Journal of Computer Science, 9 (4): 526-533, 2013
ISSN 1549-3636
© 2013 Science Publications
doi:10.3844/jcssp.2013.526.533 Published Online 9 (4) 2013 (http://www.thescipub.com/jcs.toc)

Protecting the Online User's Information Against Phishing Attacks Using Dynamic Encryption Techniques

C. Emilin Shyni and S. Swamynathan

Department of Information Science and Technology,
Anna University, Chennai-600025, Tamilnadu, India

Received 2012-07-09, Revised 2012-08-24; Accepted 2013-11-08

ABSTRACT

A phishing attack is a criminal activity which mimics a certain legitimate webpage using a fake webpage with an intention of luring end-users to visit the fake website thereby stealing their personal information such as usernames, passwords and other personal details such as credit card information. Phishing has seen an alarming trend of increase in both the volume and the sophistication of phishing attacks. According to a description of phishing by APWG, the ways phishers steal consumers' personal information consist of social engineering and technical subterfuge. In technical-subterfuge schemes, phishers furtively plant crime ware onto users' computers to intercept their online account user names and passwords, while in social-engineering schemes they send spoofed e-mails to consumers purporting to be from legitimate businesses and agencies and then mislead consumers to counterfeit websites. When a user wants to access the website, the server sends an encrypted security code to the user through the communication protocol. If the user's login name is not valid it will show an error message. If the user's name is valid, the website checks the user's registered account and sends an acknowledgement to that user. The legitimate or true webpage mimicked by the fake webpage is defined as the phishing target. Such phishing attacks if executed on newly created web pages prove difficult to identify as it becomes hard to tell which the phishing page is and which the target is. We anticipate that our approach would be deployed for websites requiring a high level of security and that it would ultimately help in remaining customer confidence in using web-based commerce. The automatic discovery of phishing target is proposed to solve the above problem.

Keywords: Website, Anti-Phishing, Spoofed E-Mails, Attacks

1. INTRODUCTION

Security is fundamentally about protecting assets. Assets may be tangible items, such as a Web page or customer database. A threat is any potential occurrence, malicious or otherwise, that could harm an asset. A threat can be created through vulnerabilities, which is a weakness that makes a threat possible. This may be because of poor design, configuration mistakes, or inappropriate and insecure coding techniques. Weak input validation is an example of an application layer vulnerability, which can result in input attacks. An attack is an action that exploits vulnerability or enacts a threat. Examples of attacks include sending malicious input to an application or flooding a network in an attempt to deny service. It is not possible to design and build a secure Web application until we know about threats. An increasingly important discipline and one that is recommended to form part of the application's design phase is threat modeling. Wardman *et al*. (2009) The purpose of threat modeling is to analyze the application's architecture and design and identify potentially

Corresponding Author: C. Emilin Shyni, Department of Information Science and Technology, Anna University, Chennai, Tamilnadu, India Tel: 91 44 22358797 Fax: 91 44 22201213

vulnerable areas that may allow a user, perhaps mistakenly, or an attacker with malicious intent, to compromise your system's security. The design and development of application layer software must be supported by a secure network, host and application configuration on the servers where the application software is to be deployed. To achieve this goal, a phisher first sets up a fake website that looks almost the same as the legitimate target website.

The URL of the fake website is then sent to a large number of users at random via e-mails or instant messages. Unsuspecting users who click on the link are directed to the fake website, where they are asked to input their personal information. Attackers or criminals are getting the personal information by lying about who they are and convince the user to share the account numbers, passwords and other information so that they can get all valuable information. This scam is called "phishing". In Phishing, the attacker who impersonate as legitimate authority and sends email, text, or pop-up messages that appear to come from a bank, a government agency, an online seller or another organization with which the user does business. The message asks the customer to click to a website or call a phone number to update the account information or claim a prize or benefit. Xiang *et al.* (2011) suggest something bad will happen if they don't respond quickly with the personal information. In reality, legitimate businesses should never use email, pop-ups, or text messages to ask for the personal information.

