MAT JOURNALS
Journal of Computer Science Engineering and Software Testing
Volume 2 Issue 2

Phishing Detection and Prevention: A New Approach

Rahmathulla K.
M E A Engineering College, Kerala, India
E-mail: rahmathk123@gmail.com

Abstract

Phishing is that the act of making an attempt to accumulate data admire usernames, passwords, and master card details (and generally, indirectly, money) by masquerading as a trustworthy entity in associate degree transmission. Communications purporting to be from widespread social websites, auction sites, banks, on-line payment processors or IT directors square measure unremarkably would not to lure unsuspecting public. Phishing emails might contain links to websites that square measure infected with malware. Phishing is usually allotted by email spoofing or instant electronic communication, and it usually directs users to enter details at a pretend web site whose look and feel square measure virtually similar to the legitimate one. During this paper, we tend to propose a replacement end-host based mostly anti-phishing algorithmic program that we tend to decision LinkGuard, by utilizing the generic characteristics of the hyperlinks in phishing attacks. These characteristics square measure derived by analyzing the phishing information archive provided by the Anti-Phishing Working Group (APWG). Because it is supported the generic characteristics of phishing attacks, LinkGuard will find not solely familiar, however, additionally unknown phishing attacks. We have got enforced LinkGuard in Windows XP. Our experiments verified that LinkGuard is effective to find and forestall each familiar and unknown phishing attacks with borderline false negatives.

Keywords: *Network security, phishing attacks, hyperlink, linkguard algorithm*

INTRODUCTION

The word `Phishing' at first emerged in Nineties. The first hackers typically use `ph' to switch `f' to supply new words within the hacker's community, since they sometimes hack by phones. Phishing could be a new word created from `fishing', it refers to the act that the

assailant attract users to go to a faked data processor by causation them faked e-mails (or instant messages), and stealthily get victim's personal info equivalent to user name, password, and national security ID, etc. This info then will be used for future target advertisements or maybe fraud attacks (e.g., transfer cash from victims' bank account).

The often used attack technique is to send e-mails to potential victims that looked as if it would be sent by banks, on-line organizations, or ISPs. In these e-mails, they are going to frame some causes, e.g. the secret of your master card had been mis-entered for several times, or they are providing upgrading services, to attract you visit their information processing system to evolve or modify your account variety and secret through the link provided within the e-mail. You may then be connected to a counterfeited information processing system when clicking those links. The style, the functions performed, typically even the address of those faked websites are just like the important information processing system. It is very troublesome for you to grasp that you just are literally visiting a malicious web site. If you input the account variety and secret, the attackers then with success collect the knowledge at the server aspect, and is in a position to perform their next step actions thereupon data (e.g., withdraw cash out from your account).

Phishing itself is not a brand new construct, however, it is more and more utilized by phishers to steal user info and perform business crime in recent years. At intervals one to 2 years, the amount of phishing attacks inflated dramatically. in line with Gartner opposition., for the twelve months ending Gregorian calendar month 2004, "there were 1.8 million phishing attack victims, and, therefore, the fraud incurred by phishing victims destroyed $1.2 billion" [1–6]. According to the statistics provided by the Anti-Phishing Working Group (APWG), in March 2006, the total number of unique phishing reports submitted to the APWG was 18,480; and the top three phishing site hosting countries are, the United States (35.13%), China (11.93%), and the Republic of Korea (8.85%) [2]. The infamous phishing attacks happened in China in recent years include the events to counterfeit the Bank of China (real Web site www.bank-of-china.com, counterfeited Web site www.bank-off-china.com), the Industrial and Commercial Bank of China (real Web site www.icbc.com.cn, faked web site www.1cbc.com.cn), the Agricultural Bank

of China (real website www.95599.com, faked Web site www.965555.com), etc.

In this paper, we study the common procedure of phishing attacks and review possible anti-phishing approaches. We then focus on end-host based anti-phishing approach. We first analyze the common characteristics of the hyperlinks in phishing e-mails. Our analysis identifies that the phishing hyperlinks share one or more characteristics as listed below:

- The visual link and, therefore, the actual link are not identical.
- The attackers usually use dotted decimal scientific discipline address rather than DNS name.
- Special tricks area unit accustomed write the hyperlinks maliciously.
- The attackers typically use faux DNS names that area unit similar (but not identical) with the target electronic computer.

