A

Seminar Report

on

PHISHING DETECTION ON WEBSITE

Submitted by

Aarzin Todiwala(201303100910043)

In partial fulfilment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

Computer Engineering

Under the guidance of

Prof. Dipak Dabhi

Assistant Professor

Department of Computer Engineering & Information Technology

CGPIT, UTU, Bardoli, Gujarat.

Chhotubhai Gopalbhai Patel Institute of Technology

Uka Tarsadia University, Bardoli

MAY 2016

CERTIFICATE

This is to certify that the seminar report entitled "***Phishing Detection on Website*** " has been carried out by ***Aarzin Todiwala(201303100910043)*** under my guidance in partial fulfilment of the degree of Bachelor of Technology in Computer Engineering, Chhotubhai Gopalbhai Patel Institute of Technology, UTU, Bardoli during the academic year 2015-2016.

DATE:

PLACE:

Prof. Dipak Dabhi
Assistant Professor
Dept. of CE & IT,CGPIT

Prof. Devendra V. Thakor
Head of Department
Dept. of CE & IT,CGPIT

....
Signature of Examiner

Chhotubhai Gopalbhai Patel Institute of Technology
Uka Tarsadia University, Bardoli

ACKNOWLEDGEMENT

It gives me great pleasure to present my seminar report on " PHISHING DETECTION ON WEBSITE". No work, however big or small, has ever been done without the contributions of others. First of all I would like to thank supreme power the Almighty God who is obviously the one has always guided me to work on the right path of life. Without his grace this project could not become reality. Next to him are my parents, whom I am greatly indebted for me brought up with love and encouragement to this stage.

It would be a great pleasure to write a few words, which would although not suffice as the acknowledgement of this long cherished effort, but in the absence of which this report would necessarily be incomplete. So these words of acknowledgement come as a small gesture of gratitude towards all those people, without whom the successful completion of this project would not have been possible.

I would like to express deep gratitude towards Prof. Dipak Dabhi (Assistant Professor, Dept. of CE & IT,CGPIT) who gave me their valuable suggestions, motivation and the direction to proceed at every stage. I would also like to express my same gratitude to Prof. Devendra V. Thakor(Head of Department, Dept. of CE & IT,CGPIT). They are like beam of light for us. Their kind guidance showed us the path of life and is unforgettable. They extend towards their valuable guidance, indispensable help and inspiration at times in appreciation I offer them my sincere gratitude. Last but not the least we would like to thank the Department of Computer Engineering,UTU,Bardoli for providing me with the facilities to lab, and all staff members of communication lab, it would have been impossible for me to complete my project without their valuable guidance & prompt cooperation.

I have tried my level best to make this seminar report error free, but I regret for errors, if any.

Place : Todiwala Aarzin M.
Date: (201303100910043)

Abstract

Abstract—Phishing is an act of stealing personal and sensitive user information through internet and using it for financial transactions. The goal of phishers is to carry out fraudulent transactions on behalf of the victims by using the information stealed from them. Availing the services of internet has become a dangerous task to the common people with these kinds of attacks. Many methods have been developed to fight against phishing attacks. But, as the attacker uses more sophisticated techniques each method fails to perform well in detecting the attacks. Here we propose a string matching method for detecting phishing attacks, which determines the degree of similarity a URL is having with the blacklisted URLs. Thus based on the textual properties of a URL it can be classified as phishing or non-phishing. Two string matching algorithms i.e. Longest Common Subsequence (LCS) and Edit Distance are used in the hostname comparison. The accuracy rate obtained for LCS is 99.1% and for Edit Distance it is 99.5%. And a new end-host based anti-phishing algorithm, which we call Link Guard, by utilizing the generic characteristics of the hyperlinks in phishing attacks. These characteristics are derived by analyzing the phishing data archive provided by the Anti-Phishing Working Group (APWG). Because it is based on the generic characteristics of phishing attacks, Link Guard can detect not only known but also unknown phishing attacks. We have implemented Link Guard in Windows XP. Our experiments verified that Link Guard is effective to detect and prevent both known and unknown phishing attacks with minimal false negatives. Link Guard successfully detects 195 out of the 203 phishing attacks. Our experiments also showed that Link Guard is light weighted and can detect and prevent phishing attacks in real time.

