Anti-phishing Based on Automated Individual White-List

Ye Cao
Software School, Fudan University
825#, Zhangheng Road
Shanghai, P. R. China
072053004@fudan.edu.cn

Weili Han*
Software School, Fudan University
825#, Zhangheng Road
Shanghai, P. R. China
wlhan@fudan.edu.cn

Yueran Le
Software School, Fudan University
825#, Zhangheng Road
Shanghai, P. R. China
0561119@fudan.edu.cn

ABSTRACT
In phishing and pharming, users could be easily tricked into submitting their username/passwords into fraudulent web sites whose appearances look similar as the genuine ones. The traditional blacklist approach for anti-phishing is partially effective due to its partial list of global phishing sites. In this paper, we present a novel anti-phishing approach named Automated Individual White-List (AIWL). AIWL automatically tries to maintain a white-list of user's all familiar Login User Interfaces (LUIs) of web sites. Once a user tries to submit his/her confidential information to an LUI that is not in the white-list, AIWL will alert the user to the possible attack. Next, AIWL can efficiently defend against pharming attacks, because AIWL will alert the user when the legitimate IP is maliciously changed; the legitimate IP addresses, as one of the contents of LUI, are recorded in the white-list and our experiment shows that popular web sites' IP addresses are basically stable. Furthermore, we use Naïve Bayesian classifier to automatically maintain the white-list in AIWL. Finally, we conclude through experiments that AIWL is an efficient automated tool specializing in detecting phishing and pharming.

Categories and Subject Descriptors
C.2.0 [**Computer-Communication Networks**]: General – Security and protection; H.4.3 [**Information Systems Applications**]: Communication Applications – Information Browsers

General Terms
Design, Security

Keywords
Individual White-List, Naïve Bayesian Classifier, Login User Interface, Anti-Phishing, Anti-Pharming

1. INTRODUCTION
Recently, phishing (including pharming) is a severe attack to user's confidential information [1-3]. During phishing, the attacker tricks the user into submitting his/her confidential information (such as password) into a fraudulent web site that has high visual similarities as the genuine one. Thus, the user could expose his/her password, credit card number, bank account and other important information to the attacker. Phishing, therefore, could cause great loss to the user. As a result, phishing is also embarrassing the expansion of e-commerce, due to user's distrust of the whole e-commerce environment.

Most of the techniques for phishing detection are based on blacklist [30]. In the blacklist approaches, once the user visits a web site that is in the blacklist, he/she will be warned of the potential attack. But maintaining a blacklist requires a great deal of resources for reporting and verification of the suspicious web sites. In addition, phishing sites emerge endlessly [5], so it is difficult to keep a global blacklist up to date. Contrary to blacklist, white-list approach maintains a list containing all legitimate web sites. But a global white-list approach is likewise hardly used because it is impossible for a white-list to cover all legitimate web sites in the entire cyber world.

In this paper, we present a novel approach, named Automated Individual White-List (AIWL). AIWL uses a white list that records all familiar Login User Interfaces (LUIs) of web sites for a user. A familiar LUI of a web site refers to the characteristic information of a legitimate login page on which the user wants to input his/her username/password. Every time a user tries to submit his/her sensitive information into an LUI that is not included in the white-list, the user will be alerted to the possible attack.

Here, LUI refers to the user interface where user inputs his/her username/passwords. For instance, a typical LUI is composed of URL address, page feature, DNS-IP mapping. Once the user tries to submit the confidential information into a web site that is in the white-list, LUI information of current web site will be collected and compared with the pre-stored one in the white-list. Any mismatch will also cause warning to the user.

To conveniently set up the white-list in AIWL, we use the Naïve Bayesian classifier [8, 9] to identify a successful login process. After a web site has been logged in successfully several times, it is believed to be a familiar one of the user and the LUI information of the web site can be added to the white-list automatically after user's confirmation.

AIWL is an efficient approach for anti-phishing, because:

- The number of a user's frequently logged in web sites is usually limited [13]. According to our experiment in section 4.2, AIWL will almost cover all the familiar web sites of a

* Research partly supported by NSFC (GrantNO: 60703091)
Corresponding Author: Weili Han

DIM'08, October 31, 2008, Fairfax, Virginia, USA.

user so as to detect phishing sites efficiently with low probability of wrong warnings;

- The probability of wrong warnings is very low, because our experiments described in Section 4.3 and Section 4.4 show that LUIs of popular legitimate web sites are basically stable.

The rest of our paper is organized as follows: in section 2, we introduce background and motivation of the paper; section 3 introduces the overall approach of AIWL and discusses some important issues in the approach; section 4 describes the experiments for evaluation; section discusses the advantages of AIWL on the basis of its comparison with other solutions and consider the limitations of AIWL; section 6 introduces the related work; and section 7 summarizes our paper and introduces future work.