The monthly phishing attacks report of year 2010 of Anti Phishing Working Group is (APWG) shown in **Fig. 1**.

The recent country-wise report from Anti Phishing Working group is shown in **Fig. 2**.

1.1. Anti-Phishing Techniques

Anti-phishing technique can be considered said as an approach to counter the threats put forth by phishers. This accounts to a number of techniques followed which is categorized as follows.

1.2. List Based Approach

This is probably the most straightforward solution for anti-phishing. A white list contains URL's of known legitimate sites. Many current anti-phishing techniques rely on the combination of white list and blacklist. The representative blacklist/white list based systems include Phish Tank Site Checker, Google Safe Browsing, Fire Phish and CallingID Link Advisor. These anti-phishing solutions are usually deployed as toolbars or extensions of web browsers to remind the users whether they are browsing a safe website. Blacklist suffers from a window of vulnerability between the time a phishing site is launched and the site's addition to the blacklist as it requires frequent updating which is the case for white list also.

1.3. Heuristics Based Approach

This technique rates the phishing possibility of a given webpage using reputation scores either obtained from the anti-phishing community or computed from the given webpage. However the reliability of the reputation scoring is a great challenge.

1.4. Content Based Approach

This method is used to measure the similarity between two given web pages by calculating the similarity between the content elements (text, image, layout) contained in the web pages. Algorithms are used to compute visual similarity to detect the phishing web pages which have higher similarities to phishing targets. It requires finding the phishing target prior to the similarity comparison computation. It also combines TF-IDF retrieval algorithm to determine the likelihood that a given webpage is a phishing webpage. Words with highest TF-IDF weight on a given webpage can be used to classify the webpage as legitimate or not.

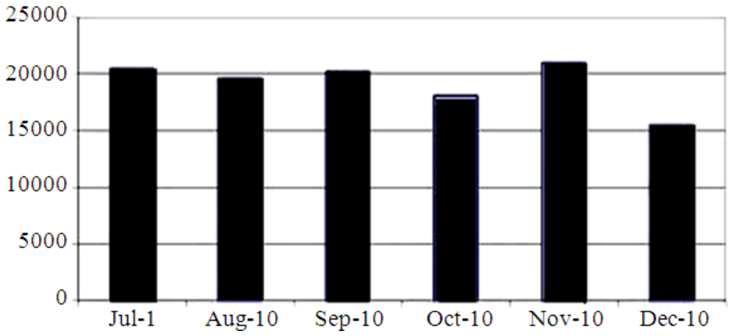


Fig. 1. Monthly Phishing attacks of year 2010

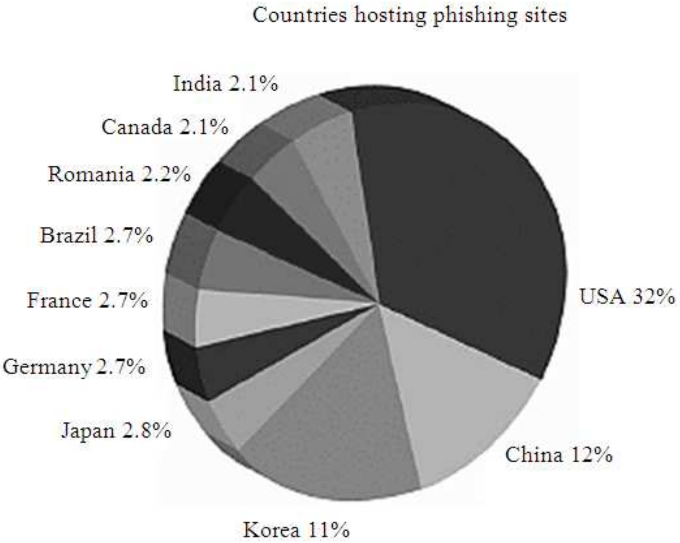


Fig. 2. Recent phishing report according to the country wise

1.5. Hybrid Approach

This approach usually combines any of the above mentioned techniques to classify a webpage as legitimate or not.

