

IDENTIFYING URL OBFUSCATION TECHNIQUES IN FACEBOOK PHISHING URLS USING STRUCTURAL DETECTION RULES AND JACCARD SIMILARITY

Yasir Muin ¹, Arifandy Mario Mamonto ², Hairil Kurniadi Sirajuddin ³,
Saiful Do Abdullah ⁴

Department of Informatics, Faculty of Engineering, University Khairun

Email: yasirmuin@unkhair.ac.id

Abstract

Phishing URLs distributed through social media may employ structural and character-level obfuscation to make malicious links more difficult to recognize. Although previous studies have extensively investigated phishing URL classification, the distribution of specific URL obfuscation patterns and the changes in token composition associated with canonicalization require further empirical examination in Facebook-related phishing URL datasets. In particular, the relationship between rule-based structural detection and token-set similarity has not been sufficiently established in the context of this study. This research aims to identify structural obfuscation patterns, examine their co-occurrence, and evaluate changes in URL token composition using Jaccard similarity. A total of 1,000 phishing URLs collected from Facebook were analyzed using ten predefined structural detection rules. Jaccard similarity was calculated by comparing token sets before and after canonicalization. The results show that 292 URLs (29.2%) matched at least one detection rule, producing 329 detections. Long URLs (140 detections), excessive dashes (88), excessive subdomains (46), and URL shorteners (42) accounted for 96.0% of all detections. The average Jaccard similarity was 0.9991, while Hex/URL Encoding exhibited a median value of 0.934. These findings characterize the structural patterns captured by the proposed rules and indicate that Jaccard similarity distinguishes changes in token composition more clearly for encoding-related cases than for most structurally detected cases.

Keywords: Canonicalization, Cybersecurity, Facebook, Jaccard similarity, Phishing URL, URL obfuscation

INTRODUCTION

Phishing remains one of the most persistent cybersecurity threats because it exploits user inattention rather than technical vulnerabilities in

a system. Uniform Resource Locators (URLs) are the primary attack vector for phishing distribution, whether through email, short messages, or social media platforms. Social media

platforms such as Facebook exhibit distinctive distribution characteristics, including internal redirect mechanisms (l.php?u=), third-party link-shortening features, and social interaction-driven propagation that accelerate the reach of malicious URLs to users.

To evade detection, phishing actors employ various obfuscation techniques, namely manipulations of URL structure intended to appear legitimate or to be difficult to recognize. Research on phishing URL detection over the past decade has been dominated by black-box machine learning and deep learning approaches. Razaque et al. [1] identified that IoT security evaluations still insufficiently link real-time monitoring with cross-application query systems, while Sasi et al. [2] developed a taxonomy of IoT attacks without specifically addressing URL obfuscation. Tian et al. [3] proposed a modality-based taxonomy for malicious URL detection and highlighted the vulnerability of detection systems to content-cloaking tactics and geolinguistic bias in the datasets used. Related research on social media spam classification also provides context for understanding malicious content distribution in online social networks [13]. A broader review by Kale et al. [14] examined malicious URL detection techniques,

features, and datasets, providing further context for research on URL-based threats.

Several recent studies have begun to highlight the limitations of conventional classification models against dynamically obfuscated URLs. Zhang et al. [4] developed AdaptPUD to address concept drift in tailored phishing URLs, yet still treated obfuscation as a disturbance the model must overcome rather than as a primary object of identification. Xie et al. [5] proposed a dual-branch TCN model with mask attention to address missing URL features, but manual feature extraction remained time-consuming. Linh et al. [6] developed a Hybrid Embedding Transformer for real-time phishing classification, yet the transformer-based approach still demanded substantial computational resources and offered limited interpretability.

Similarity-based methods such as Jaccard Similarity have proven effective in other cybersecurity domains. Arikkat et al. [7] applied Jaccard Similarity to measure the correspondence of predicted MITRE ATT&CK Tactics and Techniques for Android applications, achieving high similarity scores, while Choi et al. [8] integrated Large-Language-Model-based expert feedback for knowledge calibration of phishing URL detection models. Sruthi and Manohar Naik [9]

developed an LSTM-based Siamese Network capable of generalizing to zero-day attack patterns, and in a follow-up study [10] integrated domain-adapted GloVe embeddings with an attention mechanism. Nevertheless, the direct application of Jaccard Similarity to measure the impact of obfuscation on URL token composition, particularly within the context of social media platforms such as Facebook, remains largely unexplored.