We then propose associate degree end-host primarily based anti-phishing algorithmic program that we have a tendency to decision LinkGuard, supported the characteristics of the phishing link. Since LinkGuard is character-based, it will notice and forestall not solely proverbial phishing attacks, however, conjointly unknown ones.

PHISHING ATTACK PROCEDURE AND PREVENTION METHODS

In this paper, we assume that phishers use e-mail as their major method to carry out phishing attacks. Nonetheless, our analysis and algorithm can be applied to attacks that use other means such as instant messaging.

The Procedure of Phishing Attacks

In general, phishing attacks are performed with the following four steps:

- Phishers found out a counterfeited information processing system that appearance specifically just like the legitimate information processing system, together with fitting the internet server, applying the DNS server name, and making the online pages the same as the destination internet site, etc.
- Send great deal of spoofed e-mails to focus on users within the name of these legitimate corporations and organizations, attempting to persuade the potential victims to go to their websites.
- Receivers receive the e-mail, open it, click the spoofed hyperlink in the e-mail, and input the required

information.

- Phishers steal the personal data and perform their fraud akin to transferring cash from the victims' account.

Approaches to Prevent Phishing Attacks

Enhance the Security of the Web Sites

The business websites resembling the online sites of banks will take new strategies to ensure the security of users' personal info. One technique to boost the safety is to use hardware devices. Parenthetically, the Barclays bank provides a hand-held card reader to the users. Before searching within the internet, users ought to insert their mastercard into the cardboard reader, and input their (personal identification number) PIN code, then the cardboard reader can turn out a once security parole, users will perform transactions solely when the proper parole is input [7–12]. Another technique is to use the biometry characteristic (e.g., voice, fingerprint, iris, etc.) for user authentication. Parenthetically, Paypal had tried to interchange the only parole verification by voice recognition to boost the safety of the online website. With these strategies, the phishers cannot accomplish their tasks even when they have gotten half of the victims' info. However, all these techniques want extra hardware to understand the authentication between the users and also the internet sites, thence can increase the value and produce sure inconvenience. Therefore, it still wants time for these techniques to be wide adopted.

Block the Phishing E-Mails by Various Spam Filters

Phishers typically use e-mails as `bait' to attract potential victims. SMTP (Simple Mail Transfer Protocol) is that the protocol to deliver e-mails within the web [11]. It is a really straightforward protocol that lacks necessary authentication mechanisms. Data regarding sender, adore the name and email.

There are several (technical or non-technical) ways to prevent phishing attacks:

1) Educate users to understand how phishing attacks work and be alert when phishing-alike e- mails are received.

2) Use legal methods to punish phishing attackers. 3) Use technical methods to stop phishing attackers.

In this paper, we only focus on the third one. Technically, if we can cut off one or several of the steps that needed by a phishing attack, we then successfully prevent that attack. In what follows, we

briefly review these approaches.

Detect and Block the Phishing Web Sites in Time

If we will cite the phishing websites in time, we tend to then will block the sites and forestall phishing attacks. It is comparatively simple to (manually) confirm whether or not a web site could be a phishing site or not, however, it is troublesome to seek out that phishing sites call at time. The Web address of the sender, route of the message, etc., will be counterfeited in SMTP. Thus, the attackers will channel massive amounts of spoofed e-mails that area unit appeared from legitimate organizations. The phishers hide their identities once causation the spoofed e-mails, therefore, if anti-spam systems will confirm whether or not an e-mail is distributed by the declared sender (Am I Whom I Say I Am?), the phishing attacks are going to be decreased dramatically. From this time, the techniques that preventing senders from counterfeiting their Send ID (e.g., SIDF of Microsoft) will defeat phishing attacks with efficiency [8].

