International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

AN INTELLIGENT CLIENT SIDE -SHEILD AGAINT PHISHING AND SPOOFING ATTACKS

Bontha Varsha[1]; Katikam Mahesh[2]; Dr .k. Prasada Rao[3]

[1]PG Student: Department of Master of Computer Applications, Tirumala Institute of Technology and Sciences, Satuluru -522601, Narasaraopet, Andhra Pradesh, India

[2]Assistant Professor : Department of Master of Computer Applications, Tirumala Institute of Technology and Sciences, Satuluru-522601, Narasaraopet, Andhra Pradesh, India

[3]Professor: Department of Master of Computer Applications, Tirumala Institute of Technology and Sciences, Satuluru-522601, Narasaraopet, Andhra Pradesh, India

Abstract: The safeguarding of personal identification numbers and passwords is a major challenge for cyber security. Fraudulent login pages requesting personal information trick billions of users every day. Many malicious techniques exist to trick people into visiting malicious websites, including phishing emails, clickjacking, malware, SQL injection, session hijacking, man-in-the-middle attacks, denial of service, and cross-site scripting. The perpetrator makes a fake but convincingly similar website in order to trick victims into giving over their passwords. Numerous security solutions have been proposed by researchers to prevent these vulnerabilities; however, these techniques are both inefficient and prone to error. We present and execute a client-side defense mechanism that uses machine learning to identify phishing attempts and identify bogus web pages. As a proof of concept, our machine learning system is used by the Google Chrome plugin PhishCatcher to categorize URLs as trustworthy or suspicious. The random forest classifier checks a login page for fakery after obtaining four web features. The accuracy and correctness of the extension were tested on several real- life web apps. The results demonstrated a precision and accuracy rate of 98.5% when tested on 400 real URLs and 400 classified phished ones. Using forty phishing URLs, we measured the latency of our tool. We enhanced Random Forest with XGBOOST, a method that screens datasets using forest trees or groups of estimators to optimize features more effectively and provide high accuracy.

Keywords: Cybersecurity, Phishing Detection, Machine Learning,Random Forest, XGBoost, Fake Website Detection,URL Classification, Google Chorme Extension ,PhishingCatcher, Web Security, Client- side Defense, Login Page Protection , Precision and Accuracy, Real-time Detection, Feature Extraction.

I. INTRODUCTION

In October 2022, Inria faced a phishing attack, with deceptive French emails prompting users to verify webmail accounts via a convincing fake login page. Users unknowingly entered Inria credentials on the fake page, posing a serious threat. Attackers could exploit harvested data for malicious activities. A Google Chrome extension, PhishCatcher uses random forest algorithm to discern legitimate from spoofed login pages, enhancing user protection against phishing attacks. However, despite their effectiveness, existing anti-phishing solutions have

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

notable drawbacks. Many systems rely on static rule-based approaches, making them less adaptable to evolving phishing techniques. Additionally, machine learning-based models depend on historical data, which may result in delayed responses to novel phishing patterns. Overly strict identification criteria can also generate false positives, frustrating users and reducing trust in the system. Furthermore, the lack of user-friendly interfaces and ineffective warning mechanisms can hinder the overall effectiveness of anti-phishing measures.

User cybersecurity is enhanced by PhishCatcher, a client-side defense that relies on machine learning. To identify and prevent online phishing attacks, Random Forest and the XGBOOST algorithm are utilized. To ensure a reliable and effective solution, the suggested Google Chrome extension is tested on real web apps for latency, accuracy, and precision.

The security of user information and personal details is increasingly at risk due to phishing assaults. When it comes to identifying fraudulent websites, current security measures have issues with latency and accuracy. XGBOOST and Random Forest are two machine learning techniques that PhishCatcher uses to combat this. To safeguard users from cyber threats like phishing and other fraudulent actions, such as fake login pages, a robust defense mechanism is required.

To operate a program smoothly, certain hardware and software must be in place, and this is known as the software requirements. These prerequisites typically aren't part of the application installation package and need to be installed separately.

Platform - A platform is a foundational piece of hardware or software that allows software to execute on a computer. Computer hardware, software, languages, and runtime libraries are all examples of common platforms. Operating systems are among the most fundamental components of any computer. While it's possible for software to incompatibly with later versions of the same OS, backward compatibility is typically preserved. Although it is not necessarily the case, most Windows XP software will not run on Windows 98.

Linux versions employing Kernel v2.2 or v2.4 are rarely compatible with software developed using later Linux Kernel v2.6 capabilities.

APIs and drivers – Newer drivers or an API are required by software that makes advantage of high-end display devices. Media-related tasks, especially game development, are handled by Microsoft's DirectX suite of application programming interfaces.

Web browser - The default browser is utilized by the majority of web apps and software that is dependent on Internet technologies. Microsoft Internet Explorer is widely used by Windows users and makes use of ActiveX components, which are not without their risks.