Another issue reinforcing the urgency of this study is concept drift and model adaptability. Li et al. [11] highlighted that machine-learning-based security systems are vulnerable to the tension between detection sensitivity and computational constraints as data continuously evolves, while van Geest et al. [12] emphasized that many phishing detection systems fail to adequately address real-world applicability despite high accuracy on test datasets. These conditions underscore the need for an identification approach that is computationally lighter and does not rely on model retraining, yet remains adaptive to new obfuscation variants emerging on social media platforms.

Related research on evolutionary feature selection for machine-learning-based malware classification further illustrates the role of feature representation in security analysis

[15], although malware classification and structural URL obfuscation remain distinct research problems. Beyond the core body of work on URL obfuscation, a wider set of studies shapes the context of phishing detection research. Attribution-focused surveys examine how cyber threat actors and Advanced Persistent Threat groups are identified and profiled [16, 18, 35], including network-level tactics, techniques, and procedures analysis [31] and natural-language-processing-based threat intelligence extraction [32, 33]. Large language models have increasingly been applied across the threat detection pipeline, from general cyber threat detection [19] to malicious code deobfuscation [24], while parallel work raises concerns about the security of LLMs themselves [39]. Complementary systematic reviews cover Windows malware detection [20], cyber data breach countermeasures [21], automated penetration testing [22], and domain-specific applications such as cyber threat intelligence in smart agriculture [23], underscoring that obfuscation and evasion are recurring themes across the broader malware and threat-intelligence literature, not only in URL-based phishing.

Within phishing detection specifically, graph-based approaches using encrypted traffic features [29],

general-purpose intelligent phishing detection systems [30], real-time hybrid detection frameworks [34], and modular defensive architectures such as StealthPhisher [40] each propose alternative technical strategies. Continuous multi-task pre-training has also been explored to address class imbalance [25], while Arabic-language smishing detection [17] demonstrates that obfuscation strategies extend across channels. Ransomware-focused studies [26, 36, 38], Android malware static analysis [27], and generative AI reviews [28, 37] illustrate adversarial evasion strategies. Trustworthy threat intelligence pipelines [41] and behavioural evidence that educated users remain susceptible to phishing [42] reinforce the need for detection approaches that are technically lightweight and interpretable.

Based on the discussion above, this study addresses a research gap regarding the identification of obfuscation techniques in phishing URLs circulating specifically on Facebook. Unlike prior studies that treat obfuscation as a disturbance, this study positions obfuscation techniques as the primary object of identification and applies Jaccard Similarity to quantify how much each technique alters URL token composition after canonicalization. The objectives are: (1) to identify the

type and frequency of obfuscation techniques in Facebook phishing URLs, (2) to analyze co-occurrence patterns among techniques within a single URL, and (3) to measure the impact of each technique on URL token composition.

LITERATURE REVIEW

Malicious URL Detection and Structural Obfuscation

Malicious URL detection has been investigated through a variety of approaches, ranging from feature-based analysis to machine-learning and deep-learning models. Tian et al. [3] reviewed malicious URL detection techniques, datasets, and code repositories, providing an overview of the methodological approaches used to identify malicious links. Osmanoglu et al. [14] likewise examined detection techniques, features, and datasets, highlighting the importance of feature selection and data characteristics. These reviews provide a foundation; however, identifying a malicious URL and characterizing specific structural obfuscation techniques represent different analytical objectives.

Structural URL obfuscation involves modifying the appearance or composition of a URL in ways that make suspicious characteristics less apparent. Relevant patterns include unusually long URLs, excessive

hyphens, multiple subdomains, URL shortening, percent encoding, and atypical structural components. Such patterns can be examined directly from URL strings without relying on page content or trained classification models. In this study, structural obfuscation is operationalized through a predefined set of ten detection rules.

Phishing URLs in Social Media Environments

Social media platforms provide an important context for studying malicious links because URLs circulate through posts, comments, private messages, and user interactions. Research on Twitter spam classification [13] provides related context for social networking environments. However, findings from one platform cannot automatically be generalized to another. The present study focuses specifically on phishing URLs collected from Facebook to characterize structural patterns observed in this platform's dataset.

Rule-Based Identification and Co-Occurrence

Rule-based analysis provides a transparent way to identify predefined structural characteristics in URLs. Unlike predictive models, rule-based methods apply explicit conditions to each URL, making it possible to report exactly which characteristics triggered a detection. Co-occurrence analysis complements individual rule counts

by identifying combinations of structural characteristics within the same URL, describing dataset structure without assuming a single pattern is sufficient to determine maliciousness.