SIDF could be a combination of Microsoft's display for E-mail and, therefore, the SPF (Sender Policy Framework) developed by Meng Weng Wong [13]. Each display and SPF check e-mail sender's name to verify if the e-mail is distributed from a server that is licensed to send e-mails of that domain, and from that to see whether or not that e-mail use spoofed e-mail address. If it is faked, the web service supplier will then verify that e-mail could be a spam e-mail

The spoofed e-mails employed by phishers are one kind of spam e-mails. From this time of read, the spam filters may also be wont to filter those phishing e-mails [1, 4]. For instance, blacklist, whitelist, keyword filters, Bayesian filters wit h self-learning talents, and E-Mail Stamp, etc., will all be used at the e-mail server or consumer systems. Most of those anti-spam techniques perform filtering at the receiving facet by scanni nanogram the contents and also the address of the received e-mails. And they all have pros and cons as discussed below. Blacklist and whitelist cannot work if the names of the spamers are not known in advance. Keyword filter and Bayesian filters can detect spam based on content, hence can detect unknown spasm. But they can also result in false positives and false negatives. Furthermore, spam filters are designed for general spam e-mails and may not very suitable for filtering phishing e-mails since they generally do

not consider the specific characteristics of phishing attacks.

Install Online Anti-Phishing Software in User's Computers

Despite all the above efforts, it is still possible for the users to visit the spoofed Web sites. As a last defense, users can install anti-phishing tools in their computers. The anti-phishing tools in use today can be divided into two categories: blacklist/whitelist based and rule-based.

Category I

When a user visits a Web site, the anti-phishing tool searches the address of that site in a blacklist stored in the database. If the visited site is on the list, the anti-phishing tool then warns the users. Tools in this category include ScamBlocker from the EarthLink company, PhishGuard, and Netcraft, etc. [5, 9, 10]. Though the developers of these tools all announced that they can update the blacklist in time, they cannot prevent the attacks from the newly emerged (unknown) phishing sites.

Category II

This class of tools uses bound rules in their package, and checks the protection of an online web site in step with these rules. Samples of this sort of tools embrace SpoofGuard developed by Stanford, TrustWatch of the GeoTrust etc. [3, 7]. SpoofGuard checks the name, address (includes the port number) of an online web site, it additionally checks whether or not the browser is directed to this address via the links within the contents of e-mails. If it finds that the name of the visited computing machine is comparable to a widely known name, or if they are not victimisation the quality port, SpoofGuard can warn the users. In TrustWatch, the safety of an internet website is decided by whether or not it is been reviewed by associate in nursing freelance sure third party organization. Each SpoofGuard and TrustWatch give a toolbar within the browsers to apprise their users whether or not the net website is verified and sure. It is straightforward to watch that everyone the higher than defense ways are helpful and complementary to every different, however, none of them are excellent at the present stage. Within the remainder of the paper, we tend to specialize in end-host primarily based approach associate in nursing propose an end-host based LinkGuard algorithmic program for phishing detection and interference. To this end, our work follows the same approach as [3]. Our work differs from in that: 1) LinkGuard is based on our careful analysis of the characteristics of phishing hyperlinks whereas SpoofGuard is more

like a framework.

2) LinkGuard has a verified very low false negative rate for unknown phishing attacks whereas the false negative property of SpoofGuard is still not known.

LINKGUARD

Classification of the Hyperlinks in the Phishing E-Mails

In order to (illegally) collect useful information from po-tential victims, phishers generally tries to convince the users to click the hyperlink embedded in the phishing e-mail. A hyperlink has a structure as follows.

 Anchor text <\a>

where `URI' (universal resource identifiers) provides the n ec-essary information needed for the user to access the net-worked resource and `Anchor text' is the text that will be displayed in user's Web browser. Examples of URIs are http://www.google.com, https://www.icbc.com.cn/login.html, ftp://61.112.1.90:2345, etc. `Anchor text' in general is used to display information related to the URI to help the user to better understand the resources provided by the hyperlink. In the following hyperlink, the URI links to the phishing archives provided by the APWG group, and its anchor text “Phishing Archive” informs the user what is the hyperlink is about.

<a href“http://www.antiphishing.org/phishing archive.html”>
Phishing Archive

Note that the content of the URI will not be displayed in user's Web browser. Phishers, therefore, can utilize this fact to play trick in their `bait' e-mails. In the rest of the paper, we call the URI in the hyperlink the actual link and the anchor text the visual link. After analyzing the 203 (there are altogether 210 phishing e-mails, with 7 of them with incomplete information or with malware attachment and do not have hyperlinks) phishing e-mail archives from Sep. 21st 2003 to July 4th 2005 provided by APWG [6]. We classified the hyperlinks used in the phishing e-mail into the following categories:

1) The hyperlink provides DNS domain names in the anchor text, but the destination DNS name in the visible link does not match that in the actual link. For instance, the following hyperlink:

https://secure.regionset.com/EBanking/logon/appears to be linked to secure.regionset.com, which is the portal

of a bank, but it actually is linked to a phishing site www.profusenet.net.