Contents

List of Figures

List of Tables

1 INTRODUCTION

The word Phishing initially emerged in 1990s. The early hackers often use ’ph’ to replace ’f’ to produce new words in the hacker’s community, since they usually hack by phones. Phishing is a new word produced from ’fishing’, it refers to the act that the attacker allure users to visit a faked Website by sending them faked e-mails (or instant messages), and stealthily get victim’s personal information such as user name, password, and national security ID, etc. These information then can be used for future target advertisements or even identity theft attacks (e.g., transfer money from victims bank account). The frequently used attack method is to send e-mails to potential victims, which seemed to be sent by banks, online organizations, or ISPs. In these e-mails, they will makeup some causes, e.g. the password of your credit card had been mis-entered for many times, or they are providing upgrading services, to allure you visit their Website to conform or modify your account number and password through the hyperlink provided in the e-mail. You will then be linked to a counterfeited Website after clicking those links[1].
The style, the functions performed, sometimes even the URL of these faked Websites this work was supported by the National Natural Science Foundation of China (NSFC) under contract No. 60503049. are similar to the real Web site. It’s very difficult for you to know that you are actually visiting a malicious site. If you input the account number and password, the attackers then Successfully collect the information at the server side, and is able to perform their next step actions with that information (e.g., withdraw money out from your account).Phishing itself is not a new concept, but it’s increasingly used by phishers to steal user information and perform business crime in recent years. Within one to two years, the number of phishing attacks increased dramatically.According to Gartner Inc., for the 12 months ending April 2004, ’there were 1.8 million phishing attack victims, and the fraud incurred by phishing victims totaled $1.2 billion’ .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%). The infamous phishing attacks happened in China in recent years include the events to counterfeit the Bank of China (real Web site www.bank-ofchina.com, counterfeited Web site www.bank-offchina.com),the Industrial and Commercial Bank.cn, faked website www.1cbc.com.cn), the Agricultural Bank of China (real website www.95599.com,faked Web site www.965555.com), etc. In this research paper, we study the commonprocedure of phishing attacks and review possible anti-phishing approaches. We then focus on end-host based antiphishing approach. We first analyze the common characteristics of the hyperlinks in phishing e-mails[1].

1.1 Background

Such effects include corporate information loss and national security secrets. This work shows that phishing attacks are increasingly more invasive and sophisticated, e.g. spear phishing with personalized content. Hong highlights phishing countermeasures including (1) filtering, black listing sites, and taking down sites, (2) user interface assistance, e.g. better and more appropriate warnings, and (3) proactive user training towards the improved recognizing and avoiding of attacks[2].

Many studies focus on classifier algorithms as a first line of defense for improved detection of malicious email or websites. For instance, investigate user awareness of website design. They found that attacks were more successful using various forms of visual deception that rely on a lack of user knowledge of security indicators. Findings in suggest that these cues may be missed because of positive site features that elicit trust, such as brand logos or mass notifications by a recognized bank relying on a moderate chance that the user has an account[2].

1.2 Motivation

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

1) Phishers set up a counterfeited Web site which looks exactly like the legitimate Web site, including setting up the web server, applying the DNS server name, and creating the web pages similar to the destination Website, etc[2].

2) Send large amount of spoofed e-mails to target users in the name of those legitimate companies and organizations, trying to convince the potential victims to visit their Web sites[2].

3) Receivers receive the e-mail, open it, and click the spoofed hyperlink in the e-mail, and input the required information[2].

4) Phishers steal the personal information and perform their fraud such as transferring money from the victims account[2].

1.3 Objectives

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 research 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[2].

1.3.1 Detect and block the phishing Web sites in time:

If we can detect the phishing Web sites in time, we then can block the sites and prevent phishing attacks. It's relatively easy to (manually) determine whether a site is a phishing site or not, but it's difficult to find those phishing sites out in time. Here we list two methods for phishing site detection. 1) The Web master of a legal Web site periodically scans the root DNS for suspicious sites (e.g. www.1cbc.com.cn vs. www.icbc.com.cn)[2].

Since the phisher must duplicate the content of the target site, he must use tools to (automatically) download the Webpages from the target site. It is therefore possible to detect this kind of download at the Web server and trace back to the phisher. Both approaches have shortcomings. For DNS scanning, it increases the overhead of the DNS systems and may cause problem for normal DNS queries, and furthermore, many phishing attacks simply do not require a DNS name. For phishing download detection, clever phishers may easily write tools which can mimic the behavior of human beings to defeat the detection[2].

1.3.2 Enhance the security of the web sites:

The business Websites such as the Web sites of banks can take new methods to guarantee the security of users personal information. One method to enhance the security is to use hardware devices. For example, the Barclays bank provides a hand-held card reader to the users. Before shopping in the net, users need to insert their credit card into the card reader, and input their (personal identification number) PIN code, then the card reader will produce a onetime security password, users can perform transactions only after the right password is input. Another method is to use the biometrics characteristic (e.g. voice, fingerprint, iris, etc.) for user authentication. For example, PayPal had tried to replace the single password verification by voice recognition to enhance the security of the Website.With these methods, the phishers cannot accomplish their tasks even after they have gotten part of the victims information .However, all these techniques need additional hardware to realize the authentication between the users and the Web sites, hence will increase the cost and bring certain inconvenience.Therefore, it still needs time for these techniques to be widely adopted[2].

1.3.3 Block the phishing e-mails by various spam filters:

Phishers generally use e-mails as bait to allure potential victims. SMTP (Simple Mail Transfer Protocol) is the protocol to deliver e-mails in the Internet. It is a very simple protocol which lacks necessary authentication mechanisms. Information related to sender, such as the name and email address of the sender, route of the message, etc., can be counterfeited in SMTP. Thus, the attackers can send out large amounts of spoofed e-mails which are seemed from legitimate organizations. The phishers hide their identities when sending the spoofed e-mails, therefore, if anti-spam systems can determine whether an e-mail is sent by the announced sender (Am I Whom I Say I Am?), the phishing attacks will be decreased dramatically. From this point, the techniques that preventing senders from counterfeiting their Send ID (e.g. SIDF of Microsoft) can defeat phishing attacks efficiently[2].