II. EXISITING SYSTEM

Email, online posts and reviews, online news, and other forms of online data are utilized. Phishing URLs or fake websites claiming to have won jackpots can take use of this content access to fool unsuspecting users. A pop-up window prompts users to provide their login information whenever they visit these URLs or counterfeit

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

websites. Criminals get unauthorized access to financial or banking websites, steal the user's funds, or access other sensitive information by using this information.

III. EXISTING STSTEM

Email, online posts and reviews, online news, and other forms of online data are utilized. Phishing URLs or fake websites claiming to have won jackpots can take use of this content access to fool unsuspecting users. A pop-up window prompts users to provide their login information whenever they visit these URLs or counterfeit websites. Criminals get unauthorized access to financial or banking websites, steal the user's funds, or access other sensitive information by using this information.

IV. PROPOSED SYSTEM

Since the detection rates of many machine learning and signature-based approaches to avoiding such URLs are unreliable, this study employs the Random Forest method to identify phishing URLs. If you want more accurate predictions, you should use the Random Forest technique, which helps with feature selection and optimization. For optimal feature selection and noise filtering, random forest employs a network of trees. The author elaborates further in the main paper. In order to train our algorithm to predict if a URL is safe or phishing, we used PHISHTANK, a dataset that contains thousand of regular and malicious URLs. Similar to Random Forest, we have a novel algorithm called XGBOOST. Comparatively more accurate and feature-optimized than Random Forest, it employs estimators or forest trees to filter datasets. As an add-on, XGBOOST was included.

V. CONCLUSION

Online apps provide high-quality services in areas such as e-commerce, social connectivity, virtual education, virtual health services, online banking, digital marketing, and multiplayer gaming, which is why users rely on them. In order to gain access to protected content on the internet, users must first register for an account. Advancements in online spoofing have made users' security and privacy more vulnerable. There are a lot of approaches to stop online spoofing, both academic and commercial, but they all have their limitations. With the use of supervised machine learning, we developed PhishCatcher, an intuitive browser extension that can identify phishing attempts. In contrast to previous approaches, our system does classification within the browser itself. In order to fix issues with web applications, it reduces latency and increases tool efficiency. Our plug-in is designed with a user-friendly interface to ensure clarity. The phishing features of a URL are shown through a drop-down menu that displays when a user enters it. There are 30 features in the set, and they are organized into four categories based on decision trees. The random forest classifier can distinguish between legitimate and fraudulent login pages by aggregating decision trees. There are 400 legitimate and 400 malicious URLs in the evaluation and testing collection. The confusion matrix of TPM, TN, FPM, and FN serves as the foundation for testing and evaluation. Using a recall, accuracy, and precision of 98.5%, our plug-in produced remarkable categorization results. We measured an average latency of 62.5 milliseconds when

we ran the plug- in over forty phished URLs. Although there are thirty features in the set, performance might be improved with the addition of more automated functions. By training on larger datasets, discriminative classifiers like SVM may detect fake or real URLs. Both performance analysis and assessment tools can benefit from the development of new technology.

VI. REFERNECE

[1] 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.

https://doi.org/10.3390/app11209554.

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

https://doi.org/10.1109/MITP.2021.3052931.

[3] M. Bugliesi, S. Calzavara, R. Focardi, and W. Khan, ‘‘Automatic and robust client-side protection for cookie-based sessions,’’ in Proc. Int. Symp. Eng. Secure Softw. Syst. Cham, Switzerland: Springer, 2014, pp. 161–178. https://doi.org/10.1007/978-3-319-04897-0_11.

[4] 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. https://doi.org/10.1109/ICC.2013.6654797.

[5] 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.

https://www.informatica.si/index.php/informatica/article/view/452.

[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. https://doi.org/10.1145/1982185.1982511.

[7] 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. https://doi.org/10.1145/1754393.1754395.

[8] 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. 539–546C. Yue and H. Wang, ‘‘BogusBiter: A transparent protection against phishing attacks,’’ ACM Trans. Internet Technol., vol. 10, no. 2, pp. 1–31, May 2010.

https://doi.org/10.5555/1813488.1813559.

[9] 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.

https://doi.org/10.1145/1460877.1460905.

[10] 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.

https://doi.org/10.1145/1314389.1314391.

[11] 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. https://doi.org/10.1007/978-3-540-73614-1.

[12] 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.

https://doi.org/10.1145/1242572.1242659.

[14] 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. https://doi.org/10.1007/978-3-540-73614-1

[15] B. Schneier, ‘‘Two-factor authentication: Too little, too late,’’ Commun. ACM, vol. 48, no. 4, p. 136, Apr. 2005. https://doi.org/10.1007/978-3-540-73614-1.

[16] 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.

https://doi.org/10.1007/11552055_4.

[17] 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. https://doi.org/10.1007/11663812_7.