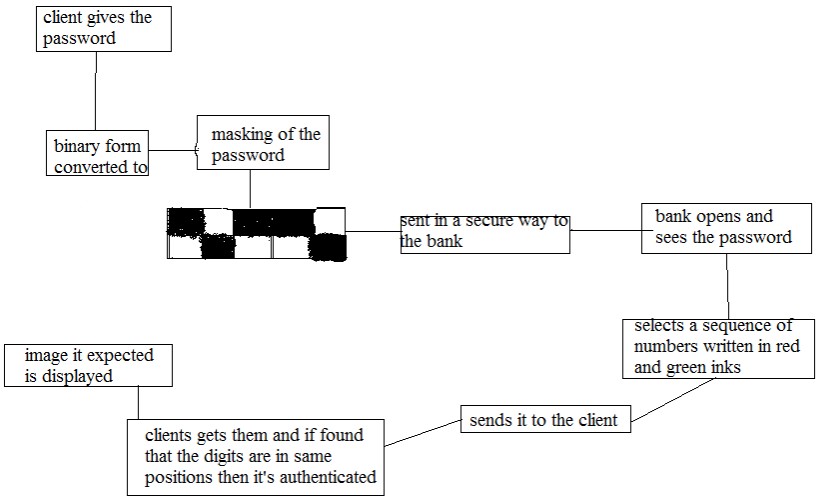


**Fig. 1: Process of DPD**

over by the bank, but rather computed by the user machine as a function of the password characters entered so far, and the transcript received from the bank.

- Hence for each input character will result in a valid-looking image – but the attacker cannot tell whether it is the right one or not!

**A tough challenge**

- The attacker manages to display a doppelganger window[12] on the user's screen, and the user is tricked to perform a password authentication.
- However, since the doppelganger window outwardly looks like a valid DPD window but is not, then the attacker manages to have the user establish a potentially un-encrypted session[13] directly with the attacker. The user enters the first character of the password.
- Now the attacker has to guess what image this corresponds to-note that this set may be substantially larger than the number of alphanumeric symbols, given that the correct image is also a function of the user name and of a secret value only known to the bank
- Therefore, we can see that DPD achieved a higher degree of protection against this attack than other methods for mutual authentication[13], as other such methods do not protect at all against this type of attack.

**Man in the middle attacks**

The attacker performs a man-in-the-middle attack in which he opens a doppelganger window for the user, and then performs a valid DPD connection to the bank in which he claims to be the user. He forwards all information received from the user to the bank. He obtains information back, and computes the valid image from this. The image is sent to the user, and displayed in the doppelganger window[14]. The user recognizes it, and enters the next password character, and so on. As a result of this attack, the attacker manages to learn the entire password sequence and the entire image sequence.

While one might argue that this achieves the same degree of security as would traditional methods for mutual authentication, this is actually not the case. The reason is that as a result of having to interact with the bank, the bank will learn the IP address of the attacker. If the attacker launches multiple attacks over a short period of time, then this will be noticeable by the bank, since many users will log in from the same IP address in this interval of time. Moreover, if the attacker is located in a geographic area quite different from the victim user,

then this will also be evident from the IP address, and special actions can be taken by the bank.

Further, because of the interactive nature of this attack, it is inherently more difficult for attackers to perform. Therefore, we automatically exclude more naïve attackers.

These security measures are therefore heuristic[15] and pattern-based, and are closely related to the fraud-detection techniques employed by credit card companies and telecoms. Therefore, while not achieving perfect security, our technique provides better security against this attack than both traditional password authentication techniques, and previously proposed mutual authentication techniques.

## CONCLUSION

The phishing problem is currently one of the biggest threats to computer security, and more attempts must be made to apply different security techniques at the different choke-points of the phishing problem in order for it to be addressed. It's believed that DPD protocol is an interesting attempt to apply cryptographic techniques to that problem, and is practical for situations where the servers computational load is not an issue.

More sophisticated techniques like Phool Proof antiphishing systems and SSL/User session Aware User Authentication etc are looked into.

## REFERENCES


1. Bellovin, S.M. and Merritt, M., 'Encrypted key exchange: password-based protocols secure against dictionary attacks.Proceedings of the IEEE symposium on Security and Privacy, IEEE Press,May. 72-84 (1992).
2. Bellovin, S.M and Merritt. "Augmented Encrypted Key exchange: a password based protocol against dictionary attacks and password file compromise',CCS'93: Proceedings of the 1st ACM Conference on Computer and Communications Security, New York, NY,USA: ACM Press,pp.244-250
3. Oppliger, R., and S. Gajek, "Effective Protection against Phishing and Web Spoofing" Proceedings of the 9th IFIP TC6 and TC11 Conference on Communications and Multimedia Security
4. Lamport, L., "Password Authentication with Insecure Communication,"
5. Dhamija, R., and J.D. Tygar, "The Battle Against Phishing: Dynamic Security Skins"
6. M.Bhadra and I.Hajjeh,"Key exchange Authentication using shared secrets," computer,Mar-2006
7. Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft] By Markus Jakobsson, Steven Myers Published by Wiley-Interscience, ISBN 0470086092, 9780470086094 (2006).
8. Jonathan Katz , Rafail Ostrovsky , Moti Yung, Efficient Password-Authenticated Key Exchange Using Human-Memorable Passwords, Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques: Advances in Cryptology, p.475-494, May 06-10 (2001).
9. Amir Herzberg and Ahmad Gbara. Trustbar: Protecting (even naive). web users from spoofing and phishing attacks (2004).
10. Collin Jackson, Dan Simon, Desney Tan, and Adam Barth. An evaluation of extended validation and picture-in-picture phishing attacks, In Usable Security (2007).
11. Jonathan Katz , Rafail Ostrovsky , Moti Yung, Efficient Password-Authenticated Key Exchange Using Human-Memorable Passwords, Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques: Advances in Cryptology, 475-494, May 06-10 (2001).
12. Tara Whalen , Kori M. Inkpen, Gathering evidence: use of visual security cues in web browsers, Proceedings of Graphics Interface 2005, May 09-11, Victoria, British Columbia (2005).

13. Steven M. Bellovin , Michael Merritt, Augmented encrypted key exchange: a password-based protocol secure against dictionary attacks and password file compromise, Proceedings of the 1st ACM conference on Computer and communications security, p.244-250, November 03-05, Fairfax, Virginia, United States (1993).
14. Fred Cate. Liability for phishing (chapter 18), In Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft. Editors Markus Jakobsson and Steven Myers, (2006).
15. Neil Chou, Robert Ledesma, Yuka Teraguchi, Dan Boneh, and John C. Mitchell. Client-side defense against web-based identity theft, April (2004).
16. Tara Whalen, Kori M. Inkpen, Gathering evidence: use of visual security cues in web browsers, Proceedings of Graphics Interface May 09-11, 2005, Victoria, British Columbia (2005.
17. Warwick Ford , Burton S. Kaliski, Jr., Server-Assisted Generation of a Strong Secret from a Password, Proceedings of the 9th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, p.176-180, June 04-16 (2000).
18. Aaron Emigh. Online identity theft: Technology, chokepoints and countermeasures. In DHS Report,(2005).