SIDF is a combination of Microsoft's Caller ID for E-mail and the SPF (Sender Policy Framework) developed by Meng Weng Wong. Both Caller ID and SPF check e-mail sender's domain name to verify if the e-mail is sent from a server that is authorized to send e-mails of that domain and from that to determine whether that e-mail use spoofed e-mail address. If it's faked, the Internet service provider can then determine that e-mail is a spam e-mail[2].

The spoofed e-mails used by phishers are one type of spam e-mails. From this point of view, the spam filters can also be used to filter those phishing e-mails. For example, blacklist, whitelist, keyword filters, Bayesian filters with self learning abilities, and E-Mail Stamp, etc., can all be used at the e-mail server or client systems. Most of these anti-spam techniques perform filtering at the receiving side by scanning the contents and 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[2].

1.3.4 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 antiphishing tools in use today can be divided into two categories: blacklist/whitelist based and rule-based[2].

Category I: When a user visits a Web site, the antiphishing 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. 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[2].

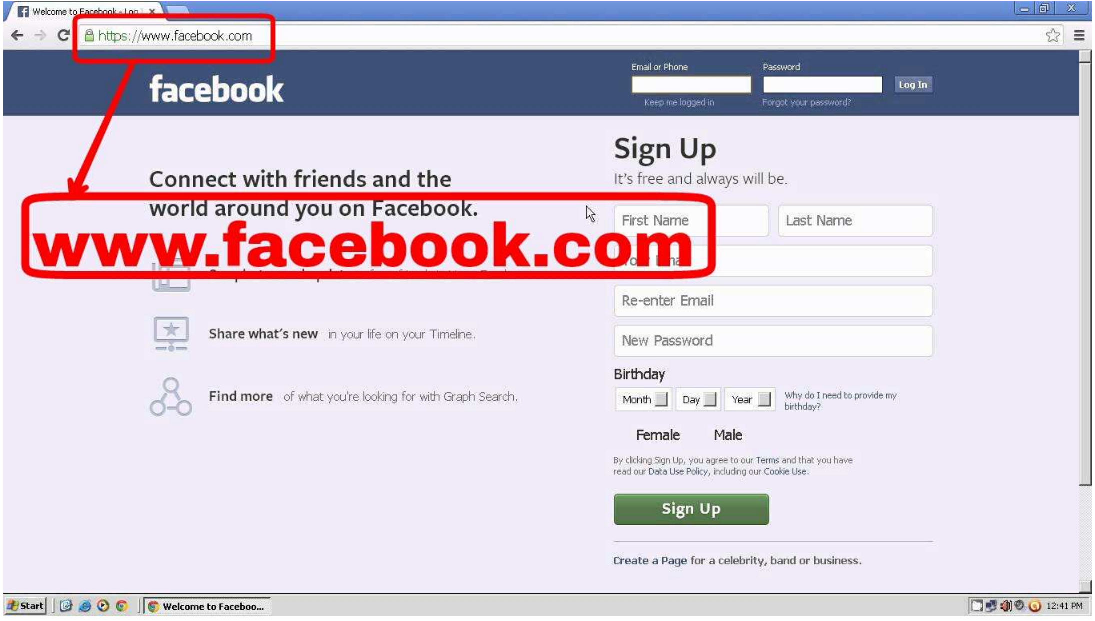
Category II: this category of tools uses certain rules in their software and checks the security of a Website according to these rules. Examples of this type of tools includeSpoof Guard developed by Stanford, Trust Watch of the GeoTrust, etc. SpoofGuard checks the domain name, URL (includes the port number) of a Web site, it also checks whether the browser is directed to the current URL via the links in the contents of e-mails. If it finds that the domain name of the visited Web site is similar to a well-known domain name, or if they are not using the standard port, Spoof Guard will warn the users. In TrustWatch, the security of a Web site is determined by whether it has been reviewed by an independent trusted third party organization. Both SpoofGuard and TrustWatch provide a toolbar in the browsers to notify their users whether the Web site is verified and trusted. It is easy to observe that all the above defense methods are useful and complementary to each other, but none of them are perfect at the current stage. In the rest of the research paper, we focus on end-host based approach and propose an end host based LinkGuard algorithm for phishing detection and prevention. To this end, our work follows the same approach. Our work differs from in that: 1) LinkGuard is based on our careful analysis of the characteristics of phishing hyperlinks whereas Spoof Guard is more like a framework; 2) LinkGuard has a verified very low false negative rate for unknown phishing attacks whereas the false negative proper of Spoof Guard is still not known. In next section, we first study the characteristics of the hyperlinks in phishing e-mails and then we propose the LinkGuard algorithm[2].

1.4 Applications:

- From some equations and
- SetWinEventHook, BHO (browser helper object).

1.4.1 For Example:

- Original Website



Connect with friends and the world around you on Facebook.

It's free and always will be.

Share what's new in your life on your Timeline.