2. BACKGROUND AND MOTIVATION

2.1 Phishing, Pharming and Defense

Phishing attackers use both social engineering and technical subterfuge to steal user's identity data as well as financial account information [3]. By sending "spoofed" e-mails, social-engineering schemes lead users to counterfeit web sites that are designed to trick recipients into divulging financial data such as credit card numbers, account usernames, passwords and social security numbers. In order to persuade the recipients to respond, phishers often hijack brand names of banks, e-retailers and credit card companies. Furthermore, technical subterfuge schemes often plant crimewares, such as Trojan, keylogger spyware, into victims' machines to steal user's credentials.

Phishing attack not only leads to great loss to users but also influences the expansion of e-commerce. Rampant phishing attacks would cause the whole e-commerce environment to be dangerous and aggressive. Furthermore, it is difficult for common users to distinguish fraudulent web site from the genuine one. Thus, users would feel hesitant to use e-banking and online-shopping services in such an environment.

Pharming is a special kind of phishing. Pharming crimeware misdirects users to fraudulent sites or proxy servers typically through DNS hijacking or poisoning, so it is harder for a common user to distinguish pharming web sites from legitimate sites, because pharming web sites have the same visual features and URLs as the genuine ones. In [32], Karlof et al. described a new attack called dynamic pharming which hijacks DNS, takes advantage of the <iframe> tag to contain the legitimate web page in its own malicious page and designs a javascript to monitor user's interactions with the legitimate web site.

According to the study of Zhang et al. [2], there are four categories in past work of anti-phishing: studies to understand why people fall for phishing attacks; methods of training people not to fall for phishing attacks; user interfaces for helping people make better decision about trustable email and web sites; and automated tools to detect phishing.

AIWL proposed in this paper contributes a novel approach to the development of phishing detection tools by checking the LUI. Furthermore, AIWL can recognize pharming sites by checking the IP addresses of the web site. Once IP address of the current web site does not match with the pre-stored legitimate IP addresses, AIWL will alert the user to the possible pharming attacks. According to the experiment in section 4.3, the popular web sites' IP addresses are basically stable, so AIWL can detect pharming attack efficiently. Last but not least, AIWL can also defend against dynamic pharming effectively without any extra cost. AIWL records the DOM path of the input widgets in web page as the content of LUI in the white-list. Once the DOM path has changed, AIWL will detect that and alert the user. According to the experiment in section 4.4, the popular web sites' DOM paths in the login page are basically stable, so AIWL can also detect dynamic pharming efficiently.

2.2 Naïve Bayesian Classifier

The Naïve Bayesian classifier [8, 9] is thought to be one of the most effective approaches to learning of the classification of text documents. Given an amount of classified training samples, an application can learn from these samples so as to predict the class of the unmet sample using the Bayesian classifier.

Naïve Bayesian classifier is used in Anti-Spam filtering to classify an email as a good one or a Spam [6, 7]. Each email is represented by a feature vector $\vec{x} = (x_1, x_2, x_3, \ldots, x_n)$ where each of the property, $x_1, x_2, x_3, \ldots, x_n$ is independent. Each attribute x_i ($1<=i<=n$) is a binary attribute (0 or 1) that indicates whether the corresponding property exists or not. For example, x_1 is set to be 1 if the specific email has a pre-specific feature (maybe a keyword "notification").

Combining Bayesian and the theorem of total probability, it follows that with the vector $\vec{x}$ of an email, the probability that the email belongs to a category c is:

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

Because $x_1, x_2, x_3, \ldots, x_n$ is conditionally independent, we can compute $P(C = c \mid \vec{X} = \vec{x})$ as:

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

where $P(X_i \mid C)$ and $P(C)$ can be calculated easily from training samples.

The Naïve Bayesian classifier is proved to be very effective by a large number of empirical studies. [10, 11]

2.3 Motivation

In this paper, we introduce AIWL, a novel anti-phishing solution which uses an automated individual white-list to help user distinguish a familiar LUI from unfamiliar ones.

AIWL can efficiently defense phishing even pharming, because it can check the LUI when a user wants to submit his/her password

to a web site. Once the difference is detected, AIWL will alert user to the potential attacks. Due to the stability of user's familiar LUIs and features of LUIs, AIWL will prompt the wrong warning with very low rate.

Furthermore, the white-list in AIWL is automatically maintained by a Naïve Bayesian classifier. A certain amount of successful logins of the same web site will make the web site to be identified as a familiar one of the user. After that, the LUI information of the web site will be added to the white-list automatically after the user's confirmation. The user, therefore, can maintain a white-list with the least operations.

3. AUTOMATED INDIVIDUAL WHITE-LIST APPROACH FOR ANTI-PHISHING

3.1 Approach Overview

In a white-list approach, all legitimate web sites are listed; any web site that is not in the list is recognized as invalid and causes warning to the user. It's unpractical to build a global white-list of the whole World Wide Web because of the large scale and rapid increasing volume.