1.6. Literature Survey

1.6.1. Phish Net

Huang *et al*. (2011) states that, a blacklist scheme used to detect phishing attacks is discussed. Blacklist approach based techniques reliance on exact match with blacklisted entries makes it easy for phishers to evade. Phish Net exploits this observation using two components. In the first component, five heuristics to enumerate simple combinations of known phishing sites to discover new phishing URL's. The second component consists of an approximate matching algorithm that dissects a URL into multiple components that are matched individually against entries in the blacklist (Lakshmi and Vijaya, 2012).The first grows blacklists by generating new URL variations from the original ones but after vetting them through DNS and content matching. The second component consists of an approximate matching data structure that assigns a score to each URL based on piece-wise similarity with existing URLs. PhishNet suffers from low false positives and is remarkably effective at flagging new URLs that were not part of the original blacklist. Evaluation with real-time blacklist feeds discovered around 18,000 new phishing URLs from a set of 6,000 new blacklist entries and it leads to very few false positives, 3% and negatives, 5%.

1.7. Phishhark

Bargadiya *et al*. (2010) describes that, based on the characteristics of phishing URLs and web pages, heuristics to differentiate legitimate from illegitimate web pages are discussed. As blacklists are not the most effective in detecting phishing sites because of their short lifetime, heuristics appears as a privileged way at time 0. Based on the characteristics of phishing URLs and web pages, twenty heuristics parameters were defined and implemented in a toolbar called "Phishark". These heuristics were tested for its effectiveness and heuristics that differentiates legitimate web pages were considered (Vishwanath *et al*., 2011). It concludes which heuristics tests-for both URL and page content analysis-are decisive to identify a legitimate webpage from a phishing site.Tests were conducted on up to 1230 URLs comparing its performance to some of the most popular heuristics based techniques obtaining 98% true negative rate and 2% false negatives.

1.8. Web Communities

Aburrous *et al*. (2010) states that, a community on the web is defined as a set of sites that have more links (in either direction) to members of the community than to non-members. Members of such a community can be efficiently identified in a maximum flow minimum cut framework, where the source is composed of known members and the sink consists of well known non-members. A focused crawler that crawls to a fixed depth can approximate community membership by augmenting the graph induced by a crawl with links to a virtual sink node. The effectiveness of the approximation algorithm is demonstrated with several crawl results that identify hubs, authorities, web rings and other topologies that are useful but not easily categorized (Bose and Leung, 2008) field of application includes focused crawlers and search engines, automatic population of portal categories and improved filtering. A new type of web community that can be efficiently formed in a maximum flow framework and introduced a maximum flow based web crawler is defined that can approximate a community by directing a focused web crawler along link paths that are highly relevant.

1.9. Visual Similarity

Wenyin *et al*. (2010) states that, anti-phishing strategy using visual characteristics is proposed to identify potential phishing sites and to measure suspicious pages' similarity to actual sites registered with the system. The first of two sequential processes in the Site Watcher system runs on local email servers and monitors emails for keywords and suspicious URLs. The second process then compares the potential phishing pages against actual pages and assesses visual similarities between them in terms of key regions, page layouts and overall styles. The approach is designed to be part of an enterprise anti-phishing solution. It extracts the Web pages' features and measures the similarity to the true pages according to three metrics: block-level (detail), layout (global) and style (overall). If the visual similarity is higher than the corresponding threshold, the system issues a phishing report to the customer.

1.10. Content-Based Approach

He *et al.* (2011) describes that whether a webpage is a phishing page or a legitimate one based on its content, HTTP transaction and search engine results. This method used CANTINA, a content-based approach to detect phishing websites, which combines a Term Frequency-Inverse Document Frequency (TF-IDF) information retrieval algorithm with heuristics and determines the likelihood that a given webpage is a phishing page. CANTINA uses the five words with the highest TF-IDF weight on a given webpage as the lexical signature of that site and submits them to Google. If CANTINA finds the URL of the site in question within the top results, it classifies that as legitimate webpage or otherwise as phishing webpage. However, its efficacy heavily depends on the reliability of the search engine and whether the lexical signature selected is really a representative and as precise as a query for the search engine. **Table 1** shows the comparison of existing techniques.

