iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

ADVANCED DETECTION OF FRAUDULENT AND MALICIOUS URLS BASED ON MACHINE LEARNING

Mrs. Laxmi Hugar,
Assistant Professor, Department of Information Technology, VJIT college, Hyderabad, India
Hyderabad, India

Kota Bhumika,
Sarvigari Greeshma,
Bommaragoni Vasavi,
Mothukuri Sai Ganesh.
Students, Department of Information Technology, VJIT college, Hyderabad, India

laxmihugar1234@gmail.com, kotabhumika55@gmail.com, sgreeshma2003@gmail.com, vasavibommaragoni@gmail.com, saiganeshmothukuri@gmail.com.

ABSTRACT
With the escalating risk of cyber fraud driven by sophisticated techniques like social engineering and phishing, the detection of malicious Uniform Resource Locators (URLs) has become imperative. This paper introduces a novel approach to detecting cyber fraud, specifically focusing on malicious URLs and text by leveraging machine learning techniques and big data analytics. Traditional methods such as signature-based detection suffer limitations in identifying new threats, prompting the need for more advanced methodologies. The proposed system utilizes a combination of machine learning algorithms, namely Support Vector Machine (SVM), Naive Bayes, Logistic Regression, Decision Tree Classifier, SGD Classifier, to classify URLs and text messages based on their behaviors and attributes. Unlike existing systems, this approach incorporates both static and dynamic characteristics of URLs, leading to a more comprehensive detection mechanism. The novelty lies in the introduction of new URL features specifically tailored for malicious URL and text detection. Advantages of the proposed system include its ability to harness the potential of these new features, enhancing the accuracy and efficiency of malicious URL and text detection. While These are showcased in this research, the framework remains flexible, allowing for the incorporation of other algorithms. Experimental results prove the efficiency of the proposed method in significantly improving malicious URL detection and Text detection making it a promising solution in combating cyber fraud.

1. INTRODUCTION

Uniform Resource Locators (URLs) are used to access resources on the Internet, consisting of a protocol identifier and a resource name (IP address or domain). Attackers often manipulate URL structures to deceive users, spreading malicious URLs that redirect to harmful sites, execute code, or distribute malware. These links can spread rapidly through shared networks via downloads or messages. Malicious URLs are involved in common cyberattacks like Drive-by Downloads, Phishing, Social Engineering, and Spam. Statistics show that malicious, botnet, and phishing URLs rank among the top attack techniques and continue to grow in frequency and severity. To combat this, two main detection approaches exist: rule-based detection, which is fast but limited to known threats, and behaviour-based detection using machine learning, which can identify new threats by analysing URL behaviours. This research focuses on using machine learning (Support Vector Machine and Random Forest) to detect malicious URLs based on novel features extracted from both static and dynamic behaviours—marking a key contribution to existing literature.

2. LITERATURE SURVEY

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

[1] Machine learning techniques effectively detect malicious URLs by analysing their behaviours, using supervised, unsupervised, and semi-supervised algorithms. Studies have shown the effectiveness of these approaches.
[2,3,4] Supervised machine learning algorithms are used to demonstrate the system's effectiveness, enhancing malicious URL detection accuracy with new features. The system remains flexible, allowing for alternative algorithm implementation.
[6,7,8] Signature - based detection identifies malicious URLs by matching them against a database of predefined patterns. It is effective for recognizing known threats, as any URL that matches a stored signature is flagged as malicious. However, its major limitation is the inability to detect new or evolving threats that do not yet have a recorded signature, making it less effective against unknown attacks.
[9, 10, 11] Dynamic URL analysis involves examining the actions performed by URLs in real-time, whereas static analysis focuses on characteristics such as lexical, content, host, and popularity-based features. These studies have utilized online learning algorithms to extract URL attributes based on static and dynamic behaviours.
[12, 13] proposed methods for detecting malicious URLs through dynamic analysis, extracting features like character patterns, semantic groups, abnormal website behaviours, and host-based correlations. Despite these advancements, challenges remain in selecting suitable algorithms and effectively extracting relevant URL attributes.

3. PROBLEM STATEMENT

The rapid growth and continual evolution of malicious URLs—used for phishing, malware distribution, and other cyberattacks—pose a serious threat to users and organizations. Traditional signature-based detection systems, which rely on known URL patterns, cannot keep pace with novel or obfuscated attacks and require frequent manual updates. Consequently, there is a critical need for an automated detection framework that can accurately identify both known and previously unseen malicious URLs. This project addresses that gap by developing a machine-learning-based system that leverages a comprehensive set of static and dynamic URL features to improve detection accuracy and adapt to emerging threats.

