Pegem Journal of Education and Instruction, Vol. 13, No. 4 , 2023 (pp. 437-446)

RESEARCH ARTICLE WWW.PEGEGOG.NET

Phish Catcher: Client side defense against web spoofing attacks using machine learning

[1]Dr. D.Madhavi, [2]G. Sreeja, [3]T. Chaitramaye, [4]V. Manasa

[1] Associate Professor, Department of Computer Science and Engineering, Sridevi Women's Engineering College, Hyderabad, India.

Email: karrimadhavi16@gmail.com

[2,3,4,] B.Tech Student, Department of Computer Science and Engineering, Sridevi Women's Engineering College, Hyderabad, India.

ABSTRACT—

Cybersecurity faces a huge issue in protecting users' personal information, including passwords and PIN codes, from unauthorised access. False login pages seeking sensitive information reach billions of visitors every day. A user may be tricked into visiting a malicious website by several means, including phishing emails, enticing adverts, click jacking, malware, SQL injection, session hijacking, man-in-the-middle attacks, denial of service, and cross-site scripting. Phishing, also known as online spoofing, is a kind of electronic trickery in which the perpetrator creates a fake but seemingly official website in order to steal sensitive information from unsuspecting consumers. Researchers have suggested many security measures to combat these vulnerabilities, however they are plagued by problems with latency and precision. We suggest and build a client-side defensive mechanism that uses machine learning to identify phishing attempts and faked websites in order to circumvent these problems. A Google Chrome plugin called Phish Catcher was created as a proof of concept. It uses our machine learning algorithm to determine whether a URL is suspicious or trustworthy. The system uses a random forest classifier to determine whether a login page is faked based on four distinct kinds of web properties. Several studies were conducted on actual web apps to evaluate the extension's precision and accuracy. Experiments conducted on 400 correctly identified phished URLs and 400 correctly identified authentic URLs reveal an impressive accuracy rate of 98.5% and precision of 98.5%. In addition, we conducted studies using forty phished URLs to assess our tool's latency. On average, Phish Catcher only took 62.5millisecondstorespond

INTRODUCTION

A French-language email was sent to members and users of France's National Institute for Research in Digital Science and Technology (Inria) in October 2022,1 requesting that they verify their webmail

Corresponding Author e-mail: karrimadhavi16@gmail.com

How to cite this article: 1Dr. D.Madhavi, 2G. Sreeja, 3T. Chaitramaye, 4V. Manasa. Phish Catcher: Client side defense against web spoofing attacks using machine learning.Pegem Journal of Education and Instruction, Vol. 13, No. 4, 2023, 437-446.
Source of support: Nil **Conflicts of Interest:** None.
DOI: 10.48047/pegegog.13.04.52

Received: 12.10.2023

Accepted: 22.11.2023 **Published:** 24.12.2023

account via the following URL: https://www.educationonline.nl/Cliquez.ici.cas.inria.fr.cas.login/login.html. It seems that Seifedine Kadry was the associate editor responsible for organising the review and publishing approval of this submission; nevertheless, clicking on this URL redirects to a bogus but quite legitimate website. 1On October 10, 2022, the individuals who utilise the Coq-club Inria website (https://www.inria.fr/en) were notified by email of a phishing attempt. Since https://cas.inria.fr/cas/login?service= looks so much like the actual Inria login page, it must be a fake.By tricking users into entering their Inria credentials on a phoney website, an attacker may get access to the actual Inria login page. The Inria and all members and users who have registered with them are the targets of this phishing attempt. Figures 1 show the authentic and fictitious Inria login pages. Users are easy prey for this phishing scam since the two websites are very identical. Part V of the VOLUME 11, 2023 report details our testing of the Phish Catcher tool against this and a small number of additional assaults. Creative Commons Attribution-Noncommercial-No Derivatives 4.0 is the licence that this work is licenced under. Take a look at https://creativecommons.org/licenses/by-ncnd/4.0/61249 for more details. With Phish Catcher, Client-Side Defence Against Web Spoofing Attacks (M. Ahmed et al., 2017).