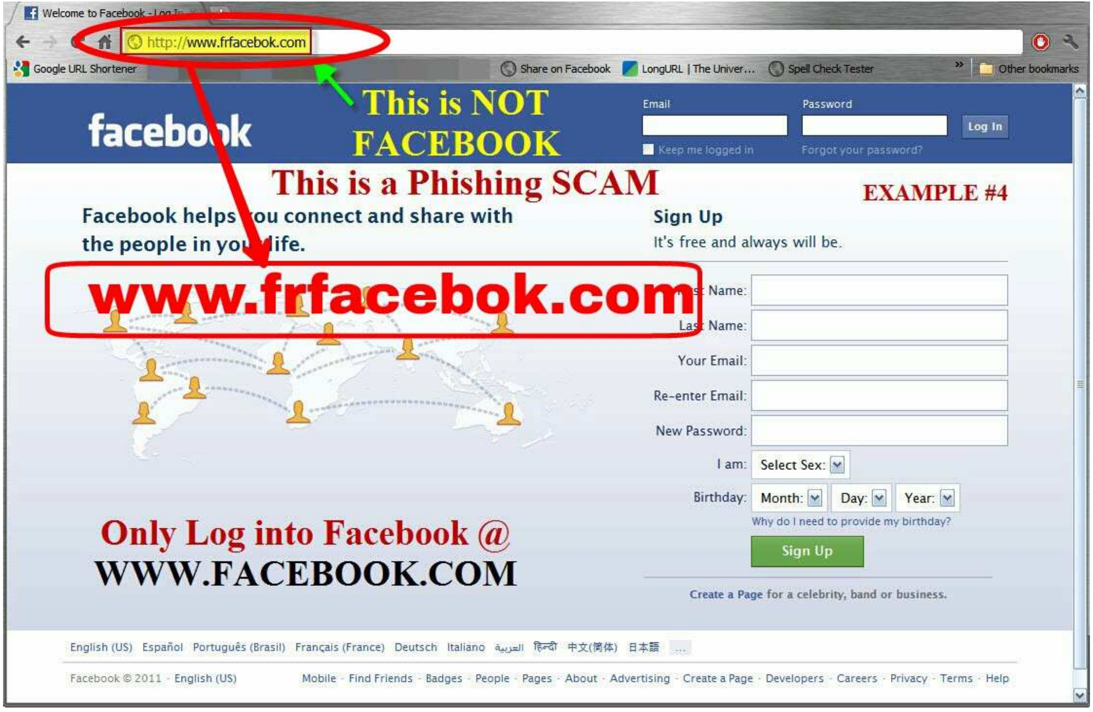
Find more of what you're looking for with Graph Search.

Why do I need to provide my birthday?

By clicking Sign Up, you agree to our Terms and that you have read our Data Use Policy, including our Cookie Use.

Create a Page for a celebrity, band or business.

- Fake Page



This is NOT FACEBOOK

This is a Phishing SCAM

EXAMPLE #4

www.frfacebok.com

Only Log into Facebook @ WWW.FACEBOOK.COM

Facebook helps you connect and share with the people in your life.

Sign Up

It's free and always will be.

Welcome to Facebook - Log In
http://www.frfacebok.com
Google URL Shortener
Share on Facebook
LongURL | The Univer...
Spell Check Tester
Other bookmarks
facebook
Email
Password
Log In
Keep me logged in
Forgot your password?
First Name:
Last Name:
Your Email:
Re-enter Email:
New Password:
I am: Select Sex:
Birthday: Month: Day: Year:
Why do I need to provide my birthday?
Sign Up
Create a Page for a celebrity, band or business.
English (US) Español Português (Brasil) Français (France) Deutsch Italiano العربية हिन्दी 中文(简体) 日本語
Facebook © 2011 · English (US)
Mobile · Find Friends · Badges · People · Pages · About · Advertising · Create a Page · Developers · Careers · Privacy · Terms · Help

2 LITERATURE SURVEY

2.1 Approximate String Matching Algorithm for Phishing Detection

Authors: Dona Abraham, Nisha S Raj

Publication: Year: 2014

Summary:

Many methods have been developed to fight against phishing attacks. But, as the attacker uses more sophisticated techniques each method fails to perform well in detecting the attacks. Here we propose a string matching method for detecting phishing attacks, which determines the degree of similarity a URL is having with the blacklisted URLs. Thus based on the textual properties of a URL it can be classified as phishing or non-phishing.

Two string matching algorithms i.e. Longest Common Subsequence (LCS) and Edit Distance are used in the hostname comparison. The accuracy rate obtained for LCS is 99.1% and for Edit Distance it is 99.5%.

2.2 Intelligent Phishing Website Detection and Prevention System by Using Link Guard Algorithm

Authors: U.Naresh, U.Vidya Sagar, C.V. Madhusudan Reddy

Publication Year: Sep.- Oct. - 2013

Summary:

By utilizing the generic characteristics of the hyperlinks in phishing attacks. These characteristics are derived by analyzing the phishing data archive provided by the Anti-Phishing Working Group (APWG). Because it is based on the generic characteristics of phishing attacks, Link Guard can detect not only known but also unknown phishing attacks. We have implemented Link Guard in Windows XP. Our experiments verified that Link Guard is effective to detect and prevent both known and unknown phishing attacks with minimal false negatives.

Link Guard successfully detects 195 out of the 203 phishing attacks. Our experiments also showed that Link Guard is light weighted and can detect and prevent phishing attacks in real time.

