Expert Systems with Applications 39 (2012) 11861–11869

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Using automated individual white-list to protect web digital identities

Weili Han [a,d,*], Ye Cao [a], Elisa Bertino [b], Jianming Yong [c]

[a] *Software School, Fudan University, Shanghai 201203, China*
[b] *Department of Computer Science, Purdue University, West Lafayette, IN 47907, USA*
[c] *School of Information Systems, University of Southern Queensland, Queensland 4350, Australia*
[d] *Key Lab of Information Network Security, Ministry of Public Security, Shanghai 201204, China*

ARTICLE INFO

Keywords:
Individual white-list
Web digital identity
Identity theft
Naïve Bayesian classifier
Anti-phishing
Anti-pharming

ABSTRACT

The theft attacks of web digital identities, e.g., phishing, and pharming, could result in severe loss to users and vendors, and even hold users back from using online services, e-business services, especially. In this paper, we propose an approach, referred to as automated individual white-list (AIWL), to protect user's web digital identities. AIWL leverages a Naïve Bayesian classifier to automatically maintain an individual white-list of a user. If the user tries to submit his or her account information to a web site that does not match the white-list, AIWL will alert the user of the possible attack. Furthermore, AIWL keeps track of the features of login pages (e.g., IP addresses, document object model (DOM) paths of input widgets) in the individual white-list. By checking the legitimacy of these features, AIWL can efficiently defend users against hard attacks, especially pharming, and even dynamic pharming. Our experimental results and user studies show that AIWL is an efficient tool for protecting web digital identities.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Web digital identities (Chen, Wu, Shen, & Ji, 2011) in the form of pairs of usernames and passwords is a commonly used mechanism to authenticate individuals wishing to carry on transactions across the World Wide Web (Web for short). Applications that rely on such mechanisms include webmail, on-line banking, and social networking services (SNSs). It is not a surprise, thus, that a variety of attacks that aim at stealing user's web digital identities are perpetrated. Among these attacks, phishing is the most widespread one. Phishing employs social engineering to trick a user into revealing his or her web digital identities to a fraudulent web site. The open source model of web pages makes it easy for attackers to create an exact replica of a legitimate site. Because such a replica can be easily created with little cost and looks very convincing to users, many such fraudulent web sites continuously appear (Fette, Sadeh, & Tomasic, 2007; Zhang, Egelman, Cranor, & Hong, 2007). As a result, phishing not only leads to a severe threat to user's web digital identities, but also erodes the fundamental premise of activities and business on the Web.

Users are not usually skillful enough to defend themselves against the theft attacks of web digital identities, especially phishing attacks (Dodge, Carver, & Ferguson, 2007; He, Horng, & Fan, 2011; Sheng et al., 2007), because fraudulent web sites generally have appearances similar to the genuine ones. Moreover, the URLs of fraudulent web sites are forged so to look very similar, and sometimes even identical, to the legitimate sites. So it is difficult for even a more careful user to detect fraudulent web sites.

Because of the potential severe damages resulting from phishing attacks, anti-phishing techniques and tools represent a very active research area in web security. Many approaches and tools have thus been developed to address the problem of phishing (Aburrous, Hossain, Dahal, & Thabtah, 2010; eBay Toolbar's Account Guard, 2011; CallingID, 2011; Chen, Huang, Chen, & Chen, 2009; Dodge et al., 2007; EarthLink Tool, 2011; GeoTrust, 2011; Google, 2011; He et al., 2011; NetCraft, 2011; SpoofGuard, 2011). There are four main topics in anti-phishing research (Zhang, Hong, & Cranor, 2007): understanding why people fall for phishing attacks; methods for educating people in order not to fall for phishing attacks; user interfaces for helping individuals in making better judgments about trustable email and legitimate web sites; and automated tools for detecting phishing.

Among the four topics, designing automated tools for detecting detecting phishing is today the focus of intense research. Approaches to the design of these tools can be categorized in four types: blacklist, white-list, heuristic, and hybrid.

- *Blacklist approach*: In the blacklist approach all web sites recognized as fraudulent web sites are listed in a list, referred to as blacklist. Since web sites are added into the blacklist after verifications, users can be sure of the illegitimacy of the web sites which cause warnings. But it takes a great deal of resources and time to maintain the blacklist. Furthermore, since fraudulent sites continously emerge, it is hard to keep the blacklist up to date.

[*] Corresponding author at: Software School, Fudan University, Shanghai 201203, China.

E-mail address: wlhan@fudan.eud.cn (W. Han).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.02.020

- *White-list approach*: Unlike the blacklist approach, the white-list approach maintains a list containing all legitimate web sites. Any web sites that do not appear the list are recognized as potential malicious web sites. Thus the white-list approach requires to list all legitimate web sites in the world and to keep the white-list up to date.

  The current white-list tools usually use a global white-list where all legitimate web sites are required to be included in the white-list. But it is obviously impossible for the administrator of the white list to cover the information of all legitimate web sites in the Internet. Thus, when such types of tools alerts, users will not be sure whether the current web site is an illegitimate one or is a legitimate one whose information is not contained in the white-list in time.
- *Heuristic approach*: The heuristic approach, adopted by the majority of anti-phishing tools, leverages the characteristics of a web site to decide the legitimacy of the web site. In a heuristic approach, web sites that have high similarity or tight relationship with legitimate web sites but actually are not the original ones are recognized as fraudulent web sites. The similarity or relationship of a web site with the legitimate ones is computed based on information collected on the legitimate web sites, referred to as a feature library (Chen et al., 2009).
- *Hybrid approach*: A hybrid approach combines the above approaches, such as a global white list and some heuristic approaches (Xiang & Hong, 2009), or a combination of a heuristic approach and a blacklist approach (eBay Toolbar's Account Guard, 2011), to recognize phishing pages.

Several experiments carried out by Zhang, Egelman, et al. (2007) have shown that the current automated tools are not effective in protecting not provide the users' digital identities.

This paper, therefore, proposes an approach, referred to as Automated Individual White-List (AIWL), to protect user's web digital identities. Although a global white-list approach is unpractical, we argue that an individual white-list approach is practical, because an individual white-list approach records the familiar legitimate web sites of a user rather than all the legitimate web sites in the world. The study of Florencio and Herley (2007) and our experiments in Section 4.3 show that a user only logs in a limited and stable number of web sites. AIWL, therefore, takes advantage of these observations to build an individual white-list to defend users against the theft attacks of web digital identities efficiently.

