Detecting Phishing Websites: User Behaviour Based

Manohar Kumar Kushwaha[1] **Mr. S. Madhu**[2]
[1]M.Tech(CS&E) [2]Assistant Professor
[1, 2]School of Computer Science and Engineering
[1, 2]Galgotias University, Greater Noida, U.P.

Abstract--- Phishing is a form of identity theft, where criminals create fake web sites that masquerade as trustworthy organizations. The goal of phishing is to trick people into giving sensitive information, such as passwords, personal identification numbers, and so on. Simply phishers creates a phishing website and then goes phishing by sending out promiscuous emails to unsuspecting users. The Phishers tries to convince the reader of the email to visit the link included in the email. When the user “bites” on the phish, the link in the email directs the user to the phishing site which appears legitimate and similar or identical to the legitimate target site. The phish is successful when the user enters confidential information on the phishing page and it is leaked to the phishers. Afterwards the phishers tries to exploit the confidential information by transferring money, opening accounts, or making purchases using the captured information. Or the phishers merely acts as a middleman and sells the information to other criminals. In this paper, we describe a novel approach to detect phishing websites that is based on the analysis of users’ online behaviours.

Keywords: Phishing, Phishing Attacks, Phishing Websites Detection, Identity Theft, User Protection.

I. Introduction

Phishing is a type of Internet scams that seeks to get a user’s credentials by fraud websites, such as passwords, credit card numbers, bank account details and other sensitive information. In other words we can say that Phishing website is a very complicated and complex issue to understand and to analyze, since it is a combination of technical and social dynamics for which there is no known single silver bullet to solve it entirely. Many users believe that using on-line banking increases the likelihood that they will become victims of phishing websites and identity theft, even though on-line banking provides more secure identity protection than paper- and mail-based systems.

In this paper we present our design, and implementation of the *user-behaviour* based phishing website detection system (UBPD). It’s (UBPD) not aim to replacing existing anti-phishing solutions, rather it complements them. It alerts users when they are about to submit credential information to phishing websites and protects users as the last line of defense. Its detection algorithm is independent from how phishing attacks are implemented, and it can easily detect sophisticated phishing websites that other techniques find hard to deal with.

II. Related Studies

Researchers have tried to understand phishing attacks by studying the human factors in phishing, usability of the security features of the systems and the techniques used in phishing attacks. To fight against phishing attacks system designers have invented novel security interfaces, automated detection systems and education methods. In this section we briefly survey the major findings in this area. Web Wallet [31] creates a unified interface for authentication. Once it detects a login form, it asks the user to explicitly indicate the intended site to login. If the intention matches the current site, it automatically fills webpage input fields. Otherwise a warning will be presented to the user. SpoofGurad [8] signature-and-rule based detection system. In this paper, analyses the host name, URL (Unified Resource Locator), and images used in the current webpage to detect the phishing websites. CANTINA [21] A Content-Based Approach to Detecting Phishing Web Sites. In CANTINA uses the TF-IDF (Term Frequency-Inverse Document Frequency) information retrieval algorithm to retrieve the key words of the current webpage, and uses Google search results to decide whether the current website is phishy.

A. *Phishing*

Jacobsson has introduced tools and models to describe a variety of phishing attacks in a uniform and compact manner. He has also presented an overview of potential system vulnerabilities and corresponding defense mechanisms [23]. Abad has presented a comprehensive process flow of a phishing attack [22]. The model includes almost every step of a phishing attack from attack preparation (creation or renting botnets, designing the phishing attacks) to turning the confidential information harvested into a financial profit. Ollmann surveys the technologies and technical security flaws phishers exploit to conduct their attacks, and provides detailed advice on what organizations can do to prevent future. Attacks [25, 26]. Watson et al. review the actual techniques and tools used by phishers, providing three examples of empirical research where real-world phishing attacks were captured using honeynets [24].

B. *Human factors in phishing attacks*

Dhamija et al. have investigated why users fall victim to phishing attacks by carrying out a controlled phishing attack user study [6]. They identified three major causes: (1) a lack of understanding of how computer systems work; (2) a Lack of attention to security; and (3) the high quality visual deception practiced by the phishers. Schechter et al. evaluated website authentication measures that are designed to protect users from man-in-the middle, phishing, and other site forgery attacks and also investigated the impact of role playing on the accuracy of the study result [29]. Their findings are: (1) users will enter their passwords even when HTTPS indicators are absent;

(2) Users will enter their passwords even if site authentication images are absent; (3) site-authentication images may cause users to disregard other important

security indicators; and (4) behaviour in role playing may not be representative of normal behaviours, because in such studies no real losses would be incurred. Jakobsson et al. have studied what makes phishing emails and web pages appear authentic [16]. Elsewhere Jakobsson summarized comprehensively what typical computer users are able to detect when they are carefully watching for signs of phishing [15]. The findings are: (a) spelling and design matter; (b) third party endorsements depend on brand recognition; (c) too much emphasis on security can backfire; (d) people look at URLs; (e) people judge relevance before authenticity; (f) personalization creates trust; (g) emails are very phishy, web pages a bit, phone calls are not; (h) padlock icons have limited direct effects; and (i) independent channels create trust.