Table 1. Comparison of existing techniques

Anti-phishing methods	Identification	Manual/ automatic	Target discovery
List approach	Yes	Manual	No
Heuristics approach	Yes	Manual	No
Similarity approach	Yes	Automatic	No

Table 2. Combination of Web community and existing techniques

Anti-phishing method	Identification	Manual/ automatic	Target discovery
Proposed method	Yes	Automatic	Yes

2. MATERIALS AND METHODS

2.1. System Architecture

We propose that users can be authenticated with the encrypted security code delivered via a reliable communication protocol on demand. The user database at the server side matches a user's name with its corresponding identity on another communication path. When a user wants to access the website, the server sends an encrypted security code to the user through the communication protocol. On receipt of the encrypted security code the user has to decrypt that code and can enter the login. The security code is encrypted with the private key and decrypted with the public key. Decryption process is done by the user. **Table 2** shows the combination of web community and existing techniques.

Finally, the method discovers the phishing target of the given webpage from within the parasitic community as the one which has sufficiently strong parasitic relationship with the given webpage. If we can find such phishing target, we can also determine the given webpage as a phishing webpage.

The admin process consisting of registration process for a web service involves the following steps. The user must choose one login name, fill in all the required information fields and provide at least one type of personal contact information (E-mail address or Phone number). The website should list all the services that it uses to deliver the security code so that the user can choose the preferred service. The use of a security question is not mandatory. It depends on the web site provider's policy or the user's wishes. However such questions make the authentication process more secure. The steps are:

- The validation page is sent to the customer. The page contains the name of the login used by the web site.

- If the customer's login name is new to the web site, the customer is asked for permission to add the login name to the websites contact list.
- After the login has been approved by both the web site and the customer, the website sends an account validation message to the user via the designated communication channel.

Next, the user starts the actual login process by browsing the login page, which contains an input field for the customer's login name and the CAPTCHA test. If the user's login name is not recognized by the website, it must be displayed in a page. If the user's account name is valid, the website checks the customer's registered account and sends an acknowledgement to that account. If the acknowledgement message is valid, the customer enters the assigned security code on the input page. On receipt of the security code, the website has to make sure that the customer submits the security code from exactly the same IP address as the customer requests to login. Hence, we propose architecture to prevent phishing attack as shown in **Fig. 3**.

Here the life span of the security code is limited. If the customer inputs an invalid security code more than n times or a delivered security code has not been used within m seconds, the website will invalidate current security code and stop the process. Then must be a very small number and m must be a very short time.

2.2. Security Analysis

Various factors affecting the value of security for the proposed solution are as follows:

2.3. Denial of Service Attack

A denial of service attack is a simple, but often extremely effective one that is difficult, if not impossible, to prevent. The goal of a denial of service attack is to deny access to particular services, effectively preventing your organization from operating. A denial of service could be launched against any part of your Internet connectivity and network infrastructure. In our proposed solution the website authenticates the customer by asking to input the security code already assigned by the website. The customer authenticates the website by first checking the sender of the acknowledgement message.

2.4. IP Spoofing

An attacker may fake their IP address so the receiver thinks it is sent from a location that it is not actually from. There are various forms and results to this attack. The attack may be directed to a specific computer addressed as though it is from that same computer. This may make the computer think that it is talking to itself. This may cause some operating systems such as Windows to crash or lock up. Our proposed solution restricts the locations that are able to launch IP-Spoofing attacks. If the attacker uses the same IP address as the user in the same local network concurrently it can be detected by the user. The lifetime of the security code is only a few second. So it is not possible for the attacker to login the protected website via the same IP address.