Attack on Inria by a phisher (Figure 1). There has been a meteoric rise in the online realm, including e-commerce, e-banking, ehealth, and e-governance, thanks to the remarkable development of contemporary technology. Social media platforms like Facebook and Twitter have attracted an ever-increasing user base due to the significant part they play in the current era's globalisation. Users may personalise their experience by creating an account on several websites. Users are required to establish a unique account in order to use the webbased specialised services. For this reason, visitors often encounter login web pages, where they are required to create an account by generating and entering an identifier (e.g., password) and a secret (e.g., username). Web requests are sent to the user the next time they need to access a remote resource or service, and the user is then sent a login form to provide their identify and secret. Currently, there is a significant threat to users' privacy from identity theft and the disclosure of sensitive information. As shown in Figure 2, the first step in a phishing assault is to receive an email that contains a link to a rogue website. There may be persuasive or enticing language in the email that makes the reader want to click and follow the link. The naive person accesses the page, thinking it is the real, trustworthy website where they have an account. The attacker receives the victim's confidential information (username, password, etc.) after the victim touches the submit or login button. Once the victim submits their secret credentials to the phishing website, the perpetrator of the assault has access to the real website. Since the introduction of web spoofing or phishing assaults, there has been a dramatic rise in online fraud, scams, and identity theft. One kind of cybercrime is web spoofing, sometimes known as phishing, in which an attacker attempts to trick a victim into divulging sensitive information. In order to compromise online systems, attackers have used a wide variety of phishing and web spoofing tactics. Attackers are increasingly using webspoofing to steal critical information pertaining to national security, but its original purpose was to facilitate identity theft. A common phishing attempt (FIGURE 2). confidentiality, proprietary information, and company trade secrets. Various forms of modern phishing, such as QR code phishing, mobile spoofing applications, spear phishing, etc., have already reached a new evolutionary level. Security measures like firewalls, digital certificates, encryption software, and twofactor authentication may be evaded by

these types of assaults and fraud techniques. Many businesses have begun using twofactor authentication systems as a precaution against financial fraud and identity theft. The sophisticated fraud methods have unfortunately rendered all of these systems susceptible. Issue 11, Volume 6, 2023 Defending Clients From Web Spoofing Attacks: PhishCatcher by M. Ahmed and colleagues Attackers often mimic the look of legitimate websites on their spoof sites by include logos, either by keeping copies of the logos or by providing links to them. The hacker could use logos and even copy and paste HTML code from the legitimate site, making little adjustments here and there. Phishers utilise email, trojan horses, key loggers, and man-in-the-middle proxies as their attack vectors to deceive users. Online banking, third-party payment systems, and e-commerce sites are the most popular targets of cybercriminals. The cryptographic security protocols SSL/TLS do not provide a foolproof defence against phishing attempts since they only target non-cryptographic components. These protocols need supplementary safeguards to be reliable against spoofing attacks. These controls may be implemented on the client side, the server side, or even both. The majority of developers choose to disregard the serverside solutions since they need adjustments to the websites. In contrast, user protection is provided by client side solutions even in the absence of server assistance. While there may be server-side solutions that may detect faked sites, this article will concentrate on client-side methods. Third party certification, passwords, or URLs form the basis of the majority of anti-spoofing technologies. There are two main types of anti-spoofing tools: stateful and stateless. Blacklists and heuristics are two automated phishing detection mechanisms that may be used to categorise them. Although they are able to identify over 90% of phishing sites, tools that depend on black/white lists fail to detect zero-day assaults. These tools also produce nearly no false positives, often known as accuracy. In addition, black-listing approaches have a number of limitations, such as being susceptible to spam URL manipulation and unable to keep up with newly emerging attack vectors. The heuristic-based approaches have shown promising results in capturing phishing sites that are not part of the blacklists. With just one percent of false positives, heuristic (content) based tools like Spoof Catch and CANTINA can detect 90% of phishing sites. Spoof Catch has a latency that is on the order of seconds and becomes worse as time goes

on. Despite their high accuracy, stateful anti-phishing algorithms rapidly deplete local storage and experience performance degradation over time. As the user continues to explore the web, Spoof Catch stores more and more login page photos in its local storage, increasing the visual similarity comparison. The time it takes to compare a received login page picture to all of the login images in storage is also increased. Building on previous work in this area, we have created an ML-based stateless anti-phish tool. To help prevent future scams, several well-known researchers have suggested machine learning approaches to identify bad URLs. In ML methods, a large number of URLs are considered training data. It is suggested that, using the statistical attributes derived from the training sets, one may determine whether the URL being accessed is a fraud or not. When it comes to URL identification with ML, training data is king. A mathematical model is the end result of processing training data. Gathering characteristics from training data should be your first priority, since basic strings may not be enough to forecast the test URL's state. Finally, a real model is generated from the training data using the projected model. Many researchers employ machine learning methods like Logistic Regression (LR), Naïve Bayes, and Support Vector Machines (SVM) for this purpose; nevertheless, these algorithms have a number of vulnerabilities. To counteract online spoofing attacks, we present and build Phish Catcher, a stateless client-side programme. An add-on for Google Chrome called Phish Catcher uses machine learning and the random forest method to determine whether a login page is real or fake. Impressive results were obtained when we tested the efficacy and precision of the Phish Catcher on actual web apps.