C. *Design guidelines*

Wu et al. have discovered that the security tools such as security toolbars are not effective enough to protect people from falling victim to phishing attacks by conducting two studies of three security toolbars and other browser security indicators [28]. Based on their findings, the authors suggest that the alert should always appear at the right time with the right warning message; user intentions should be respected, and if users must make security-critical decisions they should be made consciously; and it is best to integrate security concerns into the critical path of their tasks so that users must address them.

D. *Technology countermeasures*

Many anti-phishing email filters have been invented to fight phishing at the email level, as it is the primary channel for phishers to reach victims. SpamAssassin, 2 PILFER [19], and Spamato [20] are typical examples of those systems. They apply predefined rules and characteristics often found in Phishing emails to analyses incoming emails. PHONEY is a phishing email detection system that tries to detect phishing emails by mimicking user responses and providing fake information to suspicious web sites that request critical information. The web sites' responses are forwarded to the decision engine for further analysis [18]. Web Wallet [29] tries to create a unified interface for authentication. It scans pages for the login form. If such a form exists, then it asks the user to explicitly indicate his/her intended site to login. If the intention matches the current site, it automatically fills webpage input fields. Otherwise a warning will be presented to the user. CANTINA [3o] detects phishing websites based on the TF-IDF information retrieval algorithm. It retrieves the key words of the current webpage and uses Google search to analyses if the current webpage is in the top 30 or not. Ying Pan et al. have invented a phishing website detection system which examines the anomalies in web pages, in particular, the discrepancy between a web site's identity and its structural features and HTTP transactions [27].

III. DESIGN

A. *Phishing Nature and Detection Philosophy*

Simply, phishing attack typically involves sending individuals an e-mail request for information that appears to come from a legitimate company, such as a bank, retailer, or other e-commerce website (the "spoofed company"). Through the use of a false "from" address, copies of company logos, Web links, and graphics, these e-mails have the look and feel of a message that recipients might expect to receive from a company with whom they do business. Often the message makes reference to new security measures allegedly being undertaken by the spoofed company, and asks recipients to verify or reconfirm confidential personal information such as account numbers, Social Security Numbers, passwords, and other sensitive information. To provide a sense of urgency, the message may indicate that the recipient's account will be suspended or cancelled if the information is not verified by a certain date. Phishing attacks can be detected if we can detect such a mismatch. One approach is to predict a user's perception and then compare it with the actual fact understood by the system. CANTINA is an example of this approach [21]. CANTINA makes use of TF-IDF (Term Frequency- Inverse Document Frequency) for detecting phishing sites. TFIDF is a well-known information retrieval algorithm that can be used for comparing and classifying documents, as well as retrieving documents from a large corpus. So, Phishing websites can be detected when both of the following two conditions are met: First as the, current website has rarely or never been visited before by the user and Second as the data, which the user is about to submit, is bound to website other than the current one.

B. *System Design*

1) *Detection work flow:*

User-behaviour based phishing website detection system (UBPD) has two working modes First as the training mode and second as the detection mode. In training mode, User-behaviour based phishing website detection system runs quietly in the background, and focuses on learning newly created binding relationships or updating the existing binding relationships. Only in detection mode, User-behaviour based phishing website detection system checks whether any of the user's binding relationships would be violated if the user submitted data is sent to the current website. The mode in which UBPD runs is decided by checking whether the webpage belongs to a website 1 whose top level domain1 is in the user's personal whitelist, or 2 with which the user has shared authentication credentials. User-behaviour based phishing website detection system (UBPD) has three components**.** First as the user profile, second the monitor and last as the detection engine. The user profile contains the data to describe the user's binding relationships and the user's personal whitelist. The profile must be constructed before the system can detect phishing websites. The monitor collects the data the user intend to submit and the identity of the destination websites, and activates the detection engine. The detection engine uses the data provided by the monitor to detect phishing websites and update the user profile if necessary.