2) Dotted decimal IP address is used directly in the URI or the anchor text instead of DNS name. See below for an example.

 SIGN IN

3) The hyperlink is counterfeited maliciously by using certain encoding schemes. There are two cases:

a) The link is formed by encoding alphabets into their corresponding ASCII codes. See below for such a hyperlink.

<a href="http://%34%2E%33%34%2E%31%39%35%2E

Table 1: *The Categories of Hyperlinks in Phishing E-Mails.*

Category	Number of Links	Percentage
1	90	44.33%
2	85	41.87%
3.a	19	9.36%
3.b	16	7.88%
4	67	33%
5	4	2%

%34%31:%34%39%30%33/%6C/%69%6E%64%65%78 %2E%68%74%6D" > www.citibank.com

while this link is seemed pointed www.citibank.com, it actually points to http://4.34.195.41:34/l/index.htm.

b) Special characters (e.g. @ in the visible link) are used to fool the user to believe that the e-mail is from a trusted sender. For instance, the following link seems is linked to amazon, but it actually is linked to IP address 69.10.142.34. http://www.amazon.com:fvthsgbljhfcs83infoupdate @69.10.142.34.

4) The link does not offer destination info in its anchor text and uses DNS names in its URI. The DNS name within the URI typically is comparable with a famed company or organization. Let us say, the subsequent link looks to be sent from paypal, however, it truly is not. Since paypal-cgi is actually registered by the phisher to let the users believe that it has something to do with paypal

 Click here to confirm your account

5) The attackers utilize the vulnerabilities of the target Web site to redirect users to their phishing sites or to launch CSS (cross site scripting) attacks. For example, the following link

<a href="http://usa.visa.com/track/dyredir.jsp?rDirl=http://200.251.251.10/.verified/" > Click here <a> Once clicked, will redirect the user to the phishing site 200.251.251.10 due to a vulnerability of usa.visa.com.

Table 1 summarizes the number of hyperlinks and their percentages for all the categories. It is ascertained that almost all of the phishing e-mails use faked DNS names (category 1, 44.33%) or dotted decimal science addresses (category 2, 41.87%). cryptography tricks are of t times used (category 3a and 3b, 17.24%). And phishing attackers usually attempt to fool users by putting in DNS names that area unit terribly similar with the $64000 e-commence sites or by not providing destination data within the anchor text (category 4). Phishing attacks that utilize the vulnerability of websites (category 5) area unit of little variety (2%) and that we leave this sort of attacks for future study. Note that a phishing hyperlink can belong to several categories at the same time. For instance, an attacker may use tricks from both categories 1 and 3 at the same time to increase his success chance. Hence, the sum of percentages is larger than 1. Once the characteristics of the phishing hyperlinks are understood, we are able to

design anti-phishing algorithms that can detect known or unknown phishing attacks in real-time. We present our LinkGuard algorithm in the next subsection.

The LinkGuard Algorithm

LinkGuard works by analyzing the differences between the visual link and the actual link. It also calculates the similarities of a URI with a known trusted site. The following terminologies are used in the algorithm.

v_link: visual link; a_link: actual_link; v_dns: visual DNS name; a_dns: actual DNS name;

sender_dns: sender's DNS name.

```
int LinkGuard(v_link, a_link} {
1.  v_dns = GetDNSName(v_link);
2.  a_dns = GetDNSName(a_link);
3.  if ((v_dns and a_dns are not
4.  empty) and (v_dns != a_dns))
5.  return PHISHING;
6.  if (a_dns is dotted decimal)
7.  return POSSIBLE_PHISHING;
8.  if(a_link or v_link is encoded)
9.  {
10. v_link2 = decode (v_link);
11. a_link2 = decode (a_link);
12. return LinkGuard(v_link2, a_link2);
13. }
14. /* analyze the domain name for
15. possible phishing */
16. if(v_dns is NULL)
17. return AnalyzeDNS(a_link);
18. }
```