The main contributions of AIWL are as follows:

- AIWL is the tool that employs an individual white-list, automatically maintained by a Naïve Bayesian classifier, to protect user's web digital identities. In AIWL, any web site that does not match the individual white-list is classified as a fraudulent web site, and AIWL will alert the user who is trying to submit his or her account information to such a web site. Compared with the traditional blacklist approach and global white-list approach, this individual white-list approach is more practical.
- AIWL offers an effective solution to defend users against pharming attacks, including dynamic pharming (Karlof, Tygar, Wagner, & Shankar, 2007). AIWL keeps track of the features of login pages (e.g., IP addresses, Document Object Model (DOM) paths of input widgets) in the individual white-list to detect these attacks. AIWL can recognize pharming by checking the IP addresses of web sites. In addition, AIWL is able to effectively defend users against dynamic pharming by checking the Document Object Model (DOM) paths of the input widgets in the web page. Because the dynamic pharming attack embeds a legitimate login web page into the phishing site, the DOM paths will be modified, and thus AIWL can detect the attack based on such modification.

The rest of the paper is organized as follows: Section 2 introduces some background knowledge needed for the discussion in the paper; Section 3 describes AIWL in details Section 4 reports experimental results and user studies concerning the efficiency of AIWL; Section 5 analyzes some important issues in AIWL and discusses the limitations of AIWL; Section 6 introduces related work; and Section 7 outlines the conclusions and our future work.

2. Background

2.1. Phishing and pharming

A phishing attack (APWG, 2011; Fette et al., 2007) usually involves sending a user a fake e-mail claiming to be from a legitimate web site, leading the user to a fraudulent web site which looks very similar to the legitimate one, and tricking the user into exposing his or her web digital identity. Once the user submits his or her account information to such a fraudulent web site, the attackers are able to impersonate the victim and steal victim's personal information, such as financial information.

Pharming is a special kind of phishing. It is harder to detect and, of course, to defend against. By DNS (domain name server) hijacking or poisoning, pharming crimeware misdirects a user to a fraudulent web sites or a proxy server. Note that during a pharming attack the browser's address bar displays the genuine URL of a legitimate site. Therefore, it is more difficult for a user to distinguish a pharming web site from a legitimate one. An even more difficult attack to defend against is dynamic pharming, described by Karlof et al. (2007). Such an attack hijacks a DNS, takes advantage of the $\langle iframe \rangle$ tag to copy a legitimate web page in its own malicious page, and uses a Javascript to monitor user's interactions with the copy of the legitimate web site page.

2.2. Naïve Bayesian classifier

The Naïve Bayesian classifier (Androutsopoulos, Koutsias, Cbandrinos, & Spyropoulos, 2000; Bouchaala, Masmoudi, Gargouri, & Rebai, 2010; Duda & Hart, 1973; Mitchell, 1997; Pavon, Diaz, Laza, & Luzon, 2009) is considered one of the most effective approaches for learning how to classify text documents. Given a set of classified training samples, an application can learn from these samples so as to predict the class of an unmet sample.

Naïve Bayesian classifiers are widely used in anti-spam filtering in order to distinguish legitimate email messages from spam (Androutsopoulos et al., 2000; Sahami, Dumais, Heckerman, & Horvitz, 1998). Each email is represented by a feature vector $\vec{x} = (x_1, x_2, x_3, \ldots, x_n)$ where all features x_1, x_2, $x_3, \ldots, x_n$ are independent from each other. Each feature $x_i (1 \leqslant i \leqslant n)$ takes a binary value (0 or 1) indicating whether the corresponding property appears in the email. For example, x_1 is set to 1 if the email has a specific property (e.g., the presence of the *advertising* keyword); otherwise, x_1 is set to be 0. Given the feature vector $\vec{x}$ of an email, by applying the Bayes' theorem, we can calculate as follows the probability that the email belongs to a category c (spam or legitimate):

$$P(C = c|\overrightarrow{X} = \vec{x}) = \frac{P(C = c) \cdot P(\overrightarrow{X} = \vec{x}|C = c)}{\sum_{k \in \{spam, legitimate\}} P(C = k) \cdot P(\overrightarrow{X} = \vec{x}|C = k)} \quad (1)$$

Because $x_1, x_2, x_3, \ldots, x_n$ are assumed to be independent, we can calculate $P(C = c|\overrightarrow{X} = \vec{x})$ as:

$$\frac{P(C = c) \cdot \prod_{i=1}^{n} P(X_i = x_i|C = c)}{\sum_{k \in \{spam, legitimate\}} P(C = k) \cdot \prod_{i=1}^{n} P(X_i = x_i|C = k)} \quad (2)$$

where $P(X_i = x_i|C = c)$ and $P(C = c)$ can be calculated easily from training samples.

The Naïve Bayesian classifier has been proved to be very effective by a large number of empirical studies (Domingos & Pazzani, 1996; Mitchell, 1997; Langley, Wayne, & Thompson, 1992). AIWL leverages the Naïve Bayesian classifier to automatically identify legitimate web sites.

3. Automated individual white-list approach

3.1. Construct an individual white-list

To construct an individual white-list for a user, the familiar legitimate web sites of the user should be identified. In AIWL, we assume that the web sites where an individual user has successfully accessed the anticipatory services after submitting his or her account information are familiar legitimate web sites for the user. The reason is that the aim of malicious web sites is stealing user's web digital identities. The malicious web sites would not provide the same services as the legitimate web sites they forged, because it is hard for attackers to get the personal data of the user from the servers of legitimate web sites and provide the user the services which make the user believe he or she are interacting with the legitimate web sites. For instance, a malicious web site that forges ebay.com can forge a login page that has high visual similarity as the one of ebay.com, but it cannot provide the trade data for users, because it is hard for the attacker to get those data from ebay.com. Even though the attacker can get the data, getting these data would greatly increase the cost of the attacks, and for the attackers it would not be convenient to pay such high costs for short-lived (Fette et al., 2007) malicious sites. To verify this assumption, we have checked 100 phishing web sites from PhishTank.com (PhishTank, 2011)[1] and found that none of them provide users with the services of the legitimate web sites they forge. Therefore, it is reasonable to use a successful login process to build the individual white-list. AIWL leverages a Naïve Bayesian classifier to recognize a successful legitimate login process, and then a legitimate web site.

3.1.1. Features used in classification

Based on our investigation on current web sites, we represent each login process by a number of common features, namely: **Inbrowserhistory, HasNopasswordField, Numberoflink, HasNoUsername and Opertime**.