3 METHODOLOGY

3.1 URL Classification Algorithm

The idea behind our method is to check the amount of resemblance of a URL with the list of phishing URLs. Here instead of performing an entire URL match, the incoming URL is divided into different tokens and check the amount of similarity with the corresponding tokens of the blacklisted URLs. The scores are calculated based on the number of the occurrence of each token in the blacklist. These tokens used for approximate matching are identified from the lexical structure of a URL. The typical structure of a URL.[1]

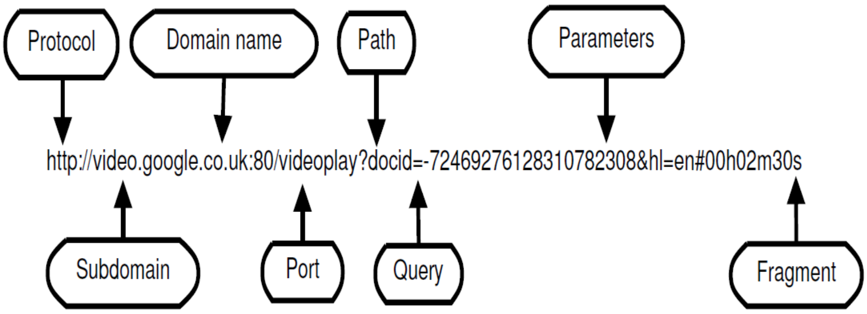

Protocol
Domain name
Path
Parameters
http://video.google.co.uk:80/videoplay?docid=-7246927612831078230&hl=en#00h02m30s
Subdomain
Port
Query
Fragment

Fig. 1: Structure of a URL

where,

• protocol declares how the web browser communicate with a web server on fetching a document.

• subdomain is a sub-division of the main domain name.

• domain name is a reference that identifies a website on the internet. A domain name always includes a top level domain.

• port number in a URL specifies port through which the web resource is served and HTTP uses port number 80.

• path is used to specify the location of the requested resource on the web server.

• query contains the data to be passed to the server.

• parameters are the name-value pairs in a query.

• fragment if present, specifies a part within the overall resource.

The four tokens considered for our method are IP Address, Hostname, Directory Structure and Brand names. The flow of the proposed method is shown in Figure 2. Individual scores are calculated for each entities and the final score is calculated as a weighted sum. If the final score is greater than the threshold then the URL can be flagged as phishing.[1]

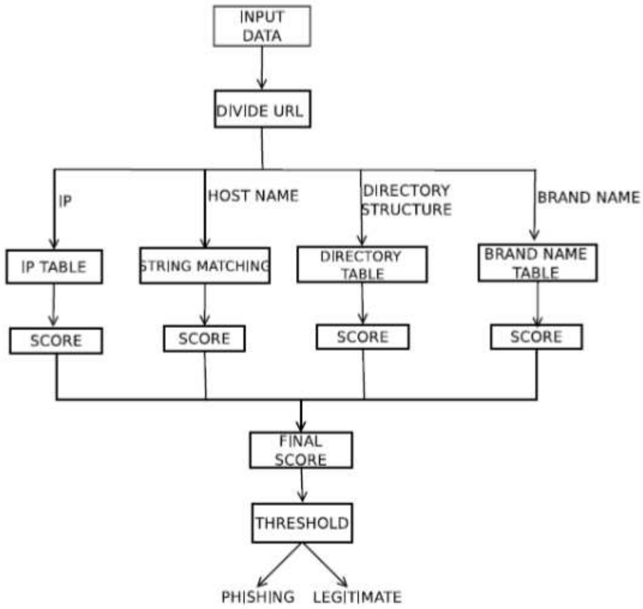


Figure 1: Classification of URL

3.1.1 IP Address: An IP address is a numerical label assigned to each device in a computer network that uses Internet Protocol for com- munication. Each Web site on the internet have at least one Internet Protocol (IP) address. But this address is not always shown in Web browsers on visiting a web site. The utility ‘ping’ is used to contact a web site by its name and it return the IP address. The ping method will fail if the Web site is unreachable temporarily or permanently. Host names are the entity used to obtain the IP address. Here, we collect IPs of all URLs in the blacklist but some attempts failed since the websites are removed. An indexed table is generated for the IPs in such a way that the IP address is the key and the values associated with a key are the URLs with key as IP address.[1]

The number of URLs associated with each key is the pa- rameter used for calculating the score. Based on this parameter a score is calculated for each distinct IPs in the table, using the equation 1

$$score = \frac{n_i - min\{n_i\}}{max\{n_i\} - min\{n_i\} + 1} \qquad (1)$$

where where min{n i } (max{n i }) is the minimum (maxi- mum) of the number of phishing URLs hosted by blacklisted IP addresses.[1]

3.1.2 Directory Structure: Directory specifies the location of the requested resource on the web server. Phishers can host more than one attack from the same directory., i.e by changing the filename of a URL, phishers can launch any number of attacks from a single directory. The directories of each URL is organised to an indexed table as in the case of IP then the score for each directory is calculated based on the equation 1 where n i being the number of URLs with same directory.[1]

3.1.3 Hostname: Hostname is the name with which a website is identified on the internet. For example: google.com, paypal.com etc. Phishers plant attacks on well reputed hostname by including such hostnames as a subdomain or as a path component in their phishing URL. Therefore, a direct matching of hostnames of suspicious URLs is not a reliable method, thus here we go for approximate matching. First the host names are extracted from phishing URLs. Then the hostname of an incoming URL is checked for its similarity using approximate string matching. The maximum similarity rate obtained is the score for that hostname. Two algorithms are used here for approximate string match- ing. i.e. Longest Common Subsequence and Edit Distance. [1]