Jaccard Similarity for URL Token-Set Comparison

Similarity measures are used in cybersecurity applications to compare sets of features or behaviors. Arikkat et al. [7] applied Jaccard similarity in mapping Android application behaviors. For two token sets A and B, Jaccard similarity is defined as $J(A,B) = |A \cap B| / |A \cup B|$. In this study, Jaccard similarity compares the token set extracted from an original URL with the token set obtained after canonicalization to measure token composition preservation.

RESEARCH METHODOLOGY

This study employs a quantitative descriptive approach based on structural analysis of phishing URLs. The research was carried out through six sequential stages, as shown in Figure 1.

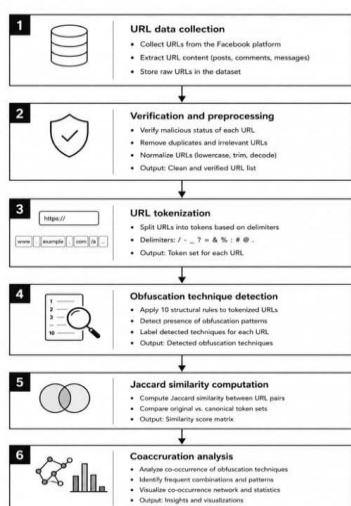


Figure 1. Flow diagram of the research stages.

URL Data Collection & Preprocessing

A total of 1,000 phishing URL samples circulating on Facebook were collected from public posts, comments, and accessible messages. Malicious status was verified via PhishTank and manual inspection. Each URL was normalized to lowercase and separated into scheme, authority, path, and query components, then tokenized using standard delimiters (., /, ?, =, &, -).

Obfuscation Detection Criteria

Every URL was evaluated against ten structural detection rules formulated from manipulation patterns commonly used by phishing actors, summarized in Table 1.

Table 1. Obfuscation technique detection criteria

Technique	Detection criterion
-----------	---------------------

Long URL (>75)	Total URL length exceeds 75 characters
Excessive Dashes	Number of hyphens in domain name exceeds threshold
Excessive Subdomains	Number of subdomain levels exceeds normal threshold
URL Shortener	Domain matches known URL-shortening service list
Hex/URL Encoding	Contains percent-escape sequences (%XX)
Double Slash Redirect	Double slash (//) present in path segment
IP Address	Host portion is a numeric IP address
@ Symbol	The @ symbol is present in authority portion
Punycode (IDN)	Domain label begins with xn-- prefix
Non-standard Port	URL specifies port number other than 80 or 443

To measure how far each technique alters token composition, Jaccard Similarity was computed between original and canonicalized token sets using Eq. (1):

$$J(A, B) = |A \cap B| / |A \cup B| \quad (1)$$

IMPLEMENTATION AND DISCUSSION

Of the 1,000 phishing URL samples analyzed, 292 URLs (29.2%) contained at least one obfuscation technique, while 708 URLs (70.8%) matched none of the criteria, as illustrated in Figure 2. A total of 329 detections were recorded across all samples. The distribution is summarized in Table 2 and visualized in Figure 3.

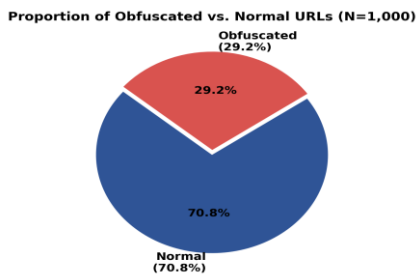


Figure 2. Proportion of obfuscated vs. normal URLs (N=1,000).

Table 2. Distribution of detected obfuscation techniques

Technique	URLs	% of total (N=1,000)	% of detections (n=329)
Long URL (>75)	140	14.0%	42.6%
Excessive Dashes	88	8.8%	26.7%
Excessive Subdomains	46	4.6%	14.0%
URL Shortener	42	4.2%	12.8%
Hex/URL Encoding	6	0.6%	1.8%
Double Slash Redirect	3	0.3%	0.9%
IP Address	2	0.2%	0.6%
@ Symbol	2	0.2%	0.6%
Punycode (IDN)	0	0.0%	0.0%
Non-standard Port	0	0.0%	0.0%

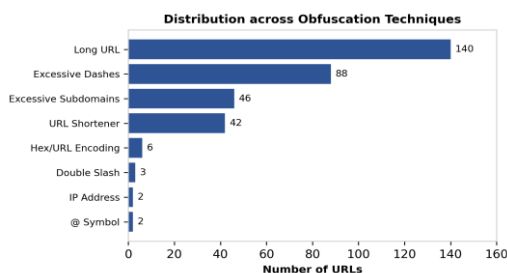


Figure 3. Distribution of URL counts across technique categories.