- **Inbrowserhistory**: Inbrowserhistory indicates whether a user has visited the current web site before by checking the browser's (e.g., Internet Explorer) history. Because phishing web sites are always short-lived (Fette et al., 2007), a web site already being visited is more likely to be a familiar web site of the user.
- **HasNopasswordField**: HasNopasswordField represents whether the web page redirected after the login process has a password field. In the usual case, if a user submits his or her account information to a web site and logs in successfully, the user will be directed to a functional page that provides services to the user. The password field will not be displayed again in this functional page. In contrast, if the login process fails, the user is always asked to fill the account information, e.g., username/password again and resubmit it, which makes the password field appear in the web page redirected after a login process. Thus, a login process followed by a web page without password field is likely to be a successful login process.
- **Numberoflink**: Numberoflink represents the number of links appearing in the web page redirected after the login process. If a user submits his or her account information to a web site and logs in successfully, the redirected page always contains a number of links to provide various kinds of services to the user. On the contrary, if the login process fails, the web page always contains a simple retry form or a warning message that has fewer links than a functional page. Thus, a login process followed by a web page containing more links is more likely to be a successful login process. In our Naïve Bayesian Classifier in AIWL, Numberoflink is a boolean value which represents whether the number of links is higher or lower than a pre-defined threshold. We determine the optimum threshold by our experiments.
- **HasNoUsername**: HasNoUsername indicates whether the web page redirected after the login process has the username already filled in the text field. In the usual case, the username/password field is always provided again after a failed login process for user's retrial. In many web sites, the username, which is the same as the previously entered one, is automatically filled for the user in the retry form. Thus, if there is no username filled in the text field of the web page redirected after a login process, the login process is likely to be a successful one.
- **Opertime**: Opertime represents how much time a user takes in an entire login process from when submitting account information to when finishing the session. In the usual case for failed login processes, the user is led to a warning page or a retry login page after submitting the account information. Depending on the specific case, the user would finish the current session immediately by closing the page or begin another login process. Thus, the login process would not take a long time. On the contrary, if the login process is successful, the user will stay in the web site for a longer time to use the services in the web page. Thus, a login process with longer Opertime is more likely to be a successful one. In our Naïve Bayesian classifier in AIWL, Opertime is a boolean value which represents whether the operation time is higher or lower than a pre-defined threshold. An experiment was conducted to determine the optimum threshold.

3.1.2. Naïve Bayesian classifier in AIWL

We use a Naïve Bayesian classifier to learn how to accurately identify successful login processes.

Each login process is represented with the vector $\vec{x} = (x_1, x_2, x_3, x_4, x_5)$, where x_1 represents whether Inbrowserhistory is true or false; x_2 represents whether HasNopasswordField is true or false; x_3 represents whether Numberoflink is larger than a threshold; x_4 represents whether HasNoUsername is true or false; x_5 represents whether Opertime is larger than a threshold.

The following probability can be easily calculated from the training samples where C is the category-denoting variable.

- $P(C = success)$ refers to the probability of successful login processes in all samples.
- $P(C = fail)$ refers to the probability of failed login processes in all samples.
- $P(X_i = x_i|C = success)$ refers to the probability of each feature x_i being present in a successful login process.
- $P(X_i = x_i|C = fail)$ refers to the probability of each feature x_i being present in a failed login process.

By substituting the above probabilities into Eq. (3), we can calculate the probability of a login process with the vector $\vec{x}$ belonging to *success* category as:

[1] PhishTank (http://www.phishtank.com/) is a free community anti-phishing site where users can submit suspicious web sites. After verification, the actual phishing sites are added into the blacklist of PhishTank.

$$P(C = success|\overrightarrow{X} = \vec{x}) = \frac{P(C = success) \cdot \prod_{i=1}^{5} P(X_i = x_i|C = success)}{\sum_{k\in\{success,fail\}} P(C = k) \cdot \prod_{i=1}^{5} P(X_i = x_i|C = k)} \quad (3)$$

AIWL leverages the Naïve Bayesian classifier to maintain the individual white-lists. The experiment in Section 4 show that the classifier efficiently identify login processes and build the individual white-lists as well.

3.2. Login user interface

In AIWL, we record LUI (Login User Interface) information in the individual white-lists. The LUI information refers to the features of the web page where the user inputs his or her account information.

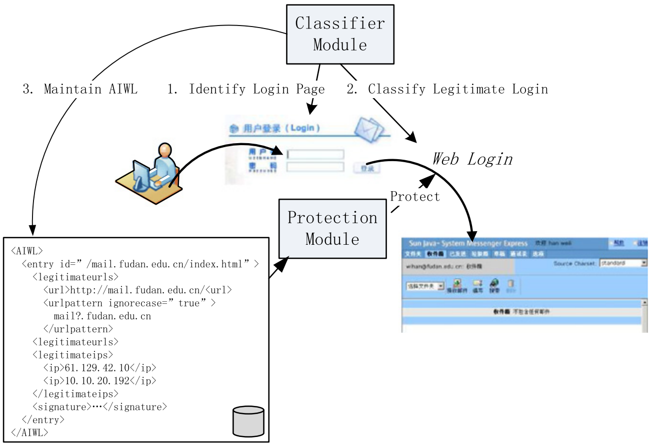
The LUI information in AIWL includes the following information for a web site:

- **URL**: URL refers to the Unified Resource Locator of the web site. It is the basic information about the web site. We use the URL as an index to organize LUI information in the individual white-list.
- **InputArea**: InputArea includes the FormUsernamePath and FormPasswordPath, which record the Document Object Model (DOM) paths of the input widgets in the web page. For example, FormUsernamePath is usually expressed as "*mainframe/loginform/username*". This feature can help AIWL in detecting dynamic pharming (Karlof et al., 2007), although AIWL is not specifically designed to defend only against dynamic pharming. Dynamic pharming takes advantage of the ⟨*iframe*⟩ tag to embed a legitimate web page in its own malicious page so as to monitor user's interactions with the legitimate web site. However, AIWL can detect dynamic pharming attack efficiently by comparing the current InputArea information with the pre-stored one in the individual white-list, because the DOM paths are changed during dynamic pharming.
- **IPs**: IPs is a list of legitimate IP addresses mapping to a URL. It is difficult for a user to distinguish pharming web sites, because pharming web sites have the same URLs and the same visual features as the legitimate ones. However AIWL can detect pharming immediately by matching IP addresses. All the IP addresses mapping to the domain are obtained and included in the IPs.

We use the above information to describe the login user interface so that AIWL can effectively detect different kinds of the theft attacks of web digital identities.

3.3. The Framework of AIWL

As is shown in Fig. 1, AIWL consists of two main components: the classifier module and the protection module.



```
<AIWL>
  <entry id=" /mail.fudan.edu.cn/index.html" >
    <legitimateurls>
      <url>http://mail.fudan.edu.cn/<url>
      <urlpattern ignorecase=" true" >
        mail?.fudan.edu.cn
      </urlpattern>
    <legitimateurls>
    <legitimateips>
      <ip>61.129.42.10</ip>
      <ip>10.10.20.192</ip>
    </legitimateips>
    <signature>…</signature>
  </entry>
</AIWL>
```

Fig. 1. Basic framework of AIWL.