- Longest Common Subsequence (LCS): LCS is to find the longest subsequence common to a set of sequences. In LCS, the problem is broken down into smaller, but simpler subproblems, until the solution become trivial. The solution to a higher problem depends on the solution to several lower subproblems. The LCS function is defined as follows. Let two sequences X and Y are defines as follows: X = (x 1 , x 2 ...x m) and Y = (y 1 , y 2 ...y n). Let LCS(X i , Y j), the longest common subsequence of X i and Y j is given by:[1]

$$
LCS(X_i, Y_j) = \begin{cases} \phi & if\ i = 0\ or\ j = 0 \\ LCS(X_{i-1}, Y_{j-1}) + 1 & if\ \ x_i = y_j \\ longest(LCS(X_i, Y_{j-1}), LCS(X_{i-1}, Y_j)) & if\ x_i \neq y_j \end{cases}
$$

- Edit Distance: It is a method of quantifying how dissimilar two strings are by counting the minimum number of operations required to transform one string into another. The edit distance between a = a 1 ...a i and b = b 1 ...b j is given by d a,b and is defined as:

$$
d_{a,b}(i,j) = \begin{cases} max(i,j) & if\ min(i,j) = 0 \\ min \begin{cases} d_{a,b}(i-1,j) + 1 \\ d_{a,b}(i,j-1) + 1 \\ d_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & otherwise \end{cases}
$$

The value d a,b shows the rate of dissimilarity between a and b. The similarity rate is calculated by subtracting it from 1 as shown in equation 2

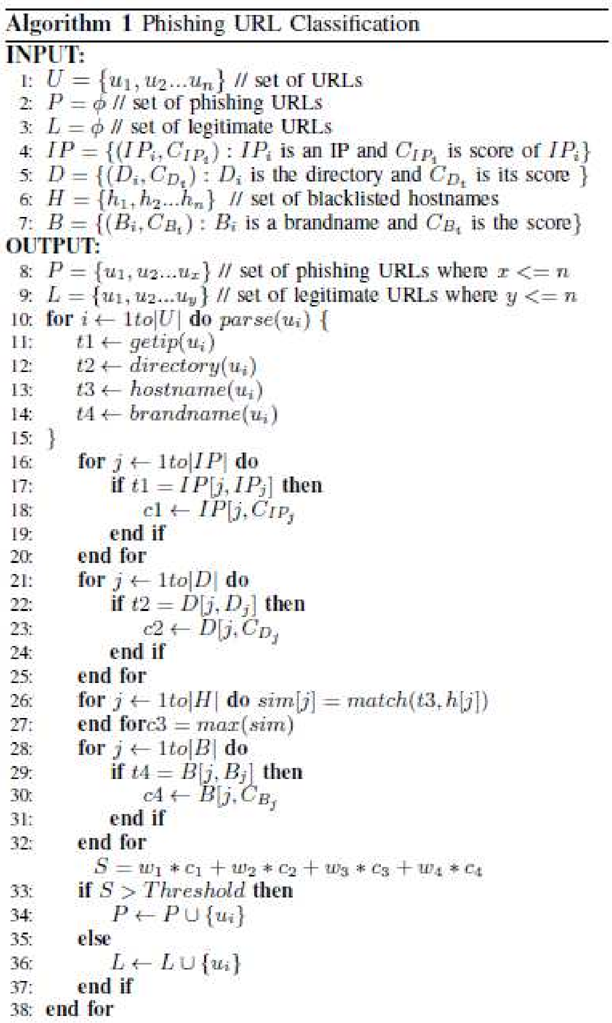
$$sim(a,b) = 1 - d_{a,b} \qquad (2)$$

3.1.4 Brand Names: Phishers usually target on well reputed sites where money transactions takes place. Based on the PhishTank reports the most affected brands are online shopping sites line Paypal, eBay and bank websites. A list of most affected brand names are generated with the help of PhishTank reports. In our method, we check for the existence of any of these brands in the URL and a score is assigned based on the frequency of it using the equation 1. In testing phase, if an incoming URL contains a brand name of our list, then the corresponding score is assigned for the brand name token of that URL.[1]