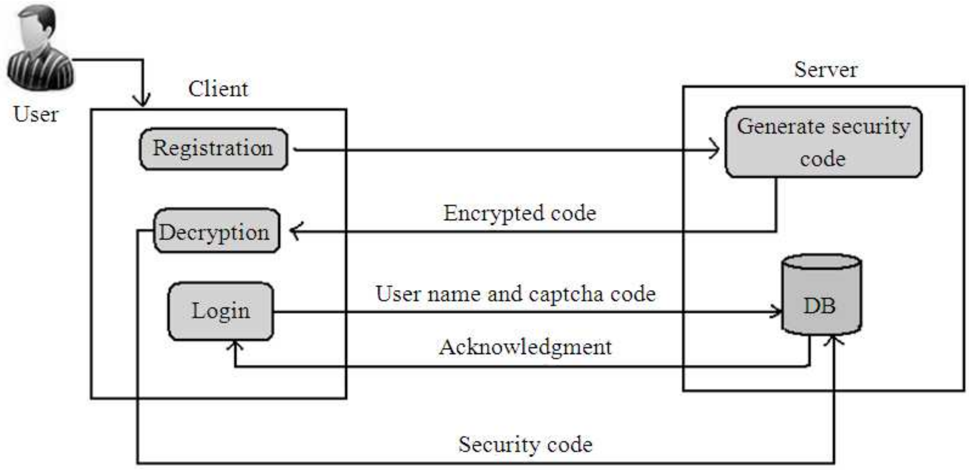


Fig. 3. Proposed system architecture diagram

2.5. Server Spoofing

A C2MYAZZ utility can be run on Windows 95 stations to request LANMAN authentication from the client. The attacker will run this utility while acting like the server while the user attempts to login. If the client is tricked into sending LANMAN authentication, the attacker can read their username and password from the network packets sent. In our proposed solution, customers do not require a preset password to login into a website; thus no passwords can be stolen.

2.6. Man in the Middle Attack

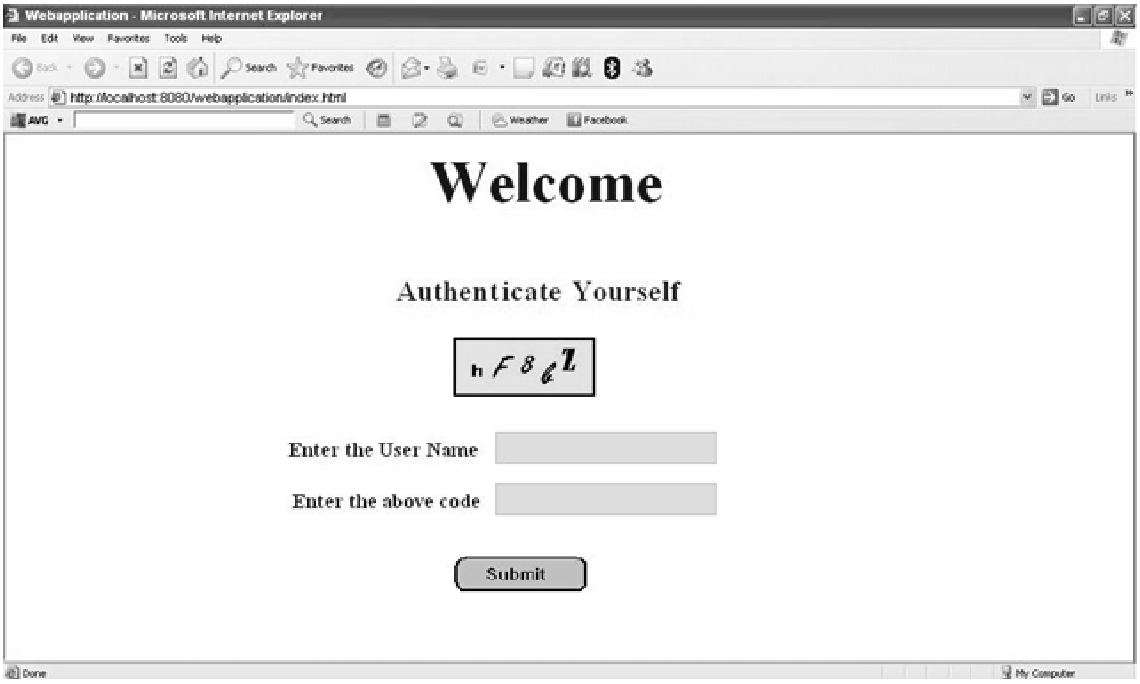
An attacker may watch a session open on a network. Once authentication is complete, they may attack the client computer to disable it and use IP spoofing to claim to be the client who was just authenticated and steal the session. This attack can be prevented if the two legitimate systems share a secret which is checked periodically during the session. In our proposed solution, suppose that the attacker can discover both the customer's web account name and the security code for the current session. Since the life span of the security code is very short (i.e., a20s) it would be of little use to the attacker.