Structural Dominance & Length Profile

Long URL (140), Excessive Dashes (88), Excessive Subdomains (46), and URL Shortener (42) combined account for 316 of 329 detections (96.0%), demonstrating that obfuscation on Facebook operates predominantly at the structural level. The continuous URL length profile is shown in Figure 4 and Table 3 / Figure 5.

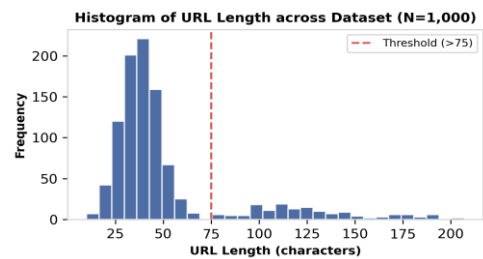


Figure 4. Histogram of URL length across dataset (N=1,000).

Table 3. Distribution of URL length categories

Length Range (chars)	URLs	Percentage (%)
1 - 25	135	13.5%
26 - 50	535	53.5%
51 - 75	190	19.0%
76 - 150	115	11.5%
> 150	25	2.5%

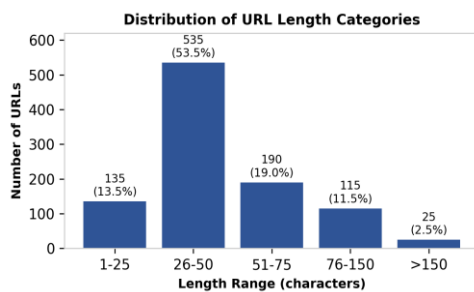


Figure 5. Distribution of URL length categories.

Average URL length across the dataset is 50.6 characters. More than half (53.5%) fall within 26–50 characters, confirming that the Long URL rule isolates a deliberate upper tail rather than reflecting typical platform links.

Technique Layering and Co-Occurrence

The distribution of technique counts per URL is shown in Table 4 and Figure 6. Most obfuscated URLs (88.7%) utilized only a single technique. Two-technique combinations appeared in 29 URLs (9.9%), and three techniques in 4 URLs (1.4%).

Table 4. Distribution of the number of techniques per URL

Techniques / URL	URLs	Percentage (%)
0 (Normal)	708	70.8%
1 Technique	259	25.9%
2 Techniques	29	2.9%
3 Techniques	4	0.4%

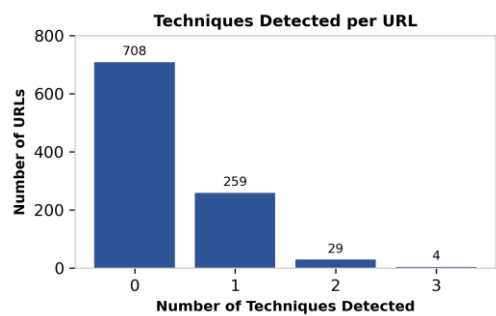


Figure 6. Distribution of obfuscation techniques per URL.

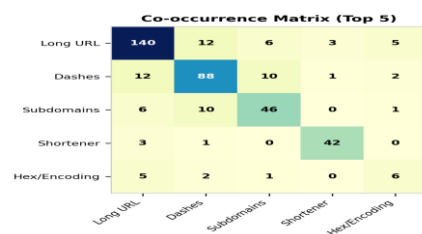


Figure 7. Co-occurrence matrix of top obfuscation techniques.

The co-occurrence matrix (Figure 7) reveals that the top co-occurring pairs are Long URL with Excessive Dashes (12) and Excessive Subdomains with Excessive Dashes (10), reflecting structural reinforcement.

Jaccard Similarity Analysis

The overall average Jaccard Similarity was 0.9991. Distribution across all detections is shown in Figure 8 and by category in Figure 9.