3.1.5 Atlast the Sumation of the All the values:[1]

$$score = \sum_{i=1}^{4} w_i \times c_i$$

3.1.6 Algorithm:



Algorithm 1 Phishing URL Classification

```
INPUT:
  $U = \{u_1, u_2 ... u_n\}$ // set of URLs
  $P = \phi$ // set of phishing URLs
  $L = \phi$ // set of legitimate URLs
  $IP = \{(IP_i, C_{IP_i}) : IP_i$ is an IP and $C_{IP_i}$ is score of $IP_i\}$
  $D = \{(D_i, C_{D_i}) : D_i$ is the directory and $C_{D_i}$ is its score $\}$
  $H = \{h_1, h_2 ... h_n\}$ // set of blacklisted hostnames
  $B = \{(B_i, C_{B_i}) : B_i$ is a brandname and $C_{B_i}$ is the score$\}$
OUTPUT:
  $P = \{u_1, u_2 ... u_x\}$ // set of phishing URLs where $x <= n$
  $L = \{u_1, u_2 ... u_y\}$ // set of legitimate URLs where $y <= n$
  for $i \leftarrow 1 to |U|$ do $parse(u_i)$ {
      $t1 \leftarrow getip(u_i)$
      $t2 \leftarrow directory(u_i)$
      $t3 \leftarrow hostname(u_i)$
      $t4 \leftarrow brandname(u_i)$
  }
      for $j \leftarrow 1 to |IP|$ do
          if $t1 = IP[j, IP_j]$ then
              $c1 \leftarrow IP[j, C_{IP_j}$
          end if
      end for
      for $j \leftarrow 1 to |D|$ do
          if $t2 = D[j, D_j]$ then
              $c2 \leftarrow D[j, C_{D_j}$
          end if
      end for
      for $j \leftarrow 1 to |H|$ do $sim[j] = match(t3, h[j])$
      end for$c3 = max(sim)$
      for $j \leftarrow 1 to |B|$ do
          if $t4 = B[j, B_j]$ then
              $c4 \leftarrow B[j, C_{B_j}$
          end if
      end for
        $S = w_1 * c_1 + w_2 * c_2 + w_3 * c_3 + w_4 * c_4$
      if $S > Threshold$ then
          $P \leftarrow P \cup \{u_i\}$
      else
          $L \leftarrow L \cup \{u_i\}$
      end if
  end for
```

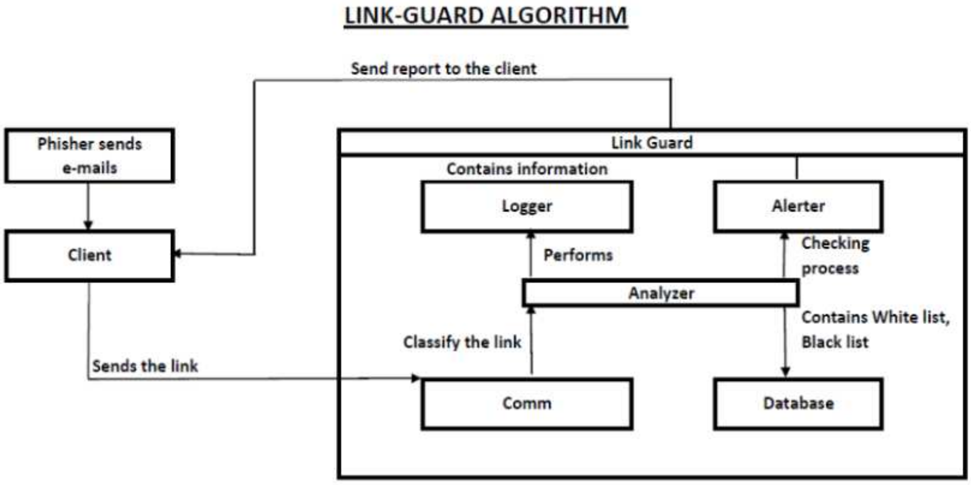


Figure 2: Link- Guard Algoritm

3.2 Link Guard Algorithm:

We have implemented the LinkGuard algorithm in Windows XP. It includes two parts: a whook.dll dynamic library and a LinkGuard executive.[2]

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: [2]

1) install a BHO (browser helper object) for IE to monitor user input URLs; [2]

2) install an event hook with the SetWinEventHook provided by the Windows operating system to collect relevant information; [2]

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. [2]

It is a stand alone windows program with GUI (graphic user interface). Analyzer, Alerter, Logger, Comm, and Database.The functionalities of these 5 parts are given below:[2]

Comm: Communicate with the whook.dll of all of themonitored processes, collect data related to user input fromother processes (e.g. IE, outlook, firefox, etc.), and send these data to the Analyzer, it can also send commands (suchas block the phishing sites) from the LinkGuard executiveto whook.dll. The communication between the LinkGuard process and other processes is realized by the shared memory mechanism provided by the operating system. [2]

Database: Store the whitelist, blacklist, and the user input URLs.[2]

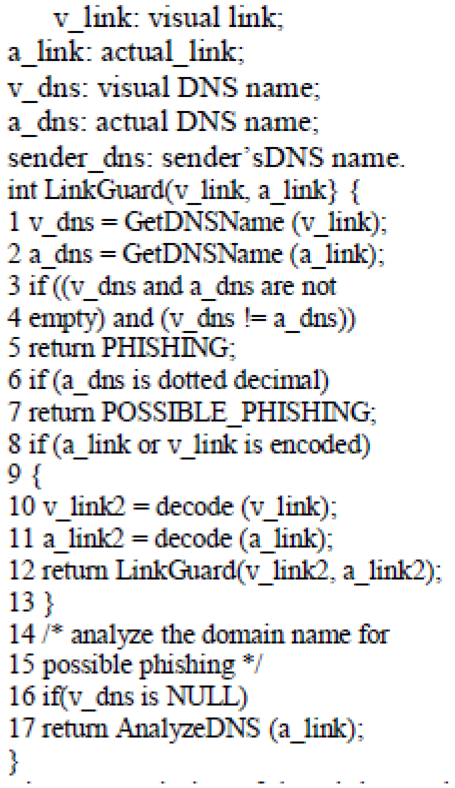
Analyzer: It is the key component of LinkGuard, which implements the LinkGuard algorithm, it uses data provided by Comm and Database, and sends the results to the Alertand Logger modules. [2]