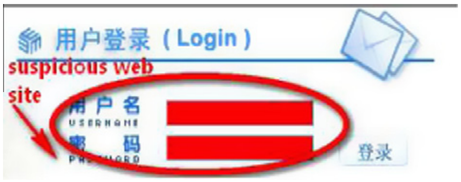


Fig. 2. Colored input widgets of a suspicious web site.

- **Classifier module**: The classifier module uses a Naïve Bayesian classifier to maintain the individual white-lists. Every time a user finishes a login process, the classifier module collects the features of the login process and use the Naïve Bayesian algorithm to determine whether the user has logged in this web site successfully. If so, this web site is believed to be a familiar web site of the user and the LUI information of the web site will be collected and added in the white-list. Details about the use of the Naïve Bayesian classifier in AIWL are given in Section 3.1.2.
- **Protection module**: When a user tries to submit his or her account information into a web site, the protection module will check whether the URL of the web site is in the white-list. If not, it means that the current web site is an unfamiliar one to the user, and thus AIWL will color in red the input widgets to warn the user. If the URL of the current web site is in the white-list, the LUI information of the current web site is collected and compared with the pre-stored LUI information in the white-list. If the LUI information matches, the web site is believed to be a legitimate one and the input widgets are colored in green; otherwise, they are colored in red.
Fig. 2 shows the input widgets of a suspicious web site colored by AIWL. Most of the defense tools use pop-up warnings to alert a user. However, the overuse of pop-ups has reduced their ability to draw user's attention to real serious security problems (Florencio & Herley, 2005). Some other defense tools use toolbars which show different types of marks to remind the user of the security level of the current web site. However, some studies have shown that these security indicators are ineffective against the high-quality theft attacks of web digital identities (Wu, Miller, & Garfinkel, 2006). Thus, AIWL colors the input widgets of the web site which is the focus of the user's current task in order to provide a stronger signal than toolbar indicators. Our user study, reported in Section 4.4, shows that coloring input widgets has better effect on alerting the user than toolbar indicators and pop-up warnings.

4. Experimental results

4.1. Constructing the Naïve Bayesian classifier

The Naïve Bayesian classifier was constructed to enable AIWL to recognize a successful login process. We simulated login processes for 34 web sites. 18 of 34 web sites are phishing web sites from PhishTank.com (PhishTank, 2011). The other 16 web sites are legitimate web sites. For every legitimate web site, both the successful login process and the failed one were simulated. We simulated failed login processes by purposely using wrong passwords. Thus, there are altogether 50 login processes acting as training samples. We designed a data-collecting tool which works as a plug-in for

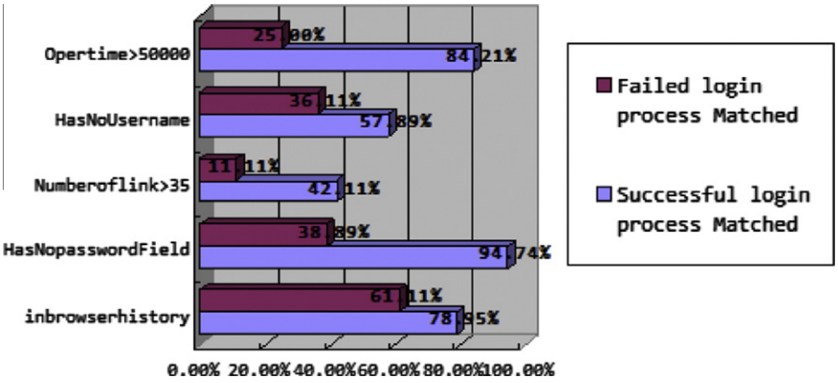


Fig. 3. Percentage of login processes matching the features.

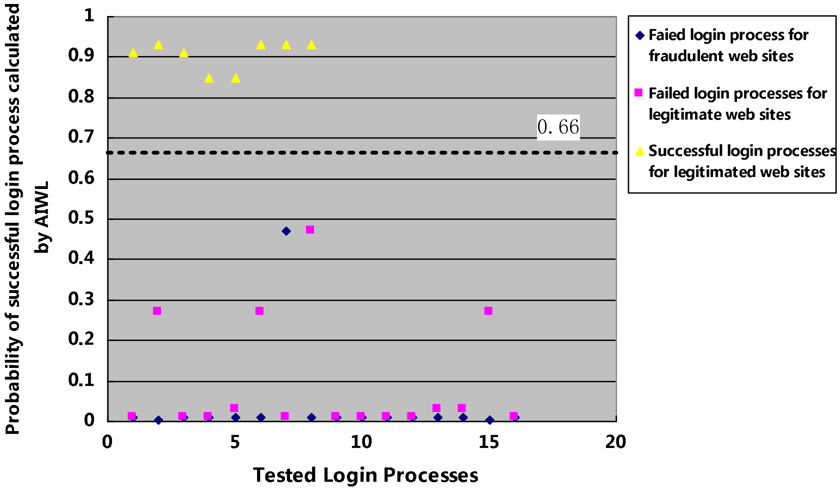


Fig. 4. Threshold of classification by AIWL.

Internet Explorer. Every time a login process was completed, the features listed in Section 3.1.1 were collected and the result of the login process, which was successful or failed, was specified manually. After all training samples of login processes had been simulated, the results were analyzed and the probabilities listed in Section 3.1.2 were calculated. All parameters required to construct to the Naïve Bayesian classifier in Eq. (3) are shown in Fig. 3, which shows the percentages of login processes matching each of the five features. In Fig. 3, the threshold for Numberoflink is set to 35 and the threshold for Opertime is set to 50000. These two thresholds were determined after repeated experiments for the best performance in classification.

When we analyzed the web sites to determine the threshold of Numberoflink, we found that different web sites had different design styles. For example, a web site may display 544 links after a user successfully logs in, and dispaly 77 links after the user fails to login in, whereas another web site may display only 19 links after a user successfully logs in, and only 2 links after the user fails to login in. Thus, setting the threshold of Numberoflink to 35 results in matching only 42.11% of the successful login processes. Although the match rate of 42.11% is small, it works effectively in the Naïve Bayesian classifier as shown in Section 4.2.

4.2. Efficiency of the login process classification

In this section, we evaluate the efficiency of AIWL in classifying login processes. We simulated login processes in various web sites and examined whether the classifier module of AIWL correctly classifies these login processes.

In the experiment, the classifier module of AIWL worked in the back-end of a browser (Internet Explorer) to collect the features listed in Section 3.1.1 after each login process was completed. Then the result of the training was used to calculate the probability of the login process to be a successful one.