The LinkGuard algorithm works as follows. In its main routine LinkGuard, it first extracts the DNS names from the actual and the visual links (lines 1 and 2). It then compares the actual and visual DNS names, if these names are not the same, then it is phishing of category 1 (lines 3-5). If dotted decimal IP address is directly used in actual dns, it is then a possible phishing attack of category 2 (lines 6 and 7). We will delay the discussion of how to handle possible phishing attacks later. If the actual link or the visual link is encoded

```
int AnalyzeDNS (actual_link) {
/* Analyze the actual DNS name according to
the blacklist and whitelist*/
1.  if (actual_dns in blacklist)
2.  return PHISHING;
3.  if (actual_dns in whitelist)
4.  return NOTPHISHING;
5.  return PatternMatching(actual_link);
6.  }
7.  int PatternMatching(actual_link){
8.  if (sender_dns and actual_dns are
    different)
9.  return POSSIBLE_PHISHING;
10. for (each item prev_dns in seed_set)
```

```
11. {
12. bv = Similarity(prev_dns, actual_link);
13. if (bv == true)
14. return POSSIBLE_PHISHING;
15. }
16. return NO_PHISHING;
17. }

18. float Similarity (str, actual_link) { 32
    if (str is part of actual_link)
19. return true;
20. int maxlen = the maximum string
21. lengths of str and actual_dns;
22. int minchange = the minimum number
    of
23. changes needed to transform str
24. to actual_dns (or vice verse);
25. if (thresh<(maxlen-
    minchange)/maxlen<1)
26. return true
27. return false;
28. }
```

(Categories 3 and 4), we first decode the links, then recursiv ely call LinkGuard to return a result (lines 8-13). When there is no destination information (DNS name or dotted IP address) in the visual link (category 5), LinkGuard calls AnalyzeDNS to analyze the actual dns (lines 16 and 17). LinkGuard, therefore, handles all the 5 categories of phishing attacks. In AnalyzeDNS, if the actual dns name is contained in the blacklist, then we are sure that it is a phishing attack (lines 18 and 19). Similarly, if the actual dns is contained in the whitelist, it is, therefore, not a phishing attack (lines 20 and 21). If the actual dns is not contained in either whitelist or blacklist, PatternMatching is then invoked (line 22).

PatternMatching is designed to handle unknown attacks (blacklist/whitelist is useless in this case). For category 5 of the phishing attacks, all the information we have is the actual link from the hyperlink (since the visual link does not contain DNS or IP address of the destination site), which provide very little information for further analysis. In order to resolve this problem, we try two methods: First, we extract the sender e-mail address from the e-mail. Since phishers generally try to fool users by using (spoofed) legal DNS names in the sender e-mail address, we expect that the DNS name in the sender address will be different from that in the actual link. Second, we proactively collect DNS names that are manually input by the user when she surfs the Internet and store the names into a seed set, and since these names are input by the user by hand, we assume that these names are trustworthy. PatternMatching then checks if the actual DNS name of a hyperlink is different from

the DNS name in the sender's address (lines 23 and 24), and if it is quite similar (but not identical) with one or more names in the seed set by invoking the Similarity (lines 25-30) procedure.

Similarity checks the most chance of actual dns and also the DNS names in seed set. The similarity index between 2 strings is set by scheming the borderline range of changes (including insertion, deletion, or revision of a personality within the string) required to rework a string to the opposite string. If the amount of changes is zero, then the 2 strings square measure identical; if the amount of changes is little, then they are of high similarity; otherwise, they are of low similarity. For example, the similarity index of `microsoft' and `micr0s0ft' is 7/9 (since we need change the 2 `0's in micr0s0ft to `o'. Similarly, the similarity index of `paypal' and `paypal-cgi' is 6/10 (since we need to remove the last 4 chars from paypal-cgi), and the similarity index of `95559' and `955559' is 5/6 (since we need to insert a `5' to change `95559' to `955559').

If the two DNS names are similar but not identical, then it is a possible phishing attack. For instance, PatternMatching can easily detect the difference between www.icbc.com.cn (which is a good e-commence web site) and www.1cbc.com.cn (which is a phishing site), which has similarity index 75%. Note that PatternMatching may treat www.1cbc.com.cn as a normal site if the user had never visit www.1cbc.com.cn before. This false negative, however, is unlikely to cause any severe privacy or financial lose to the user, since she actually does not have anything to lose regarding the Web site www.icbc.com.cn (since she never visits that Web site before)!