Alerter: When receiving warning messages from Analyzer, it shows the related information to alert the users and send back the reactions of the user back to the Analyzer. [2]

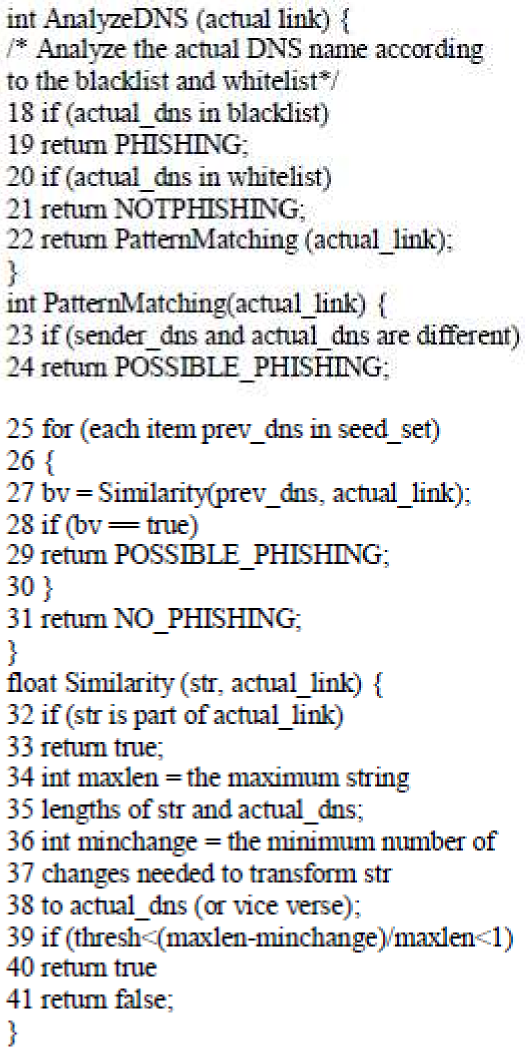
Logger: Archive the history information, such as use revents, alert information, for future use. [2]

After implemented the LinkGuard system, we have designed experiments to verify the effectiveness of our algorithm.[2] Since we are interested in testing Link Guard's ability to detect unknown phishing attacks, we set both whitelist and black list to empty in our experiments. Our experiments showed that Phishing Guard can detect 195 phishing attacks out of the 203APWG archives (with detection rate 96%). For the 8 undetected 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 512MBmemory, 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.

Algorithm:



```
    v_link: visual link;
a_link: actual_link;
v_dns: visual DNS name;
a_dns: actual DNS name;
sender_dns: sender'sDNS name.
int LinkGuard(v_link, a_link} {
v_dns = GetDNSName (v_link);
a_dns = GetDNSName (a_link);
if ((v_dns and a_dns are not
empty) and (v_dns != a_dns))
return PHISHING;
if (a_dns is dotted decimal)
return POSSIBLE_PHISHING;
if (a_link or v_link is encoded)
{
v_link2 = decode (v_link);
a_link2 = decode (a_link);
return LinkGuard(v_link2, a_link2);
}
/* analyze the domain name for
possible phishing */
if(v_dns is NULL)
return AnalyzeDNS (a_link);
}
```



```
int AnalyzeDNS (actual link) {
/* Analyze the actual DNS name according
to the blacklist and whitelist*/
    if (actual_dns in blacklist)
        return PHISHING;
    if (actual_dns in whitelist)
        return NOTPHISHING;
    return PatternMatching (actual_link);
}
int PatternMatching(actual_link) {
    if (sender_dns and actual_dns are different)
        return POSSIBLE_PHISHING;

    for (each item prev_dns in seed_set)
    {
        bv = Similarity(prev_dns, actual_link);
        if (bv == true)
            return POSSIBLE_PHISHING;
    }
    return NO_PHISHING;
}
float Similarity (str, actual_link) {
    if (str is part of actual_link)
        return true;
    int maxlen = the maximum string
    lengths of str and actual_dns;
    int minchange = the minimum number of
    changes needed to transform str
    to actual_dns (or vice verse);
    if (thresh<(maxlen-minchange)/maxlen<1)
        return true
    return false;
}
```

Fig2 The subroutines used in the LinkGuard algorithm.

3.3 Comparative Analysis

Table 1: Comparative Analysis

Name of Technique	Accuracy	Used For	Any Softwares
Link-Guard Algorithm	96%	Emails	Yes,BHO & SetWinEventHook.
URL Classification Algorithm	99%	Websites and Others	No, Only Formulas

4 CONCLUSION AND FUTURE SCOPE

Phishing has becoming a serious network security problem, causing finical lose of billions of dollars to both consumers and e-commerce companies. In this research paper, we have studied the characteristics of the hyperlinks that were embedded in phishing e-mails. We then designed an anti phishing algorithm, Link-Guard, based on the derived characteristics and URL classification.

References

[1] Dona Abraham and Nisha S. Raj, Approximate String Matching Algorithm for Phishing Detection,In SCMS School of Enggineering and Technology,2014 IEEE.

[2] U. Naresh , U. Vidya Sagar and C.V.Madhusudan Reddy,Intelligent Phishing Website Detection and Prevention System by Using Link Guard Algorithm, IOSR Journal of Computer Engineering, 2013.