2) *Creation of the User Profile:*

User profile includes the user's binding relationships and personal whitelist. Binding relationships represent the collection of paired records, i.e., (a TopLevelDomain, aSecretDataItem). And personal whitelist represent a list of top level domains of websites. There are two types of binding relationships that *user-behaviour* based phishing website detection system (UBPD) is unaware of: the ones that already exist but users have not trained UBPD with, and the ones that users will create in the future. The personal whitelist is constructed by combining a default whitelist with the websites the user has visited more than three times (configurable) according to the user's browsing history.

3) *Update of the User Profile:*

Simply we can say that some phishing sites don't try to imitate the legitimate site very well. In fact the phishing page may only have a logo or security seal that matches one on the legitimate site. However these pages can be very dangerous because they will request several pieces of sensitive information. In addition to the manual update by users, when running in the training mode user-behaviour based phishing website detection system has an automatic method to update the user profile with the other currently unknown binding relationships. It detects whether the user is using a web authentication form. This is achieved by analyzing the HTML (Hypertext markup language) source code, such as the annotation, label, use of certain tags (such as _form_) and type of the HTML elements. If the user is using a web authentication form, and the user profile contains no binding relationships with the current website, then user-behaviour based phishing website detection system prompts a window to ask the user to update the user profile. If there is an existing binding relationship for the current website, then UBPD will replace the authentication credentials in the binding relationships with the latest values the user submits. If users have entered the authentication credentials wrongly, those credentials will still be stored, but those wrong values will be corrected, when users relog in with the correct authentication credentials.

4) *Phishing Score Calculation:*

In detection mode, user-behaviour based phishing website detection system decides whether the current webpage is a phishing webpage by calculating phishing scores. The Phishing score calculation is a two step process. First step is, for each legitimate website, with which the user has shared authentication credentials, a temporary phishing score is calculated. Each temporary phishing score is the fraction of the authentication credentials associated with a legitimate website that also appear in the data to be submitted to the current webpage. Its value ranges from 0 to 1. In the second step, those temporary phishing scores are sorted into descending order.

5) *Reuse:*

It is very common that a user shares the same authentication credentials (user names, passwords, etc) with more than one website. The two running modes and the user's personal whitelist are designed to prevent false warnings caused by reuse without compromising detection accuracy.

6) *Warning Dialogue:*

It is must be suitable for users with very limited knowledge, otherwise, as "Proceedings of the SIGCHI Conference on Human Factors in computing systems" study [32] found, users may ignore the warning or may not behave as suggested. To make the information easy to understand, the dialogue tells users that the current website, to which they are submitting credentials, is not one of the legitimate websites associated with those authentication credentials. To help users understand the detection result and make a correct decision, user-behaviour based phishing website detection system also provides information regarding the differences between the legitimate website and the possible phishing website in five aspects: the domain name, the domain registrant, the domain registration time, name servers, and IP addresses.

IV. USER'S PRIVACY

User profile contains confidential user information, it is important that it does not add new security risks. We use a one-way secure hash function to hash the confidential data before it is stored. When the system needs to determine the equivalence between the data, the system just needs to compare the hash values.

V. IMPLEMENTATION

User-behaviour based phishing website detection system is implemented as a Firefox add-on that using a JavaScript. The hash function we use is SHA-1 [17] and the encryption method we use is Twofish.5 the hash function and encryption methods can be changed, we choose to use them mainly because there are open source implementations available. The user interface of the system is implemented by using XUL, a technology developed by Mozilla [10].

REFERENCES

[1] Xun Dong, John A. Clark, Jeremy L. Jacob" User Behaviour Based Phishing Websites Detection".
[2] The Anti-Phishing work Group, http://www.apwg.org/
[3] http://en.wikipedia.org/wiki/Phishing.
[4] Isredza Rahmi A Hamid, Jemal ABA W AJY Tai-hoon KIM
[5] Toolan, F., J. Carthy, Feature Selection for Spam and Phishing Detection, In eCrime Researchers Summit (eCrime), 2010, pp. 1-12.
[6] Dhamija, R., Tygar, D., &Hearst, M. (2006).Why phishing works. In CHI '06: proceedings of the SIGCHI conference on human factors in computing systems, ACM Special Interest Group on Computer-Human Interaction 2006 (pp. 581–590). Schechter, S., Dhamija, R., Ozment, A., & Fischer, I. (2007). The emperor's new security indicators: an evaluation of website authentication and the effect of role playing on usability studies. In 2007 IEEE symposium on security and privacy, 2007.
[7] ZHANG, J., Z. DU, W. LIU, A Behaviourbased Detection Approach to MassMailing Host, In Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, vol. 4, 2007, pp. 2140-2144.
[8] N. Chou, R. Ledesma, Y. Teraguchi, D. Boneh, and J. C. Mitchell. Client-side defense against web-based identity theft. In NDSS '04: Proceedings of the 11th Annual Network and Distributed System Security Symposium, February 2004.