But an individual white-list is feasible. Because the experiment described in section 4.5 shows that a common user only logs in a limited number of web sites, AIWL makes use of this feature to build an individual white-list which records user's all familiar LUIs of web sites to defend against phishing attack efficiently.

Our work consists of two phases: training phase and practice phase. In training phase, we used a Naïve Bayesian classifier to train AIWL to identify a successful login process correctly by learning from labeled samples. In practice phase, AIWL makes use of the training result to automatically identify a familiar LUI of the user and maintain the white-list for the detection of phishing attacks.

Training Phase: In the training phase, we use a number of login processes as samples. Each login process is represented with the features described in section 3.2 and labeled as a successful login process or a failing one. AIWL uses a Naïve Bayesian classifier to learn from these labeled samples so that the classifier can label other processes correctly to build up a white list in practice phase.

Practice Phase: In the practice phase, AIWL maintains the white-list automatically and uses the white-list to detect phishing sites.

AIWL monitors user's login process and make use of the result in training phase to determine whether a login process is successful. If the user always logs in a same web site successfully, this web site is believed to be a familiar web site of the user, so AIWL will automatically collect and add the LUI information of this web site into the white-list after the user's confirmation.

Every time a user tries to submit his/her username/password to a web site, AIWL checks whether the LUI of the web site is contained in the white-list. If the LUI is not in the white-list, which means the LUI in the web site is an unfamiliar one to the user, the user will be alerted to the possible attacks. Only after user confirms to continue, will the information be submitted. If the URL is in the white-list, AIWL will submit the confidential information for the user.

3.2 Login User Interface

This section gives the exact definitions of Login User Interface (LUI). LUI is a user interface where the user inputs his/her username and password.

We define LUI as follows:

Definition 1: LUI = (URL, IPs, InputArea, CertHash, ValueHash)

In definition 1, URL means the valid Unified Resource Locator of the web site; CertHash is the hash code of the certificate for this web site, which is used when the page of URL is transferred by HTTPS; ValueHash records the hash code of the form's HTML source code in the web site. The definitions of IPs, InputArea are given in definition 2 and 3 respectively.

Definition 2: IPs = (IP1, IP2, …)

IPs is the list of the valid IP addresses mapping with certain URL in DNS. Some large-scale web sites have a number of IP addresses for load balancing; all of the IP addresses will be obtained and included in the IPs.

Definition 3: InputArea = (FormUsernamePath, FormPasswordPath)

The two elements in this definition record the DOM (Document Object Model) path of the input widgets in web page. For example, FormUsernamePath is usually expressed as "mainframe/loginform/username". The dynamic pharming attack [32] uses <iframe> tag to contain the legitimate web page in its own malicious web page, which changes the InputArea information of current web site. AIWL, therefore, can detect dynamic pharming attack efficiently by comparing current InputArea information with the pre-stored one in the white-list.

3.3 Features Used in Classification

Each login process is represented with certain features. These features include Inbrowserhistory, HasNopasswordField, Numberoflink, HasNoUsername and Opertime. This section defines and describes those features. These features were chosen based on the research of current login web sites' behavior.

3.3.1 Inbrowserhistory

Inbrowserhistory indicates whether the web site is in IE history. Phishing web sites are always short-lived [5]. A web site already visited is more likely to be a legitimate web site than a web site never previously visited, because it is nearly impossible for a user to visit a phishing site more than twice during its short life cycle.

3.2.2 HasNopasswordField

HasNopasswordField represents whether the web page redirected after the login process has the password field. In the usual case, if a user submits his/her confidential information to a web site and logs in successfully, the next page is a functional page that provides services to the user. The password field will not be displayed in this page. In contrast, if the login process fails, the user is always asked to fill the username/password and submit them again, which makes the password field appear in the web page redirected after a login process. So a login process followed by a web page which contains no password field is likely to be a successful login process.

3.2.3 Numberoflink

Numberoflink represents the number of links in the web page redirected after the login process. If a user submits his/her confidential 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 just contains a simple retry form and has fewer links than a functional page. So a login process followed by a web page containing a number of links is likely to be a successful login process. The Numberoflink feature is a Boolean value that refers to more or less than the threshold. We made repeated experiments for the optimum threshold.

3.2.4 HasNoUsername

HasNoUsername indicates whether the web page redirected after the login process has the username that was previously filled by the user in the LUI. In the usual case, the username/password field is always provided again after a failing login process for user's retry. In many web sites, the username, which is the same as the previously input one, could be filled automatically for the user in the retry form. So if there is no username in the web page redirected after a login process, the login process is likely to be a successful one.

3.2.5 Opertime

Opertime represents the time a user takes to stay in a certain web site. In the usual phishing case, the user is trapped to fill the username and password in the fraudulent site, and is led to an "Update Success" or a failure login page. It is the case that the settling time could not be long. User could close the web page immediately after reading the success message or several times of failing retrials. On the contrary, if the web site is a legitimate one, and the login process is successful, the user will stay in the web site for a longer period of time to use the services in the web page. So a web site attracting the user to stay for a longer time is more likely to be a legitimate web site. The Opertime is a Boolean value that refers to more or less than the threshold. The experiment was also conducted repeatedly for an optimum threshold.