3. RESULTS

If our system is to provide a realistic defense against phishing attacks, it must impose minimal overhead, since a solution that significantly slows to the web browsing experience will be unlikely to be adapted. **Figure 4** contains an input field for the user's login name and the CAPTCHA test. If the user's login name is not valid it will show an error message. If the user's name is valid, the website checks the user's registered account and sends an acknowledgement to that user.

Next, the customer enters the assigned security code on the input page in **Fig. 5**. On receipt of the security code, the website has to check whether the user submits a valid security code. If it is not valid, it will display the error message and the user can enter the wrong security code only n times. **Table 3** summarize the response time and explained that whether we will be allowed to use the website or not for some sample URLs.

The response time and result are tabulated in **Table 4**. By applying our tool, all the phishing sites except one legitimate site are prevented. The comparison between the previous approach and our approach is shown in **Fig. 6**.


Webapplication - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/webapplication/index.html
Go Links
AVG Search Weather Facebook
Submit
Done
My Computer

Welcome

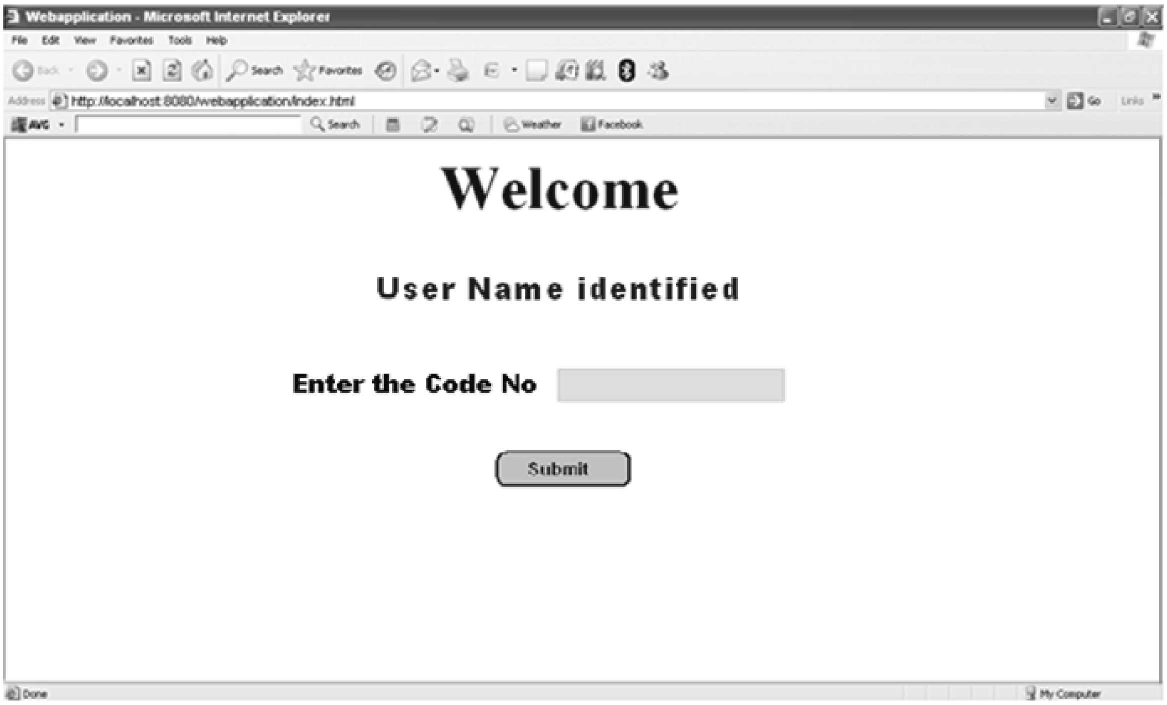
Authenticate Yourself

h F 8 & Z

Enter the User Name

Enter the above code

Fig. 4. Authentication


Webapplication - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/webapplication/index.html
Go Links
AVG Search Weather Facebook
Submit
Done
My Computer