We first used 40 login processes as samples to decide the threshold for the probabilities. 16 of the 40 samples were phishing sites, while the other 24 were legitimate ones. For those legitimate web sites, we simulated either the successful login processes (8 sites) or the failed ones (16 sites). We simulated a failed login process by purposely using a wrong password.

Fig. 4 shows the result of the experiment. Each point in Fig. 4 represents one login process: a triangle represents a successful login process for a legitimate web site; a square represents a failed login process for a legitimate web site; and a diamond represents a failed login process for a phishing site. Note that, in our experiment, we did not find any successful login processes for phishing sites, because, as we stated in Section 3.1, fraudulent web sites would not provide services to users. The height of a point in Fig. 4 indicates the probability of whether a login process is identified by AIWL to be successful. Comparing the login process results estimated by AIWL and the actual login results, we found that all actually successful login processes have a higher probability than the actually failed login processes; and there is a wide blank area between all successful login processes and failed login processes,

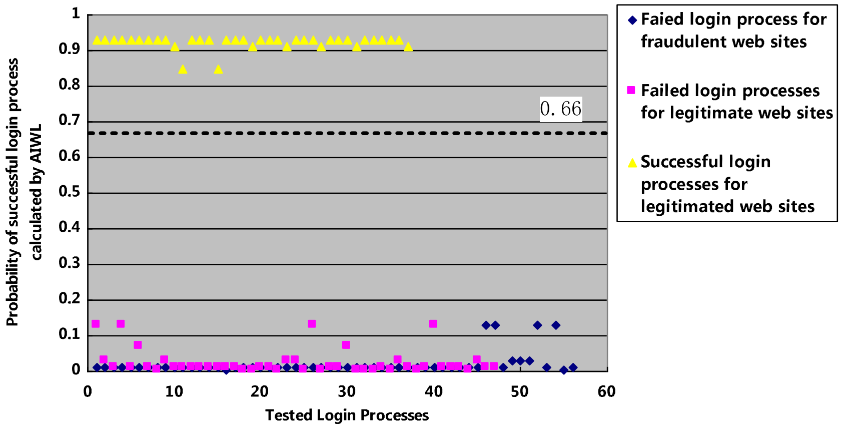


Fig. 5. Result of classification by AIWL.

no matter whether these failed login processes are for either legitimate web sites or phishing sites.

The threshold of the login process classification has thus been set to be 66%,which is a middle value of the blank area in Fig. 4, and it distinghishes successful login processes from failed login processes. That is, if the probability of a successful login calculated by the Naïve Bayesian algorithm is more than 66%, we believe this login process to be a successful one.

We used this threshold to test AIWL's classification efficiency in other 56 phishing web sites and 84 legitimate web sites. For every legitimate web site, we also simulated either the successful login processes (37 sites) or the failed ones (47 sites). As can be seen from Fig. 5, all the successful login processes are higher than the threshold and all the failed login processes are lower than the threshold.

We use true positive and false positive to evaluate the efficiency of the AIWL's classifier. True positive refers to correctly identifying a successful login process as a successful one and false positive refers to incorrectly labeling a successful login process as a failed one. The higher the true positive is, the more effective the classifier is. The lower the false positive is, the more efficient the classifier is.

Therefore, based on the threshold defined above, the result of true positives and false positives of the classifier module of AIWL for classifying login processes is perfect. The true positive is 100% and the false positive is 0%. It means that AIWL can recognize all successful login processes as successful ones and all failed login processes as failed ones. Thus, we can conclude that the classification performed by AIWL is basically perfect.

4.3. New login pages problem

The new login pages problem arises when a user submits his or her account information to a new legitimate web site for the first time. In such a situation, AIWL will alert the user, although the current web site is legitimate, because the information of the web site is not contained in the white-list.

The new login pages problem obviously exists, because it is possible for a user to create and use a new account at online services. However according to the experiment discussed in this section, the inconvenience for the user is very limited in time, because the user's familiar login pages are limited and stable.

We conducted this experiment to observe how many new web sites users log in daily. 28 people participated in this experiment. We designed a data-collecting tool working in the back-end as a plug-in of Internet Explorer to keep track of users' login records. Every time a user logged in a web site, our tool recorded the date of that time and the URL of the web site in an XML file.

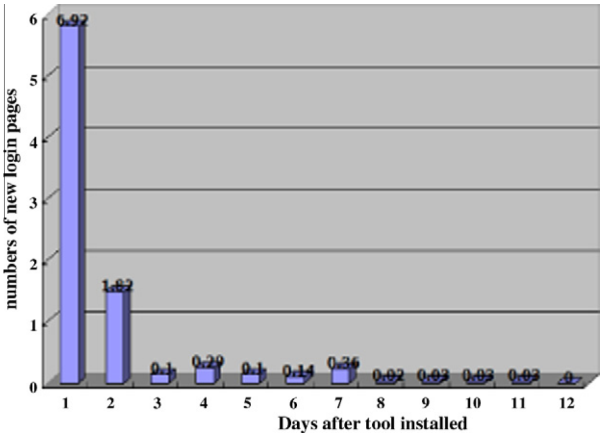


Fig. 6. Number of new login pages for users per day.

After a few of days, we collected and reviewed the XML file of each participant, and only kept the new login record for each day. For example, Shirley (one of the participants) had a login record for http://mail.yahoo.com.cn.

We calculated the number of new login pages that each participant encountered everyday, and then the average number of the 28 people was computed. Fig. 6 shows the number of users' new login pages per day. The number on the top of the column is the variance of the dataset for each day.

As is shown in Fig. 6, the number of new login pages for a user decreased quickly. In the last few days, it decreased to negligible. It means that a user has only a limited number of frequently logged in web sites. Thus, after the initial assessment, AIWL could cover most of the familiar web sites of a user and give fewer wrong warnings.

4.4. User study

4.4.1. Goals and overviews

We invited 20 volunteers to help us to evaluate the performance of AIWL. After a brief direction, the volunteers were asked to install AIWL in their local machine and use them independently. After the experiment ended, we asked volunteers to fill questionnaires about AIWL to get their feedbacks. Our goal of the study was to investigate:

- What do the users think of our security indicators used to color the input widgets of login page?

Table 1
Participants demographics.

Gender	Male	7
	Female	13
Age	18–21	6
	22–25	10
	26–30	3
	>30	1
University student	Yes	10
	No	10

- What do the users feel about the initial phase for AIWL to contain all information of the users' familiar LUIs?
- What are the users' feelings about wrong warnings because of the legitimate updates of the legitimate web sites?
- What are the users' opinions about using AIWL? Does AIWL help in building users' confidence with respect to the use of online services, and e-business services?