3.4 the Naïve Bayesian classifier

AIWL use a Naïve Bayesian classifier to learn from the classified login processes for identifying successful login process accurately.

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 calculated easily from the training samples where C is the category-denoting variable.

- P (C=success) = probability of successful login processes in all samples.
- P (C= fail) = probability of failing login processes in all samples.
- P (x_i= 1 | C=success) = probability of feature x_i that presents in a successful login process
- P (x_i= 0 | C=success) = probability of feature x_i that does not present in a successful login process
- P (x_i= 1 | C= fail) = probability of feature x_i that presents in a failing login process
- P (x_i= 0 | C= fail) = probability of feature x_i that does not present in a failing login process

By putting above terms into the formula (3), we can calculate the probability of a login process with vector $\vec{x}$ belonging to a "success" category as:

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

4. EMPIRICAL RESULT

In this section, we present the detail of the experiments to evaluate the performance of AIWL. The training process and the result of the Naïve Bayesian classifier is described in section 4.1. In section 4.2, we evaluate the effectiveness of AIWL in classifying login process. Other three experiments are performed to demonstrate some important issues of AIWL. The change rate of LUI information for most popular sites is shown in section 4.3 and section 4.4. In the experiment described in section 4.5, we evaluate the number of new LUIs that an individual user encounters everyday.

4.1 Study of Naïve Bayesian Classifier

The Naïve Bayesian classifier was used to train AIWL to identify a successful login process. We simulated login processes for 34 web sites. 18 of 34 are phishing web sites selected from PhishTank.com [12] on May 13th, 2008. The other 16 are legitimate web sites. For every legitimate web site, both the successful login process and the failing one were simulated. We simulated failing login process by purposely using wrong passwords. So there are altogether 50 login processes acting as the training samples.

A data-collecting tool was designed to work as a plug-in of Internet Explorer. When a login process was run, this tool collected the features listed in section 3.2 to represent current login process. Then the result of the login process would be specified by people. The features and results were recorded in an XML file.

A sample of the XML file is shown in Figure 1 where the tag "Account" represents each login process, the tag "HtmlURL" specifies URL of the web site, the tag "SuccessOrNot" indicates whether the login process was a successful one, the tag "OperDate" represents the date of the experiment day and other tags are the same as the feature described in section 3.2

After all the login processes had been run, the XML file was analyzed and the probabilities listed in Section 3.4 were calculated. Table 1 shows the result of the exact percentages of that login processes have each of the five features. Here the threshold of Numberoflink is set to be 35 and the threshold of Opertime is set to be 50000. These two thresholds were determined after repeated experiments for the best performance for classification.

Figure 1. The XML file that records the features and results of a login process

As can be seen in Table 1, all of the features are matched more frequently by successful login processes than by failing ones. The percentage of inbrowserhistory in both cases is high. That is because in our experiment, the failing login processes were not only simulated in phishing sites but also in legitimate sites.

The data in Table 1 can be used to make AIWL classify a new unseen login process. The efficiency of this classification is examined in the next section.

Table 1. Percentage of login processes matching the features

Feature	Successful login process Matched	Failing login process Matched
inbrowserhistory	78.95%	61.11%
HasNopasswordField	94.74%	38.89%
Numberoflink>35	42.11%	11.11%
HasNoUsername	57.89%	36.11%
Opertime>50000	84.21%	25.00%

4.2 Efficiency in Classifying Login Process

In this section, we evaluate the efficiency of the classifier in classifying login processes. We got the efficiency of the classifier by simulating login processes in various web sites and examining whether the classifier can classify these login processes correctly.

Those web sites include 10 phishing web sites and 5 legitimate web sites. The 10 phishing URLs were selected from PhishTank.com [12] on May 13th, 2008. The legitimate web sites were picked up from Email, blog and other commonly used information systems. For every legitimate web site, we simulated both the successful login process and the failing one. We simulated failing login process by purposely using wrong passwords.

We use true positive and false positive to evaluate the efficiency of the Naïve Bayesian classifier. True positive indicates the possibility of correctly identifying a successful login process as a successful one while false positive indicates the possibility of incorrectly labeling a failing login process as a successful 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.

Table 2. The result of classification by AIWL

URL	Login process Result	Probability of Successful login
163.com	*Fail*	*3%*
126.com	*Fail*	*7%*
Blogbus.com	Success	85%
Shineblog.com	Success	85%
Yahoo.com	*Fail*	*1%*
Google.com	*Fail*	*7%*
Crsky.com	*Fail*	*13%*
Whsee.com	Success	85%
Bloglines.com	Success	71%
Fc2.com	Success	93%
Phishing Site 1	*Fail*	*1%*
Phishing Site 2	*Fail*	*13%*
Phishing Site 3	*Fail*	*13%*
Phishing Site 4	*Fail*	*1%*
Phishing Site 5	*Fail*	*3%*
Phishing Site 6	*Fail*	*13%*
Phishing Site 7	*Fail*	*3%*
Phishing Site 8	*Fail*	*13%*
Phishing Site 9	*Fail*	*1%*
Phishing Site 10	*Fail*	*13%*

An automated tool working as a plug-in of Internet Explorer was designed to collect the features listed in section 3.2. Then the result of section 4.1 was used to calculate the probability of a login process to be a successful one.