RELATED WORK

A client-side utility that protects against phishing attacks: Spoof Catch

Most anti-phishing methods in the literature either fail to detect phishing attempts altogether or rely on overly complicated sets of criteria to do so, leaving users vulnerable to online spoofing assaults. We argue in this article that the user may avoid falling victim to phishing assaults by paying attention to the page's aesthetics as a whole. In order to prove our point, we develop an extension for the browser called Spoof Catch that uses visual similarities across websites to provide a client-side security mechanism. The addon makes use of four different similarity

algorithms to compare the appearance of legitimate and phished web pages. In order to test the solution, we ran comprehensive and large-scale tests, and the results show that spoof Catch can detect all phishing attempts with a tolerable amount of cost.

A system for monitoring and identifying phishing attempts

One kind of identity theft, known as "phishing," uses complex attack vectors and social engineering tactics to steal sensitive financial information from people who aren't paying attention. Phishers often use URLs that lead to malicious websites in an attempt to trick their victims into clicking on them. The primary objective of this research is to analyse the structure of URLs used in different types of phishing attempts. In many cases, we can identify a phishing URL even without knowing the specifics of the linked page's content. To help you identify a malicious URL, we've outlined a number of telltale signs. Make advantage of these attributes to create an accurate and efficient logistic regression filter. In order to determine the extent to which phishing is prevalent on the Internet nowadays, we use this filter to conduct comprehensive measurements on a number of million URLs.

Effective defence against web spoofing and phishing

The proliferation of phishing and web spoofing has become a significant problem on the Internet. Since the attacks primarily aim at non-cryptographic components like the user or the user-browser interface, they pose a significant security risk. Because of this, extra safeguards are required in addition to cryptographic security systems like SSL/TLS, which do not provide a comprehensive answer to the threats. The purpose of this article is to provide a concise overview of the literature on such techniques and their efficacy in preventing (massive) phishing and Web spoofing assaults.

METHODOLOGY

This module allows users to submit datasets in order to train algorithms.

Dataset preparation: the dataset is divided into two sections, one for training and one for testing, using this module.

Execute Existing SVM Algorithm: By using this module for SVM training, we achieved an accuracy rate of 96%. Additionally, we can see other metrics such as recall, precision, and

FSCORE. In the confusion matrix graph, the x-axis shows the predicted labels and the y-axis shows the true labels. Each yellow box displays the count of accurate predictions.

Get the Random Forest Algorithm Running: I trained the Random Forest algorithm with this module, and it achieved an accuracy of 98%.

Execute the XGBOOST Extension Algorithm: We trained the XGBOOST algorithm with this module and achieved a 99% success rate.

In this module, we can see a comparison graph showing the performance of all the algorithms. The x-axis shows the names of the algorithms, while the y-axis shows various metrics, such as accuracy. Among all the algorithms, XGBOOST achieved the highest accuracy.

Upload Testing data: we are using this module to construct code that reads TEST URLS from the testing data. Then, we are predicting if the weather URL is saved or not using the extension XGBOOST.

Deceitful Emails

RESULT AND DISCUSSION

Following the execution of the code in the preceding screen, the following result will be displayed: "Reading test URLs from test data and predicting whether the URL is saved or phishing."

In above screen before arrow =➊ symbol we can see TEST URL and after =➊ arrow symbol we can see predicted output as ‘SAFE or PHISHING’

CONCLUSION

These days, a lot of our information comes from internet sources, such news articles, emails, reviews, posts, and more. By using false phishing URLs or spoofing websites, attackers might entice regular users with

appealing promises of winning the jackpot via this online content access. If a person visits a spoofing website or clicks on one of these URLs, the attackers will ask for their login credentials. Once they have them, they may access the victim's banking or other financial accounts and steal their money or other sensitive information.

The author of this study uses the Random Forest method to identify phishing URLs since, despite the introduction of several machine learning and signature-based techniques, their detection rates are inaccurate. To improve the accuracy of predictions, the Random Forest algorithm has built-in assistance for optimising and selecting characteristics. In order to choose just the most optimised features, random forest applies a collection of trees to the dataset, filters out any extraneous data, and then makes its final selection. The author has included a wealth of additional information that may be found in the basic article. The PHISHTANK dataset, which includes thousands of legitimate and malicious URLs, was used to train the proposed algorithm. This dataset allows us to determine whether a URL is safe or malicious. The author has created a Chrome plugin that, in addition to training, analyses every URLs a user visits and notifies them of those that are either safe or phishing. While comparing the proposed Random Forest method to the current SVM algorithm.

REFERENCES

[1] W. Khan, A. Ahmad, A. Qamar, M. Kamran, and M. Altaf, ‘‘spoof Catch: A client-side protection tool against phishing attacks,’’ IT Prof., vol. 23, no. 2, pp. 65–74, Mar. 2021.