[9] Ammar ALmomani,Tat-Chee Wan, Ahmad Manasrah,Altyeb Altaher,2Eman Almomani, Karim Al-Saedi, Ahmad ALnajjar and Sureswaran Ramadass "A survey of Learning Based Techniques of Phishing Email Filtering"

[10] X. Dong, J. A. Clark, and J. Jacob. A user-phishing interaction model. In Conference on Human System Interaction, 2008.

[11] Chandrasekaran, M., K. Narayanan, S. Upadyaya, Phishing Email Detection Based on Structural Properties, Proceeding of the Cyber Security Conference, 2006.

[12] Chandrasekaran, M., V.Shankaranara Y Anan, S.Up Adhy A Y A, Cusp: Customizable and Usable Spam Filters for Detecting Phishing Emails, Proceeding 3r Annual Symposium on Information Assurance (ASIA '08), Albany, NY, 2008, pp. 10-17.

[13] Syed, N. A., N. Feamster, A. Gray, Learning To Predict Bad Behaviour, NIPS 2007 Workshop on Machine Learning in Adversarial Environments for Computer Security, 2008.

[14] Ammar Almomani, B. B. Gupta, Samer Atawneh, A. Meulenberg, and Eman Almomani "A Survey of Phishing Email Filtering Techniques"

[15] Jakobsson, M. (2007). Human factors in phishing. In Privacy & security of consumer information '07, 2007.

[16] Jakobsson, M., Tsow, A., Shah, A., Blevis, E., & Lim, Y.-K. (2007). what instills trust? A qualitative study of phishing. In Extended abstract, USEC '07, 2007. Wu, M., Miller, R. C., & Garfinkel, S. L. (2006). Do security toolbars actually prevent phishing attacks? In CHI '06: proceedings of the SIGCHI conference on human factors in computing systems, New York, NY, USA, 2006 (pp. 601–610). New York: ACM IJ7Press.

[17] Johnston, P. A. (2009). http://pajhome.org.uk/crypt/index.html.

[18] Chandrasekaran, M., Chinchain, R., & Upadhyaya, S. (2006). Mimicking user response to prevent phishing attacks. In IEEE international symposium on a world of wireless, mobile, and multimedia networks, 2006.

[19] Fette, I., Sadeh, N., & Tomasic, A. (2007). Learning to detect phishing emails. In WWW '07: proceedings of the 16th international conference on World Wide Web, New York, NY, USA, 2007 (pp. 649–656). New York: ACM Press.

[20] Albrecht, K., Burri, N., & Wattenhofer, R. (2005). Spamato-an extendable spam filter system. In 2nd Conference on email and anti-spam (CEAS), Stanford University, Palo Alto, California, USA, July 2005.

[21] Zolnikov, P., Extending Explorer with Band Objects using.NET and Windows Forms, The Code Project - C# Programming. Visited: Nov 20, 2006. http://www.codeproject.com/csharp/dotnetbandobjects.asp.

[22] Abad, C. (2005). The economy of phishing: a survey of the operations of the phishing market. First Monday, 10(9).

[23] Jakobsson, M. (2005). Modeling and preventing phishing attacks. In Phishing panel in financial cryptography '05, 2005.

[24] Watson, D., Holz, T., & Mueller, S. (2005). Know your enemy: phishing (Technical report). The Honeynet Project & Research Alliance.

[25] Ollmann, G. (2005). The pharming guide (Technical report). Next Generation Security Software Ltd.

[26] Ollmann, G. (2009). The phishing guide (Technical report). NGSS.

[27] Pan, Y., & Ding, X. (2006). Anomaly based web phishing page detection. Acsac, 0, 381–392.

[28] Phishtank (2007). http://www.phishtank.com/.

[29] Schechter, S., Dhamija, R., Ozment, A., & Fischer, I. (2007). The emperor's new security indicators: an evaluation of website authentication and the effect of role playing on usability studies. In 2007 IEEE symposium on security and privacy, 2007.

[30] Zhang, Y., Hong, J. I., & Cranor, L. F. (2007). Cantina: a content based approach to detecting phishing web sites. In WWW '07: proceedings of the 16th international conference on World Wide Web, New York, NY, USA, 2007 (pp. 639–648). New York: ACM Press.

[31] M. Wu, R. C. Miller, and G. Little. Web wallet: preventing phishing attacks by revealing user intentions. Pages 102–113, 2006.

[32] M. Wu, R. C. Miller, and S. L. Garfinkel. Do security toolbars actually prevent phishing attacks? In CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems, pages 601–610, New York, NY, USA, 2006. ACM Press.