Table 2 shows the work result of the classifier. Login Process Result indicates the actual result of the login process and

Probability of Successful login means the probability calculated by the classifier about whether the login process is successful.

We set the threshold of login process classification to be 70%. It means if the probability of successful login is more than 70%, we believe this login process is a successful one.

Based on the threshold defined above, the result of true positive and false positive of classifier for classifying login process is shown in Table 3. The true positive is 100% and false positive is 0%. It means that AIWL can recognize all the successful login process as successful ones and all the failing login process as failing ones. So we can conclude that the classifying result of the classifier is basically reliable.

Table 3. True positive and false positive of the classifier for classifying login process

	True Positive	False Positive
the Naïve Bayesian classifier	100%	0%

4.3 Efficiency of the White-List

As is introduced above, AIWL uses a white-list to detect phishing site. But if a legitimate web site frequently modifies its LUI which is stored in the white-list or users often login in a web site whose LUI is not stored in the white-list, AIWL will obviously often give a wrong warning in user's login process.

Three factors which are serious for wrong warning of the white-list will be introduced in the rest of this section. The below analysis shows the wrong warning is very low. That is, the true-positive is very high (near 100%), and the false-positive is 0 (because of the white-list).

4.3.1 Change Rate of IP address

We conducted the experiment to observe the change rate of IP addresses for 15 most commonly used login sites [13]. The 15 most 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.

An IP collection tool was designed to collect the IP addresses of those web sites and log the data in an XML file every one hour. In this XML file, the following information was recorded: URL of the web site, the corresponding IP addresses and the log time. For some web sites, there may be more than one IP address corresponding to a single URL. All of them were obtained and recorded.

The experiment began on 4/8/2008 and ended on 5/18/2008. There are altogether 984 records for every web site.

After that, another tool was designed to analyze the XML file to get the change rate of IP addresses for each web site. We found that 14 of the 15 web sites had not changed the IP addresses in this period of time. Only one web site, hi5.com, altered its IP addresses on 8:33:41, April 16th. 4 new IP addresses have replaced the 4 old ones, which never appeared in the rest of the day. From April 16th to the end of this experiment, all the 15 web sites kept unchanged. Table 4 shows the change rate of the 15 popular login sites.

With this experiment, we can conclude that the IP addresses corresponding to a certain URL only change occasionally. So IP addresses in LUI can be used to check the validity of a web site with low possibility to cause wrong warning problem.

Table 4. The change rate of the 15 popular login sites

Site	Change Rate (times/30days)	Change Time
aol.com	0	N/A
bebo.com	0	N/A
ebay.co.uk	0	N/A
ebay.com	0	N/A
google.com	0	N/A
hi5.com	1	4/16/2008 8:33:41
live.com	0	N/A
match.com	0	N/A
msn.com	0	N/A
myspace.com	0	N/A
passport.net	0	N/A
paypal.com	0	N/A
Yahoo.co.jp	0	N/A
Yahoo.com	0	N/A
Youtube.com	0	N/A

4.3.2 Change Rate of InputArea and ValueHash

We conducted the experiment to observe the change rate of InputArea and ValueHash for 11 most popular e-bank web sites in China and 15 most commonly used login sites described in section 4.3. The 11 most popular e-bank web sites are: spdb.com.cn, cmbchina.com, gdb.com.cn, 95559.com.cn, icbc.com.cn, 95599.cn, ccb.com.cn, bank-of-china.com, ecitic.com.

An automated tool was used to watch and record the change rate of certain web sites.

The experiment of banks began on 4/8/2008 and ended on 5/18/2008. The 11 web sites were checked every day. There are 41 records in all for each web site. The experiment of 15 popular sites began on 6/11/2008 and ended on 8/5/2008. The 15 web sites were checked every day. There are 56 records in all for each web site.

After analyzing the result, we find that none of the web sites had changed the InputArea and ValueHash in the login page in this period of time, which means InputArea and ValueHash in LUI can be used to check the validity of a web site with low possibility to cause wrong warning problem.

4.3.3 Number of new LUIs of user per day

We conducted this experiment to get the number of new LUIs of users per day. 8 students have participated in this experiment. 5 of them are graduate students and the others are undergraduate students. The experiment began on 2/27/2008 and ended on 3/9/2008.