False Positives and False Negatives Handling

Since LinkGuard is a rule-based heuristic algorithm, it may cause false positives (i.e., treat non-phishing site as phishing site) and false negatives (i.e., treat phishing site as non-phishing site). In what follows, we show that LinkGuard may result in false positives but is very unlikely to cause harmful false negatives.

For phishing attacks of category 1, we are sure that there is no false positive or false negatives, since the DNS names of the visual and actual links are not the same. It is also easy to observe that LinkGuard handles categories 3 and 4 correctly since the encoded links are first decoded before further analysis.

For category 2, LinkGuard could end in false positives, since victimisation dotted decimal science addresses rather than domain names could also be fascinating in some special circumstances (e.g., once the DNS names square measure still not registered). For category 5, LinkGuard may additionally end in false positives. To illustrate, we all know that each `www.iee.org' and `www.ieee.org' square measure legal internet sites. However, these 2 DNS names have a similarity index of 3/4, thus, is extremely possible to trigger a false positive.

When it is a possible false positive, LinkGuard will return a Possible Phishing. In our implementation (which will be described in the next section), we leverage the user to judge if it is a phishing attack by prompting a dialogue box with detailed information of the hyperlink. The rationale behind this choice is that users generally may have more knowledge of a link than a computer in certain circumstances (e.g., the user may know that the dotted decimal IP address is the address of his friend's computer and that www.iee.org is a respected site for electrical engineers). For category 5, LinkGuard may also result in false negatives. False negatives are more harmful than false positives, since attackers in this case will succeed in leading the victim to the phishing sites. For instance, when the sender's e-mail address and the DNS name in the actual link are the same and the DNS name in the actual link has a very low similarity index with the target site, LinkGuard will return No Phishing. For instance, PatternMatching will treat the below link as No Phishing.

 Click here

with “securehq@fdic-secure.com” as the sender address.

We note that this kind of false negatives is very unlikely to result in information leakage, since the end user is very unlikely to have information the attack interested (since the DNS name in this link is not similar with any legal Web sites).

IMPLEMENTATION AND VERIFICATION OF LINKGUARD

We have implemented the LinkGuard algorithm in Windows XP. It includes two parts: a whook.dll dynamic library and a LinkGuard executive. The structure of the implementation is depicted in Figure 1.

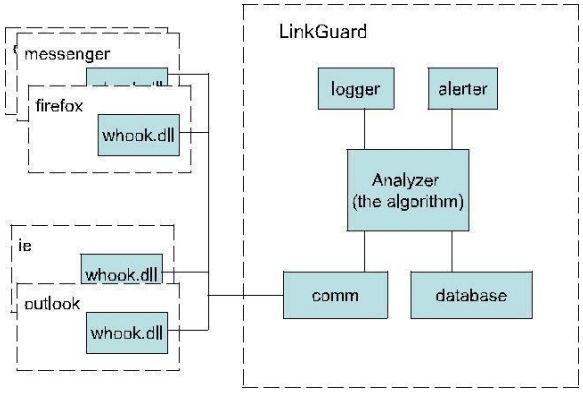

Fig. 1: *The Structure of the LinkGuard Implementation, which Consists of a whook.dll and a LinkGuard Executive.*

whook is a dynamic link library, it is dynamically loaded into the address spaces of the executing processes by the operating system. whook is responsible for collecting data, such as the called links and visual links, the user input URLs. More specifically, whook.dll is used to: 1) install a BHO (browser helper object) for IE to monitor user input URLs; 2) install an event hook with the SetWinEventHook provided by the Windows operating system to collect relevant information; 3) retrieve sender's e-mail address from Outlook; 4) analyze and filter the received windows and browser events passed by the BHO and the hook, and pass the analyzed data to the LinkGuard executive.

LinkGuard is the key component of the implementation. It is a standalone windows program with GUI (graphic user interface). It is composed of 5 parts as illustrated in Figure 1: Analyzer, Alerter, Logger, Comm, and Database. The functionalities of these 5 parts are given below:

Comm

Communicate with the whook.dll of all of the monitored processes, collect data related to user input from other processes (e.g., IE, outlook, firefox, etc.), and send these data to the Analyzer, it can also send commands (such as block the phishing sites) from the LinkGuard executive to whook.dll. The communication between the LinkGuard process and other processes is realized by the shared memory mechanism provided by the operating system.