Welcome

User Name identified

Enter the Code No

Fig. 5. Identification of the user name

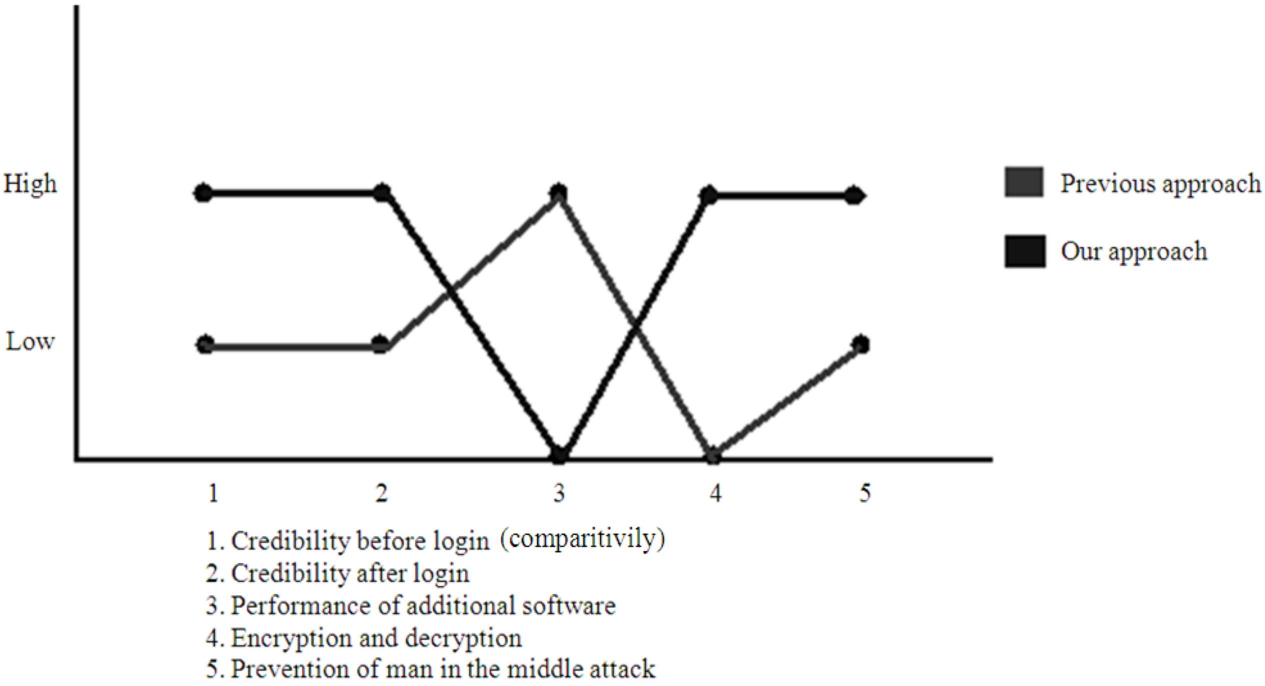


Fig. 6. Comparison between Previous approach and our approach

Table 3. Summarizing the performance overhead imposed by our scheme. The averages are calculated over 30 trials and the secret code is created using 1024-bit RSA key pairs

URL	Time taken for response (ms)	Website type	Status
www.greatindia.net	3431	Legitimate site	Allowed
88.193.226.99	517	Phishing IP in hyperlink	Not allowed
www.paypai.com	4210	Phishing Site	Allowed
convert.money.net	3521	Having phishing hyperlink	Allowed
google.com.net	2136	Phishing Site	Not Allowed

Table 4. Validation through the security code

	Time (sec)	[min, max]
Account creation	0.3	[0.2,0.6]
Secret code creation	61.0	[25.5,150.1]
Assigning Communication channel	1.3	[1.2,1.6]

4. DISCUSSION

Phishing is a big problem when measured by the standards of the volume of phishing email received by users and by the number of reported phishing sites. APWG reports more than 35000 new phishing sites per month. For example, if each received an average of 100 victims and if 400 million users are using the websites, this would imply that $100*3.7e^5 * 12/400e^5 \approx$ 7.7% of users were being published annually.