We designed a data-collecting tool working as a plug-in of Internet Explorer to keep track of users' login sessions. This tool worked in back-end. Every time a user logged in a web site, our tool recorded the current date and the URL of the web site in an XML file. The participators were asked to send the record back on March 9th, 2008.

After all the XML files were sent back, we used another automated tool to review the XML file of each student, and only kept the new LUI record for each day. For example, Shirley (one of the participators) had a login record for http://mail.yahoo.com.cn in 2008/ 3/ 4 13:40: 9, then another record of login for http://mail.yahoo.com.cn in 2008/ 3/ 7 11: 9: 0 would be omitted. Because in our view, it's not the first time for Shirley to log in http://mail.yahoo.com.cn, which means this LUI is not new for Shirley in the second case.

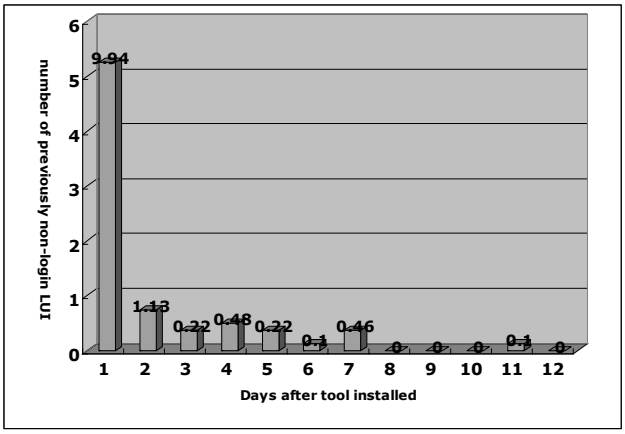

Figure 2. Number of previously non-LUI for users per day

We calculated the number of new LUIs that each participator logged in every day, and then the records of 8 students was computed. Figure 2 shows the number of new LUI of users per day. The number at the top of the column is the variance of the dataset for each day.

As is shown in Figure 2, the number of new LUIs for the user decreased relatively quickly. In the last few days, it almost decreased to zero. It means a user has only a limited number of frequently logged in web sites.

5. DISCUSSION

5.1 True Positives and False Positives

The Naïve Bayesian classifier in AIWL has a 100% true positive and a 0% false positive rate for identifying successful login process. This perfect result is based on the behavior of current login web sites, because the features used to represent a login process were chosen delicately to cope with current web sites' behavior. Once the behavior of login web sites changes in some way, AIWL may not perform so perfectly. We will keep track of the behavior of login web site and adjust the AIWL correspondingly in our future work to keep the high efficiency of the classifier.

The efficiency of the white-list is also very good. Because the content of white list is stable, the almost all legitimate sites will not be alert (high true-positive), and all phishing sites will .theoretically be alert (false-positive is 0, because AIWL uses a white-list).

5.2 Comparison with Other Solutions

This section compares our AIWL with other anti-phishing solutions. These compared solutions include Web Wallet [15], SpoofGuard [16], PwdHash [23], Dynamic Security Skins [15], and the Microsoft Phishing Filter in IE7 [31]. Web Wallet is a browser side bar to detect phishing attacks; SpoofGuard [16] is a browser plug-in that places a traffic light in the browser toolbar to indicate the security level of a web site; PwdHash is a browser extension that transparently converts a user's password into a domain-specific password; Dynamic Security Skins allows a remote web server to prove its identity in a way that is easy for a human user to verify and hard for an attacker to spoof; the Microsoft Phishing Filter in IE7 relies on the blacklist hosted by Microsoft for phishing detection and shows a warning message when encountering the suspected phishing sites.

The detailed description of every comparison term is as follows: LUI Authentication means that the solution can authenticate LUI information; Anti-Pharming means that the solution can defend against pharming.

Table 5. Comparison of solutions of anti-phishing tools

Anti-phishing Tools	LUI Authentication	Anti-Pharming
AIWL	Yes	Yes
Web Wallet	Weak	No
SpoofGuard	Weak	No
PwdHash	No	No
Microsoft Phishing Filter in IE7	No	No

Based on the result of comparison in Table 5, AIWL has more advantages over other solutions. Especially, LUI Authentication and Anti-Pharming are absent or weak in other solutions. So, AIWL provides a more tight security environment for user's confidential information.

5.3 Limitation of AIWL

This section analyzes the limitation of AIWL and possible solutions.

It is obvious that the white-list itself is the key point in this approach. If the white-list has been controlled, the whole application will lose its efficacy.

For the controlling problem, because the whole AIWL is installed in local PC, It's difficult for AIWL to defend against local-machine Trojan Horse and viruses. It could be a common problem for the secure tools installed and running in local PC. One possible solution is to store the white-list in a more secure device, e.g. a smart phone [33]. When accessing the white-list, the client installed on the PC communicates with the corresponding smart phone by blue-tooth. The white-list can be protected more securely in this way.

For the losing problem of data, a backup and restoration approach could solve this problem. The white-list can be backed up in the local machine, in a web server or in the mobile phone.