4.4.2. Participants demographics

Twenty volunteers participated to our study. The participants were required to be familiar with both the Windows operating system and the Internet Explorer browser. The participants were selected among subjects that frequently use digital identities in their daily life. Table 1 summarizes our participant's demographic characteristics. Half of our participants were university students while others were employees in companies. Of the students, 40% were undergraduates and 60% were graduate students.

4.4.3. Results

- 20% of the participants reported that they had been subject of the theft attacks of web digital identities and AIWL defended them from such attacks. Other 60% of the participants said that they had been subject of the theft attacks of web digital identities but they can detect those attacks without the help of AIWL. The reason is that some of the participants are students majoring in software engineering and with a professional knowledge about such kind of attacks.
- 85% of the participants thought that coloring input widgets is more effective than toolbar indicators and friendlier than popup warnings. They agreed that the use of color it is an effective approach for marking potentially malicious web sites.
- With respect to the training time of AIWL, 70% of the participants reported that it took less than 7 days for their white-lists to include all information of their familiar LUIs. Only 5% of the participants took more than 10 days to build their own individual white-lists.
- Of the 20 participants, 30% said that AIWL never warned them about the legitimate web sites that had already been added to the white-lists. Other 65% participants did not notice whether AIWL had warned them about the legitimate web sites that had already been added to the white-lists.. None of them reported that AIWL gave out wrong warnings for legitimate web sites because of legitimate updates to those web sites.
- Finally, 85% of the participants was confident that AIWL can help them in protecting their digital identities. With the help of AIWL, they felt more secure in their use of e-business services.

Thus, we can conclude that AIWL is an efficient and practical tool to protect users' web digital identities and can help users build the confidence in using online services, especially e-business services.

5. Discussion

5.1. Efficiency in identifying login processes

AIWL uses Inbrowserhistory, HasNopasswordField, Numberoflink, HasNoUsername and Opertime as the features to identify successful login processes. With those features, AIWL can classify login processes in 100% true positive and 0% false positive. That is, all login processes that AIWL recognizes as successful login processes are actually successful login processes and all login processes that AIWL recognizes as failed login processes are actually failed login processes. This perfect result is based on the behavior of current login web sites, because the features used to represent a login process were carefully chosen to model current web sites' behavior.

It is reasonable to choose these features during classification. We have studied 100 phishing sites in the PhishTank.com (PhishTank, 2011) about the three features: HasNopasswordField, Numberoflink,HasNoUsername to investigate the behavior of current phishing sites. Those features are not influenced by different users. We found that among those 100 phishing site, 84% of them had password fields in the web pages redirected after the login processes. The average number of links appearing in the web pages redirected after the login processes was 20, which wass far less than the threshold that is set in the experiment in Section 4.1. Finally, 83% of the phishing sites had usernames already filled in the text fields in the web pages redirected after the login processes. In the future work, we will investigate how keep track of the behavior of malicious web sites and correspondingly adjust AIWL to maintain high the classification efficiency of AIWL.

5.2. Limitations of the individual white-list

It is obvious that the white-list itself is the key element of our approach. If the white-list is compromised or lost, the whole approach would not work reliably. Because AIWL is installed at a local machine, it is difficult for AIWL to defend against Trojan Horses and viruses in the local machine that may modify the white-list. One possible solution is to store the white-list in a more secure device, e.g., a smart phone (Cao, Han, & Le, 2008). When accessing the white-list, the client module installed on the PC uses blue-tooth to communicate with the corresponding smart phone. The white-list can be protected more securely in this way.

Furthermore, AIWL has a synchronization problem. That is, if a user has more than one machine, the user must maintain multiple copies of the individual white-list. Synchronization based on a central server could be a potential solution. But, as is described in the previous paragraph, if we use a mobile phone to store the individual white-list, then the synchronization problem will not be a problem, because the AIWL instances installed at the various machines can access the same mobile phone to get the user's individual white-list.

5.3. Wrong warnings led by LUI change

An important issue of AIWL is wrong warnings resulting from legitimate changes to LUI information. It means that if LUI information of legitimate web sites has been changed for some reasons caused by the legitimate web sites rather than attackers, AIWL would mistake those legitimate web sites for fraudulent ones, because the LUI information does not match the information in the white-list.

We have conducted some experiments to observe the change rate of the LUI information for 15 widely used login sites (Florencio & Herley, 2007). The popular web sites are: aol.com, bebo.com, ebay.co.uk, ebay.com, google.com, hi5.com, live.com, match.com, msn.com, myspace.com, passport.net, paypal.com, yahoo.co.jp, yahoo.com, youtube.com. After several months of observations, we found that none of the web sites had changed their DOM paths in this period of time. So we can conclude that DOM paths in the LUI of a legitimate web site are stable and can be used to check the validity of the web site with low probability of issuing wrong warnings. But for IP addresses, we found that some web sites changed their IP addresses frequently, which may cause wrong warnings that alert users to pharming attacks when users submit their account information to legitimate web sites. On the other hand, many organizations deploy scalable content distribution networks to deliver the data from a server which is the closest to the end user. Thus, though visiting the same web site, a user connection from different locations, e.g., at home or at office, or at different times, will get different IP addresses. We also designed an experiment to collect and check the mobility of LUI information for legitimate web sites. As a result, we found that if the user changes the geographic location where he or she connects wrong warnings would occur.

This problem could be solved by integrating AIWL with other anti-phishing solutions or third party service providers. Who.is Who.is (2011) is such a service provider that runs a database query system providing information about domains, including registry status, corresponding IP address and contact information of domain name-holders. When an IP address mismatch occurs, AIWL could query Who.is for the latest IP address for the web site to confirm whether the change of the IP address is caused by the legitimate update or a pharming attack. Furthermore, the registry information of the web site being accessed by a user and the current IP address can be queried and compared to confirm the validity of the web site. If the registry information of the current IP address can be matched with the one of the current domain, it is very likely that the mismatch of IP address is caused by a legitimate update.

However integrating AIWL integrates with other servers may result in other security problems, because these servers could be compromised and provide wrong information. Thus, a trusted service provider or a trusted path to update the changed legitimate web sites would be required.