[2] B. Schneier, ‘‘Two-factor authentication: Too little, too late,’’ Commun. ACM, vol. 48, no. 4, p. 136, Apr. 2005.

[3] S. Garera, N. Provos, M. Chew, and A. D. Rubin, ‘‘A framework for detection and measurement of phishing attacks,’’ in Proc. ACM Workshop Recurring malcode, Nov. 2007, pp. 1–8.

[4] R. Oppliger and S. Gajek, ‘‘Effective protection against phishing and web spoofing,’’ in Proc. IFIP Int. Conf.

Commun. Multimedia Secur. Cham, Switzerland: Springer, 2005, pp. 32–41.

[5] T. Pietraszek and C. V. Berghe, ‘‘Defending against injection attacks through context-sensitive string evaluation,’’ in Proc. Int. Workshop

Recent Adv. Intrusion Detection. Cham, Switzerland:
Springer, 2005, pp. 124–145.

[6] M. Johns, B. Braun, M. Schrank, and J. Posegga, ‘‘Reliable protection against session fixation attacks,’’ in Proc. ACM
Symp. Appl. Comput., 2011, pp. 1531–
1537.

[7] M. Bugliesi, S. Calzavara, R. Focardi, and W. Khan, ‘‘Automatic and robust clientside protection for cookie-based sessions,’’ in Proc. Int. Symp. Eng. Secure Softw. Syst. Cham, Switzerland: Springer, 2014, pp.
161–178.

[8] A. Herzberg and A. Gbara, ‘‘Protecting (even naıve) web users from spoofing and phishing attacks,’’ Cryptol. ePrint Arch., Dept. Comput. Sci. Eng., Univ. Connecticut, Storrs, CT, USA, Tech. Rep. 2004/155, 2004.

[9] N. Chou, R. Ledesma, Y. Teraguchi, and J. Mitchell, ‘‘Client-side defense against web-based identity theft,’’ in Proc. NDSS, 2004, 1–16.

[10] B. Hämmerli and R. Sommer,
Detection of Intrusions and Malware, and Vulnerability Assessment: 4th International Conference, DIMVA 2007 Lucerne, Switzerland, July 12-13, 2007 Proceedings, vol. 4579. Cham, Switzerland: Springer, 2007.

[11] C. Yue and H. Wang, ‘‘BogusBiter: A transparent protection against phishing attacks,’’ ACM Trans. Internet Technol., vol. 10, no. 2, pp. 1–31, May 2010. [12] W. Chu, B. B. Zhu, F. Xue, X. Guan, and Z. Cai, ‘‘Protect sensitive sites from phishing attacks using features extractable from inaccessible phishing URLs,’’ in Proc. IEEE Int. Conf. Commun. (ICC), Jun. 2013, pp. 1990–1994.

[13] Y. Zhang, J. I. Hong, and L. F. Cranor, ‘‘Cantina: A content-based approach to detecting phishing web sites,’’ in Proc. 16th Int. Conf. World Wide Web, May 2007, pp. 639–648. VOLUME 11, 2023 61261 M. Ahmed et al.: PhishCatcher: Client-Side Defense Against Web Spoofing Attacks

[14] D. Miyamoto, H. Hazeyama, and Y. Kadobayashi, ‘‘An evaluation of machine learning-based methods for detection of phishing sites,’’ in Proc. Int. Conf. Neural Inf. Process. Cham, Switzerland: Springer, 2008, pp. 539–546.

[15] E. Medvet, E. Kirda, and C. Kruegel, ‘‘Visual-similarity-based phishing detection,’’ in Proc. 4th Int. Conf. Secur. privacy Commun. Netowrks, Sep. 2008, pp. 1–6.

[16] W. Zhang, H. Lu, B. Xu, and H. Yang, ‘‘Web phishing detection based on page spatial layout similarity,’’ Informatica, vol. 37, no. 3, pp. 1–14, 2013.

[17] J. Ni, Y. Cai, G. Tang, and Y. Xie, ‘‘Collaborative filtering recommendation algorithm based on TF-IDF and user characteristics,’’ Appl. Sci., vol. 11, no. 20, p. 9554, Oct. 2021.

[18] W. Liu, X. Deng, G. Huang, and A. Y. Fu, ‘‘An antiphishing strategy based on visual similarity assessment,’’ IEEE Internet Comput., vol. 10, no. 2, pp. 58–65, Mar. 2006.

[19] A. Rusu and V. Govindaraju, ‘‘Visual CAPTCHA with handwritten image analysis,’’ in Proc. Int. Workshop Human Interact. Proofs. Berlin, Germany: Springer, 2005, pp. 42–52.

[20] P. Yang, G. Zhao, and P. Zeng, ‘‘Phishing website detection based on multidimensional features driven by deep learning,’’ IEEE Access, vol. 7, pp. 15196– 15209, 2019.