6. RELATED WORK

Phishing attack now becomes a significant threat to user's authentication data. According to the study of Zhang et al. [2], there are four categories in past work of anti-phishing: studies of why people fall for phishing attacks; methods of educating people about phishing attacks; the development of better user interfaces for anti-phishing tools; and automated tools to detect phishing. Our approach AIWL proposed in this paper contributes a novel approach to the development of phishing detection tools.

Among the large number of anti-phishing tools, most of the tools defend against phishing attacks in either of the following two ways: preventing the confidential information from leaking to the attackers or making the leaked information useless. A lot of toolbars [17-23] built into web browsers use the first way to detect phishing sites. The heuristics or blacklist approach is used to validate the web site user visits. Those tools either alert the user to the possible danger or use the symbol to mark the security level of the web sites. In [30], Zhang et al. have developed a semi-automated test bed to evaluate 10 popular anti-phishing toolbars. They found that the anti-phishing tools are not as efficient as expected in preventing users from being spoofed by high-quality phishing attacks.

On the other hand, there are some tools that focus on making the leaked information useless to defend against phishing. PwdHash [23] and PassPet [24] are two of them. Both of the tools generate different passwords for various sites, so that theft of the password at one site will not yield a password that is useful at another site. In [26], Florencio and Herley introduce another novel way to stop the phishing attacks. The client reports user's account information to the server whenever the confidential information is leaked to the attackers. The server aggregates those reports to produce a black list and deny the service to these may-be-attacked accounts. One big problem of the solution is that it needs to cooperate with the legitimate organizations.

There are some other works on how to strengthen authentication process. Single Sign-On (SSO) is one of the hottest topics. Single sign-on (SSO) [27] is the mechanism whereby a single action of user authentication permits the user to access all computers and systems to which he/she has access permission, without the need to enter multiple passwords. In [28], Cosign, an open source web single sign-on package is introduced. Cosign uses the cookies to control the user's authority to certain resource without additional password prompting. But fundamentally, Cosign is dependent on user's acceptance of cookies to work. The problem of SSO is that the server side must be modified to support such authentication mechanisms.

The Naïve Bayesian classifier is thought to be one of the most effective approaches for learning to classify text documents. In [6], Sahami et al. found that by using Naïve Bayesian approach, it is possible to automatically learn effective filters to eliminate a large portion of spam. In [29], Deshpande et al. examined the effectiveness of statistically based Naïve Bayesian anti-spam filtering. They also evaluate different threshold values in order to find the optimal configuration.

AIWL automatically maintains the individual white-list containing user's familiar LUI information of web sites by using Naïve Bayesian classifier. This white-list will reach a stable state after a period of study because of the limit size of user's frequently login web sites. So AIWL can detect phishing sites effectively. AIWL also requires no modification on the server side.

7. CONCLUSION AND FUTURE WORK

This paper proposes a novel approach, named Automated Individual White-List (AIWL), for anti-phishing. Our approach, AIWL is effective in detecting phishing and pharming attacks with low false positive. AIWL stores entire LUI information rather than only a URL of a web site in the white-list to provide a more secure environment, especially it can efficiently defend the pharming.

Moreover, The Naive Bayesian classifier is used to automatically maintain the white-list for the user. As our experiment shows, AIWL identities a successful login process efficiently; so it can maintain the individual white-list smartly.

On the basis of our experiments and evaluations, the change rate of IP addresses, InputArea and ValueHash of popular web site is zero or near zero. Moreover, the life of a web site's certificate has almost lasted for years. Based on all the above, we can conclude that the change rate of LUI information is low, so using LUI information to check the validity of a web site will cause few wrong warning problems.

In addition, the familiar web site of a user is usually limited. So, as time passes by, the white-list in AIWL will be more and more stable and fit into the user. The warnings to the user will be more and more accurate.

In future, we will use a more private device (smart phone) to store white-list in a more secure environment. More experiments with larger datasets will also be preformed to make AIWL more efficient. The change rate of IP should be a big problem in AIWL, longer time-span need to be used to gather the web sites' IP and analyze.

ACKNOWLEDGEMENT

Thank Ms. Min Li proof reading and revising the wording of our paper. And thank the anonymous reviewers give us suggests and comments.

REFERENCES

[1] Identity Theft: What to Do if It Happens to You. http://www.anti-phishing.org/consumer_recs2.htm.

[2] Y. Zhang, J. Hong and L. Cranor. CANTINA: A Content-Based Approach to Detecting Phishing Web Sites. Proceeding of International World Wide Web Conference (WWW 2007), Banff, Alberta, Canada, May 2007: 639-648.

[3] Anti-Phishing Working Group. http://www.anti-phishing.org/.

[4] M. Jakobsson and S. Myers. Delayed Password Disclosure. Proceedings of the 2007 ACM workshop on Digital identity management, Nov. 2007: 17-26

[5] I. Fette, N. Sadeh, A. Tomasic. Learning to Detect Phishing Emails. Proceeding of International World Wide Web Conference (WWW 2007), Banff, Alberta, Canada, May 2007: 649-656.