5.4. Attackers' countermeasures and analysis

AIWL identifies successful login processes using the following factors: Inbrowserhistory, HasNopasswordField, Numberoflink, HasNoUsername and Opertime. Any successful login process will result in the web site to be added into the individual white-list for a user. In such case, the attackers may fake their fraudulent web sites promptly to be adapted to the algorithm of AIWL to make the fraudulent web sites be added to the white-list. Even though such an attack seems possible, it would not work for the following reasons:

- For every web site that cannot match the white-list, AIWL will alert users about the possible attacks. Thus whenever users try for the first time to submit their account information to the fraudulent web sites, AIWL will alert the users. Furthermore, because fraudulent web sites are usually short-lived (Fette et al., 2007), users would not visit the same fraudulent web sites for the second time. Thus, even though the web site has been added to the white-list by a possible false positive, it would not do any actual harm to the users.
- Some factors are not easy to fake. For example, it is hard for attackers to be aware of the history record of a user. Even if they can, the cost would be so high that the attackers would not be able to sustain the cost. Whereas, Opertime is also hard to control for the attackers. As we discussed in Section 3.1, fraudulent web sites would not be able to provide the services as legitimate web sites, so they would not be able to attract a user to stay for a long time.
- For the factors that attackers may fake, i.e., HasNopasswordField, Numberoflink, HasNoUsername, we will continue to observe behaviors of fraudulent web sites in Phishtank.com and improve our algorithm to keep the efficiency of AIWL.

6. Related work

The problem of protecting from the theft attacks of web digital identities, especially phishing attacks, has been widely investigated from several different perspectives and several approaches exist.

First, the user's own security awareness is a very important factor in ensuring a safe and secure e-business environment. Therefore, the Anti-Phishing Working Group and other financial organizations have gathered a large amount of materials giving suggestions and guidelines to users in order to avoid becoming victims of the theft attacks of web digital identities. Simulated tests and games (EBay, 2007) are also used to educate users in a more interesting way. Kumaraguru, Sheng, Acquisti, Cranor, and Hong (2007) found that embedded training works better than the current practice of sending security notices.

However, Dodge et al. (2007) concluded that even when educated, users continue to disclose information to other unauthorized parties. Thus, a lot of automated tools (Aburrous et al., 2010; CallingID, 2011; EarthLink Tool, 2011; eBay Toolbar's Account Guard, 2011; Dhamija & Tygar, 2005; Fu, Liu, & Deng, 2006; GeoTrust, 2011; Google, 2011; He et al., 2011; SpoofGuard, 2011; NetCraft, 2011) have been developed to automatically detect phishing. Those tools, which are integrated into web browsers, either alert the user of the possible danger or use some well-marked symbols to mark the security level of the web sites. Zhang, Egelman, et al. (2007) have developed a testbed to test 10 popular anti-phishing toolbars and half of the tools they tested could only correctly identify less than 50% phishing sites and many tools were vulnerable to some simple exploits.

Some researchers explored the applications of white-list to protect web digital identities (Cao et al., 2008; Chen et al., 2009; Ronda, Saroiu, & Wolman, 2008; Xiang & Hong, 2009). Ronda et al. (2008) have developed an anti-phishing plug-in for Firefox, called iTrustPage, whose underlying approach is similar to AIWL. iTrustPage tries to find out whether the untrustworthy form of the web site is the form the user intended to fill. Unlike AIWL, iTrustPage validates every form used by the user, even a simple search form, which could frequently annoy the user. In contrast, AIWL just colors in green the login pages familiar to a user, and in red the login pages unfamiliar to a user. Recently, Xiang and Hong (2009) proposed a hybrid approach which combines a global white-list and some heuristic algorithms to recognize phishing pages. Compared with AIWL, the approach by Xiang et al., which leverage the global white-list as an additional feature to improve recognizing a phishing site, is a heuristic method, and has failures. Chen et al. (2009) suggested to store authentic pages in a global store, such as APWG, and recognize a phishing page by comparing the visual features of a real page with the ones of the authentic pages. The approach by Chen et al. is also a heuristic method which uses a global white list to store the features of authentic pages. Thus, the approach by Chen et al. has the same disadvantage of the approach by Xiang et al. false positivea. In addition, the approaches by Ronda et al., Xiang et al. and Chen et al. are not able to protect against pharming.

Some other solutions providing improved protection of users' digital identities by detecting pharming attacks. Li et al. proposed to detect phishing/pharming attacks by transforming the real e-banking system into a honeypot equipped with heneytokens (Li & Schmitz, 2009). Unlike existing anti-phishing tools, this solution aims at preventing the money in the bank from being stolen by the attackers rather than protecting users' digital identities. In Karlof, Shankar, Goto, and Wagner (2007), Karlof et al. proposed a new model for web authentication to defend againest phishing and pharming attacks. The browser cookies bounding to the originating server's public key are used as authenticators to check the legitimacy of the web sites that request for users' digital identities. All of those anti-pharming solutions need to change the existing authentication models and some of them even require changes of web servers, which makes them hard to be widely applied.

In addition, new browsers, e.g., IE8, provide users with a security feature which colors the address bar green, if the corresponding web site has a legitimate certificate. But, compared with AIWL, the mechanism has at least two disadvantages: first, the mechanism is not able to alert users when the web site, which could be a phishing site, has no certificate; second, the mechanism is not able to detect pharming, including dynamic pharming.

7. Conclusion and future work

This paper proposes an approach, called Automated Individual White-List (AIWL), to protect user's web digital identities. AIWL is effective in detecting the theft attacks of web digital identities by maintaining an automated individual white-list of all web sites familiar to the user together with the LUI information of these web sites. AIWL uses a Naïve Bayesian classifier to automatically build an individual white-list for the user. As is shown by our experiments, AIWL recognizes a successful login process efficiently so it can maintain an accurate individual white-list. Furthermore, one of our experiments shows that the web sites familiar to a user are usually in a small number and stable. Thus, after the initial assessment phase, the white-list in AIWL is stable and fits well the individual user. A significant advantage of AIWL is that by checking LUI information of a web sites, AIWL can recognize phishing, pharming, and even dynamic pharming.

In the future work, we will investigate how to use a mobile device (e.g., a smart phone) to store white-lists in a more secure environment. Furthermore, we will investigate methods to maintain the individual white-list. Last but not least, we will also investigate how to integrate our individual approach with existing heuristic approaches to improve the efficiency in defending the theft attacks of web digital identities.

Acknowledgements

This paper is partly supported by "211-Project Sponsorship Projects for Young Professors at Fudan", the 863 project (Grant No: 2011AA100701) and Key Lab of Information Network Security, Ministry of Public Security (Grant NO: C11601).

References

Aburrous, M., Hossain, M. A., Dahal, K., & Thabtah, F. (2010). Intelligent phishing detection system for e-banking using fuzzy data mining. *Expert Systems with Applications, 37*(12), 7913–7921.

Androutsopoulos, I., Koutsias, J., Cbandrinos, K. V., & Spyropoulos, C. D. (2000). An experimental comparison of Naïve Bayesian and keyword-based anti-spam filtering with personal e-mail messages. In *Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval (SIGIR 2000), Athens Greece* (pp. 160–167).