When comparing the consequences of phishing, the increase in time in milliseconds is negligible. But this fundamental checking doesn't prevent complete phishing sites. Here, we have applied our tool to prevent phishing sites which are powerful than the earlier techniques.

5. CONCLUSION

In this study our main contribution includes two aspects: firstly, a new problem of discovering the phishing target of a given phishing website is proposed, which is more significant than only identifying a given suspicious website as phishing or not in previous work. Secondly, an application of the security code phishing detection is explored for this new problem. Though there are a number of methods for detecting phishing behavior and protecting users from attacks, it is still not possible to detect all phishing sites. We anticipate that our approach would be deployed for websites requiring a high level of security and that it would ultimately help in remaining customer confidence in using web-based commerce. For future work, we intend to analyze only the URL, which is given as input to the web browser by a user. The independent analysis of the URL and hyper links gives a greater performance to protect phishing. If URL, hyperlink and the content of the websites are properly analyzed, we can protect phishing in a better way.

6. REFERENCES

Aburrous, M., M.A. Hossain, K. Dahal and F. Thabatah, 2010. Intelligent phishing detection system for e-banking using fuzzy data mining. J. Exp. Syst. Appli., 37: 7913-7921. DOI: 10.1016/j.eswa.2010.04.044

Bargadiya, M., V. Chaudhari, M.I. Khan and B. Verma, 2010. Anti-phishing design using mutual authentication approach. Int. J. Comput. Applic. Inform. Technol., 1: 175-178.

Bose, I. and A.C.M. Leung, 2008. Assessing anti-phishing preparedness: A study of online banks in Hong Kong. Decis. Support Syst., 45: 897-912. DOI: 10.1016/j.dss.2008.03.001

He, M., S.J. Horng, P. Fan, M.K. Khan and R.S Run *et al*., 2011. An efficient phishing webpage detector. Exp. Syst. Applic., 38: 12018-12027. DOI: 10.1016/j.eswa.2011.01.046

Huang, C.Y., S.P. Ma and K.T. Chen, 2011. Using one-time passwords to prevent password phishing attacks. J. Netw. Comput. Applic., 34: 1292-1301. DOI: 10.1016/j.jnca.2011.02.004

Lakshmi, V.S. and M.S. Vijaya, 2012. Efficient prediction of phishing websites using supervised learning algorithms. Proc. Eng., 30: 798-805. DOI: 10.1016/j.proeng.2012.01.930

Vishwanath, A., T. Herath, R. Chen, J. Wang and H.R. Rao, 2011. Why do people get phished? Testing individual differences in phishing vulnerability within an integrated, information processing model. Decis. Support Syst., 51: 576-586. DOI: 10.1016/j.dss.2011.03.002

Wardman, B., G. Shukla and G. Warner, 2009. Identifying vulnerable websites by analysis of common strings in phishing URLs. Proceedings of the eCrime Researchers Summit, Sept. 20-Oct. 21, IEEE Xplore Press, Tacoma, WA., pp: 1-13. DOI: 10.1109/ECRIME.2009.5342610

Wenyin, L., N. Fang, X. Quan, B. Qiu and G. Liu, 2010. Discovering phishing target based on semantic link network. Future Gener. Comput. Syst., 26: 381-388. DOI: 10.1016/j.future.2009.07.012

Xiang, G., J. Hong, C.P. Rose and L. Cranor, 2011. CANTINA+: A feature-rich machine learning framework for detecting phishing web sites. ACM Trans. Inform. Syst. Sec. DOI: 10.1145/2019599.2019606