[6] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. A Bayesian approach to filtering junk email. AAAI Workshop on Learning for Text Categorization, Madison, Wisconsin, July 1998,. AAAI Technical Report WS-98-05

[7] Ion Androutsopoulos, John Koutsias, Konstantinos V. Cbandrinos and Constantine D. Spyropoulos. An Experimental Comparison of Naive Bayesian and Keyword-Based Anti-Spam Filtering with Personal E-mail Messages. Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR 2000), Athens Greece, 2000: 160-167.

[8] R. O. Duda and P.E. Hart. Bayes Decision Theory. Chapter 2 in Pattern Classification and Scene Analysis, pp. 10--43. John Wiley, 1973.

[9] T. M. Mitchell. Bayesian Learning. Chapter 6 in Machine Learning, pp. 154-200. McGraw-Hill, 1997.

[10] P. Domingos and M. Pazzani. Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier. Proceedings of the 13th International Conference on Machine Learning, Bari, Italy, 1996: 105-112

[11] P. Langley, I. Wayne and K. Thompson. An Analysis of Bayesian Classifiers. Proceedings of the 10th National Conference on Artificial Intelligence, San Jose, Califomia, 1992: 223-228

[12] PhishTank. http://www.phishtank.com/.

[13] D. Florencio and C. Herley. A Large-Scale Study of Web Password Habits. Proceeding of International World Wide Web Conference (WWW 2007), Banff, Alberta, Canada, May, 2007: 657-665.

[14] RSA Security, Protecting Against Phishing by Implementing Strong Two-Factor Authentication. 2004,https://www.rsasecurity.com/products/securid/whitepapers/PHISH_WP_0904.pdf.

[15] R. Dhamija and J. D. Tygar. The battle against phishing: Dynamic Security Skins. Proceedings of the 2005 symposium on Usable privacy and security, Pittsburgh, Pennsylvania,2005: 77-88

[16] M. Wu, R. C. Miller, G. Little. Web Wallet: Preventing Phishing Attacks by Revealing User Intentions, Symposium on Usable Privacy and Security (SOUPS) 2006, July 12-14, 2006, Pittsburgh, PA, USA.

[17] SpoofGuard. http://crypto.stanford.edu/SpoofGuard/.

[18] NetCraft, Netcraft Anti-Phishng Toolbar. http://toolbar.netcraft.com/.

[19] Google Safe Browsing for Firefox. http://www.google.com/tools/firefox/safebrowsing

[20] EarthLink Tool. http://www.earthlink.net/software/free/toolbar/.

[21] GeoTrust, Inc. TrustWatch Tool. http://toolbar.trustwatch.com/tour/v3ie/toolbar-v3ie-tour-overview.html.

[22] CallingID, Ltd. http://www.callingid.com/DesktopSolutions/CallingIDToolbar.aspx.

[23] eBay Toolbar's Account Guard. http://pages.ebay.com/help/confidence/account-guard.html.

[24] B. Ross, C. Jackson, N. Miyake et al. Mitchell. Stronger Password Authentication Using Browser Extensions, Proceedings of the 14th Usenix Security Symposium, 2005.

[25] K. P. Yee and Kragen Sitaker. Passpet: Convenient Password Management and Phishing Protection. Symposium On Usable Privacy and Security (SOUPS) 2006, July 12–14, 2006, Pittsburgh, PA, USA.

[26] D. Florencio and C. Herley. Stopping a Phishing Attack, Even when the Victims Ignore Warnings, Microsoft Research (MSR), Tech. Rep. MSR-TR-2005-142, 2005.

[27] Single Sign-On. http://www.opengroup.org/security/sso/.

[28] Cosign.http://www.umich.edu/~umweb/software/cosign/overview.html

[29] V. P. Deshpande, R. F. Erbacher, C. Harris. An Evaluation of Naýve Bayesian Anti-Spam Filtering Techniques. Information Assurance and Security Workshop, 2007. IAW '07. IEEE SMC 20-22 June 2007: 333 – 340

[30] Y. Zhang, S. Egelman, L. Cranor and J. Hong. Phinding Phish: Evaluating Anti-Phishing Tools. In Proceedings of the 14th Annual Network & Distributed System Security Symposium (NDSS 2007), San Diego, CA, 2007

[31] Microsoft Corporation. Internet Explorer 7. http://www.microsoft.com/windows/ie/default.mspx

[32] C. Karlof, J.D. Tygar, D. Wagner, et al. Dynamic Pharming Attacks and Locked Same-origin Policies for Web Browsers. ACM CCS 2007. November 2007 : 58 - 71.

[33] W. Han, Y. Wang, Y. Cao, et al. Anti-Phishing by Smart Mobile Device, IFIP International Conference on Network and Parallel Computing, 2007, Dalian, China: 295-300.