APWG. Anti-phishing working group (2011). <http://www.anti-phishing.org/>.

Bouchaala, L., Masmoudi, A., Gargouri, F., & Rebai, A. (2010). Improving algorithms for structure learning in Bayesian networks using a new implicit score. *Expert Systems with Applications, 37*(7), 5470–5475.

CallingID (2011). <http://www.callingid.com/DesktopSolutions/CallingIDToolbar.aspx>.

Cao, Y., Han, W., & Le, Y. (2008). Anti-phishing based on automated individual white-list. In *Proceedings of the 4th ACM CCS workshop on digital identity management*.

Chen, K. T., Huang, C. R., Chen, C. S., & Chen, J. y. (2009). Fighting phishing with discriminative keypoint features. *IEEE Internet Computing*, 56–63.

Chen, J., Wu, G., Shen, L., & Ji, Z. (2011). Differentiated security levels for personal identifiable information in identity management system. *Expert Systems with Applications, 38*(11), 14156–14162.

Dhamija, R., & Tygar, J. D. (2005). The battle against phishing: Dynamic security skins. In *Proceedings of the 2005 symposium on Usable privacy and security, Pittsburgh, Pennsylvania* (pp. 77–88).

Dodge, R. C., Jr., Carver, C., & Ferguson, A. J. (2007). Phishing for user security awareness. *Computers & Security, 26*.

Domingos, P., & Pazzani, M. (1996). Beyond Independence: Conditions for the optimality of the simple Bayesian classifier. In *Proceedings of the 13th international conference on machine learning, Bari, Italy* (pp. 105–112).

Duda, R. O., & Hart, P. E. (1973). *Bayes decision theory. Chapter 2 in pattern classification and scene analysis*. John Wiley, pp. 10–43.

EarthLink Tool (2011). <http://www.earthlink.net/software/free/toolbar/>.

EBay (2007). Spoof email tutorial. <http://pages.ebay.com/education/spooftutorial/Visited>.

eBay Toolbar's Account Guard (2011). <http://pages.ebay.com/help/confidence/account-guard.html>.

Fette, I., Sadeh, N., & Tomasic, A. (2007). Learning to detect phishing emails. In *Proceeding of international world wide web conference (WWW 2007), Banff, Alberta, Canada* (pp. 649–656).

Florencio, D., & Herley, C. (2005). *Stopping a phishing attack, even when the victims ignore warnings, microsoft research (MSR)*. Tech. Rep. MSR-TR-2005-142.

Florencio, D., & Herley, C. (2007). A large-scale study of web password habits. In *Proceeding of international world wide web conference (WWW 2007), Banff, Alberta, Canada* (pp. 657–665).

Fu, A. Y., Liu, W., & Deng, X. (2006). Detecting phishing web pages with visual similarity assessment based on earth mover's distance (EMD). *IEEE Transactions on Dependable and Secure Computing, 3*(4).

GeoTrust, Inc. TrustWatch Tool (2011). <<http://toolbar.trustwatch.com/tour/v3ie/toolbar-v3ie-tour-overview.html>.

Google. Google safe browsing for firefox (2011). <http://www.google.com/tools/firefox/safebrowsing>.

He, M., Horng, S. J., Fan, P., et al. (2011). An efficient phishing webpage detector. *Expert Systems with Applications, 38*(10), 12018–12027.

Karlof, C., Tygar, J. D., Wagner, D., & Shankar, U. (2007). Dynamic pharming attacks and locked same-origin policies for web Browsers. In *ACM CCS* (pp. 58–71).

Karlof, C., Shankar, U., Goto, B., & Wagner, D. (2007). *Locked cookies: Web authentication security against phishing, pharming, and active attacks*. Technical report, University of California at Berkeley, UCB/EECS-2007-25.

Kumaraguru, P., Sheng, S., Acquisti, A., Cranor, L., & Hong, J. (2007). *Teaching Johnny not to fall for phish*. CyLab Technical Report. CMU-CyLab-07-003.

Langley, P., Wayne, I., & Thompson, K. (1992). An analysis of Bayesian classifiers. In *Proceedings of the 10th national conference on artificial intelligence, San Jose, Califomia* (pp. 223–228).

Li, S., & Schmitz, R. (2009). A novel anti-phishing framework based on honeypots. In *Proceedings of the 4th annual APWG eCrime research summit (eCRS'2009), Tacoma, WA, USA*.

Mitchell, T. M. (1997). *Bayesian learning. Chapter 6 in machine learning*. McGraw-Hill, pp. 154–200.

NetCraft. netcraft anti-phishng toolbar (2011). <http://toolbar.netcraft.com/>.

Pavon, R., Diaz, F., Laza, R., & Luzon, V. (2009). Automatic parameter tuning with a Bayesian case-based reasoning system. *A Case of Study, Expert Systems with Applications, 36*(2), 3407–3420. Part 2.

PhishTank. <http://www.phishtank.com/>.

Ronda, T., Saroiu, S., & Wolman, A. (2008). iTrustPage: A user-assisted anti-phishing tool. In *Proceedings of the 2008 EnroSys conferece (EuroSys 2008), Glasgow, Scotland, UK* (pp. 261–272).

Sahami, M., Dumais, S., Heckerman, D., & Horvitz, E. (1998). A Bayesian approach to filtering junk email. In *AAAI workshop on learning for text categorization, Madison, Wisconsin*.

Sheng, S., Magnien, B., Kumaraguru, P., Acquisti, A., Cranor, L., Hong, J., et al. (2007). Anti-phishing phil: The design and evaluation of a game that teaches people not to fall for phish. In *Proceedings of the 2007 symposium on usable privacy and security, Pittsburgh, PA*.

SpoofGuard (2011). <http://crypto.stanford.edu/SpoofGuard/>.

Who.is (2011). <http://who.is/>.

Wu, M., Miller, R., & Garfinkel, S. (2006). Do security toolbars actually prevent phishing attacks? In *Proceedings of the SIGCHI conference on human factors in computing systems (CHI2006), Canada* (pp. 601–610).

Xiang, G., & Hong, J. I. (2009). A hybrid phish detection approach by identity discovery and keywords retrieval. In *Proceedings of the 18th internation conference on world wide web (WWW 2009), Madrid, Spain* (pp. 561–570).

Zhang, Y., Egelman, S., Cranor, L., & Hong, J. (2007). Phinding phish: Evaluating anti-phishing tools. In: *Proceedings of the 14th annual network & distributed system security symposium (NDSS 2007), San Diego, CA*.

Zhang, Y., Hong, J., & Cranor, L. (2007). CANTINA: A content-based approach to detecting phishing web sites. In *Proceeding of international world wide web conference (WWW 2007), Banff, Alberta, Canada* (pp. 639–648).