4. PROPOSED SYSTEM

Malicious URL and text detection using machine learning represents a cutting-edge approach to cybersecurity, leveraging advanced algorithms to identify and mitigate online threats. One of the key strengths of this methodology lies in its ability to incorporate novel URL and text features extracted from both static and dynamic analyses, enhancing its detection capabilities. Static analysis involves examining the structural attributes of URLs and the content of text strings without executing them. This may include features such as URL length, domain reputation, presence of suspicious characters or keywords, and syntactic patterns in text. extracting these static features, the detection system can identify potential indicators of malicious intent, such as obfuscated URLs or phishing attempts

5. EXPERIMENTAL SETUP

Hardware System Configuration:
Processor - Intel i3 core, RAM - 4 GB (min), Hard Disk - 500 GB
Software Requirements:
Operating System - Windows 7 or above Coding Language - Java/J2EE (JSP, Servlet) Front End - HTML, CSS. Back End - J2EE, Database - My SQL, Tool - NetBeans

6. FUTURE SCOPE

The proposed system can be extended in several promising directions. First, incorporating advanced deep-learning architectures—such as CNNs over character embeddings, LSTMs for sequence modelling, or Transformer-based encoders—could automate feature learning and capture subtler URL patterns. Second, building a real-time streaming pipeline (e.g., with Kafka or Flink) would enable low-latency classification at network gateways or within browsers, providing instant protection. Third, augmenting our feature set with graph-based and contextual signals- modelling relationships among URLs, domains, IPs, and WHOIS data—can reveal coordinated malicious infrastructures that evade standalone analysis. Fourth, strengthening adversarial

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

robustness through adversarial training and continual retraining will help counter obfuscation tactics like homoglyphs or URL shortening. Fifth, extending support for internationalized domain names and multilingual content will close gaps exploited by non-ASCII homograph attacks. Sixth, integrating user-feedback loops and active-learning frameworks allows the model to prioritize ambiguous or novel samples for human labelling, accelerating adaptation to emerging threats. Finally, fusing URL classification with NLP-driven analysis of landing-page content or email text, and embedding explainable-AI methods (e.g., SHAP or LIME), will create a unified, transparent threat-detection platform that can be continuously updated to handle concept drift and deployed even on resource-constrained edge devices.

7. CONCLUSION

In conclusion, the proposed malicious URL detection system, leveraging machine learning algorithms and big data technology, offers a promising solution to combat the increasing risk of cyber fraud. By analyzing both static and dynamic behaviors of URLs, our system introduces novel features for more accurate detection of malicious URLs. The experimental results demonstrate significant improvements in detection capability, indicating the effectiveness of the proposed approach. Moreover, the flexibility of incorporating various machine learning algorithms underscores the adaptability and scalability of the system. With ongoing advancements in cyber threats, the proposed system provides a robust and user-friendly solution for enhancing network security and protecting users from malicious online activities. Further research and development in this area hold the potential to refine and extend the capabilities of the system, making it even more resilient against evolving cyber threats**.**

8. REFERENCES

[1] Thomas G. Dietterich. Ensemble Methods in Machine Learning. International Workshop on Multiple Classifier Systems, pp 1-15, Cagliari, Italy, 2000.
[2] Leo Breiman.: Random Forests. Machine Learning 45 (1), pp. 5- 32,(2001).
[3] C. Seifert, I. Welch, and P. Komisarczuk, “Identification of malicious web pages with static heuristics,” in Telecommunication Networks and Applications Conference, 2008. ATNAC 2008. Australasian. IEEE, 2008, pp. 91–96.
[4] S. Sinha, M. Bailey, and F. Jahanian, “Shades of grey: On the effectiveness of reputation-based “blacklists”,” in Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on. IEEE, 2008, pp. 57–64.
[5] S. Sheng, B. Wardman, G. Warner, L. F. Cranor, J. Hong, and C. Zhang, “An empirical analysis of phishing blacklists,” in Proceedings of Sixth Conference on Email and Anti-Spam (CEAS), 2009.
[6] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Identifying suspicious urls: an application of large-scale online learning,” in Proceedings of the 26th Annual International Conference on Machine Learning. ACM,2009, pp. 81–688.
[7] M. Cova, C. Kruegel, and G. Vigna, “Detection and analysis of drivebydownload attacks and malicious javascript code,” in Proceedings of the19th international conference on World wide web. ACM, 2010, pp. 281– 290.
[8] S. Purkait, “Phishing counter measures and their effectiveness– literature review,” Information Management & Computer Security, vol. 20, no. 5,pp. 382–420, 2012.
[9] M. Khonji, Y. Iraqi, and A. Jones, “Phishing detection: a literature survey,” IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2091–2121, 2013.
[10] B. Eshete, A. Villafiorita, and K. Weldemariam, “Binspect: Holistic analysis and detection of malicious web pages,” in Security and Privacyin Communication Networks. Springer, 2013, pp. 149–166.
[11] G. Canfora, E. Medvet, F. Mercaldo, and C. A. Visaggio, “Detection of malicious web pages using system calls sequences,” in Availability, Reliability, and Security in Information Systems. Springer, 2014, pp.226-23.
[12] R. Heartfield and G. Loukas, “A taxonomy of attacks and a survey of defence mechanisms for semantic social engineering attacks,” ACM Computing Surveys (CSUR), vol. 48, no. 3, p. 37, 2015.
[13] D. Sahoo, C. Liu, S.C.H. Hoi, “Malicious URL Detection using Machine Learning: A Survey”. CoRR, abs/1701.07179, 2017.

iJETRM
International Journal of Engineering Technology Research & Management
Published By:
https://www.ijetrm.com/

[14] Internet Security Threat Report (ISTR) 2019–Symantec. https://www.symantec.com/content/dam/symantec/docs/reports/istr-24- 2019-en.pdf [Last accessed 10/2019].
[15] Developer Information. https://www.phishtank.com/developer_info.php. [Last accessed 11/2019].