Database

Store the whitelist, blacklist, and the user inputURLs.

Analyzer

It is the key component of LinkGuard, whichimplements the LinkGuard algorithm. It uses data provided by Comm and Database, and sends the results to the Alert and Logger modules.

Alerter

When receiving warning messages from Ana-lyzer, it shows the related information to alert the users and send back the reactions of the user back to the Analyzer.

Logger

Archive the history information, such as userevents, alert information, for future use.

After implemented the LinkGuard system, we have designed experiments to verify the effectiveness of our algorithm. Since, we are interested in testing LinkGuard's ability to detect unknown phishing attacks, we set both whitelist and blacklist to empty in our experiments. Our experiments showed that PhishgGuard can detect 195 phishing attacks out of the 203 APWG archives (with detection rate 96%). For the 8 unde-tected attacks, 4 attacks utilize certain Web site vulnerabilities. Hence the detecting rate is higher than 96% if category 5 is not included. Our experiment also showed that our implementation used by small amount of CPU time and memory space of the system. In a computer with 1.6G Pentium CPU and 512MB memory, our implementation consumes less than 1% CPU time and its memory footprint is less than 7MB.

Our experiment only used the phishing archive provided by APWG as the attack sources. We are planning to use LinkGuard in daily life to further evaluate and validate its effectiveness. Since we believe that a hybrid approach may be more effective for phihsing defense, we are also planning to include a mechanism to update the blacklist and whitelist in real-time.

CONCLUSION

Phishing has changing into a significant network security downside, inflicting fastidious lose of billions of greenbacks to each customers and e-commerce firms. And maybe additional basically, phishing has created e-commerce distrusted and fewer enticing to traditional customers. During this paper, we have studied the characteristics of the hyperlinks that were embedded in phishing e-mails. We have a tendency to then designed associate anti-phishing rule, Link-Guard, supported the derived characteristics. Since Phishig-Guard is characteristic primarily based, it cannot solely sight identified attacks, however, is also effective to the unknown ones. We have enforced LinkGuard for Windows XP. Our experiment showed that LinkGuard is light-weighted and may

find up to ninety six unknown phishing attacks in time period. We tend to believe that LinkGuard is not solely helpful for detective work phishing attacks, however can also defend users from malicious or unsought links in sites and Instant messages. Our future work includes more extending the LinkGuard algorithmic rule, in order that it will handle CSS (cross website scripting) attacks.

REFERENCES

1. I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, C.D. Spyropou. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with encrypted personal e-mail message. *In Proc. SIGIR 2000*, 2000.
2. The Anti-phishing working group. http://www.antiphishing.org/.
3. Neil Chou, Robert Ledesma, Yuka Teraguchi, Dan Boneh, John C. Mitchell. Client-side defense against web-based identity theft. *In Proc. NDSS 2004*, 2004.
4. Cynthia Dwork, Andrew Goldberg, Moni Naor. On memory-bound functions for fighting spam. In *Proc. Crypto 2003*, 2003.
5. EarthLink. ScamBlocker. http://www.earthlink.net/software/free/toolbar/.
6. David Geer. Security technologies go phishing. *IEEE Computer*. 2005; 38(6): 18–21p.
7. John Leyden. Trusted search software labels fraud site as 'safe'. http://www.theregister.co.uk/2005/09/27/untrusted search/.
8. Microsoft. Sender ID Framework. http://www.microsoft.com/ mscorp/safety/technologies/senderid/default.mspx.
9. Netcraft. Netcraft toolbar. http://toolbar.netcraft.com/.
10. PhishGuard.com. Protect Against Internet Phishing Scams. http://www.phishguard.com/.
11. Jonathan B. Postel. Simple Mail Transfer Protocol. RFC821: http://www.ietf.org/rfc/rfc0821.txt.
12. Georgina Stanley. Internet Security - Gone phishing. http://www.cyota.com/news.asp?id=114.
13. Meng Weng Wong. Sender ID SPF. http://www.openspf.org/whitepaper.pdf.