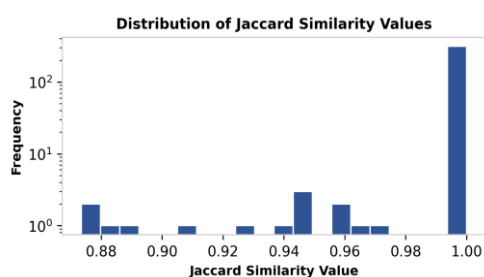


Figure 8. Distribution of Jaccard Similarity values across detections.

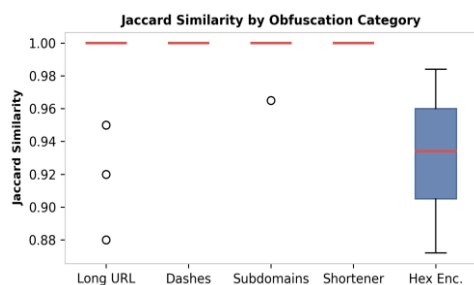


Figure 9. Distribution of Jaccard Similarity by category.

Figure 9 demonstrates that Hex/URL Encoding yields a markedly lower median Jaccard value of 0.934 (range 0.872–0.984), confirming that token composition changes significantly under character-level encoding, whereas structural techniques preserve exact token sets. A summary of overall analysis statistics is presented in Table 5.

Table 5. *Summary statistics of analysis results*

Metric	Value
Total URLs analyzed	1,000
Obfuscated URLs (≥ 1 technique)	292 (29.2%)
URLs without obfuscation	708 (70.8%)
Total technique detections	329
Average URL length	50.6 characters

Average Jaccard Similarity	0.9991
Most dominant technique	Long URL (>75) -- 140 URLs
Rarest detected techniques	IP Address and @ Symbol -- 2 URLs each
Techniques not found	Punycode (IDN) and Non-standard Port

CONCLUSION

Analysis of 1,000 Facebook phishing URLs revealed that 29.2% employed obfuscation, heavily dominated by structural techniques (Long URL, Excessive Dashes, Subdomains, Shorteners) accounting for 96.0% of detections. As many as 88.7% of obfuscated links used a single technique. An average Jaccard similarity of 0.9991 confirms that canonicalization leaves token sets intact for structural obfuscation while effectively isolating Hex/URL Encoding (median 0.934). Consequently, lightweight structural rule-based filters provide efficient coverage against social media phishing threats.

REFERENCES

- [1] Razaque, A., Hariri, S., Alajlan, A.M., Yoo, J. (2025). *A comprehensive review of cybersecurity vulnerabilities, threats, and solutions for the Internet of Things at the network-cum-application layer*. Computer Science Review, 58: 100789.

- <https://doi.org/10.1016/j.cosrev.2025.100789>
- [2] Sasi, T., Lashkari, A.H., Lu, R., Xiong, P., & Iqbal, S. (2024). *A comprehensive survey on IoT attacks: Taxonomy, detection mechanisms and challenges*. Journal of Information and Intelligence..<https://doi.org/10.1016/j.jiixd.2023.12.001>
- [3] Tian, Y., Yu, Y., Sun, J., & Wang, Y. (2025). *From past to present: A survey of malicious URL detection techniques, datasets and code repositories*. Computer Science Review, 58: 100810..<https://doi.org/10.1016/j.cosrev.2025.100810>
- [4] Zhang, Z., Wu, J., Lu, N., Shi, W., & Liu, Z. (2025). *AdaptPUD: An accurate URL-based detection approach against tailored deceptive phishing websites*. Computer Networks, 265: 111303 <https://doi.org/10.1016/j.comnet.2025.111303>
- [5] Xie, L., Zhang, H., Yang, H., Hu, Z., & Cheng, X. (2025). *A scalable phishing website detection model based on dual-branch TCN and mask attention*. Computer Networks, 263: 111230..<https://doi.org/10.1016/j.comnet.2025.111230>
- [6] Linh, D.M., Thua, H.T., & Hung, T.C. (2026). *Real-time phishing uniform resource locator detection based on hybrid embedding transformer and retraining-free inferencing*. Computers & Electrical Engineering, 136: 111001..<https://doi.org/10.1016/j.compeleceng.2026.111001>
- [7] Arikkat, D.R., Vinod, P., Rafidha Rehiman, K.A., Nicolazzo, S., Arazzi, M., Nocera, A., & Conti, M. (2025). *DroidTTP: Mapping android applications with TTP for Cyber Threat Intelligence*. Journal of Information Security and Applications, 93..<https://doi.org/10.1016/j.jisa.2025.104112>
- [8] Choi, S.H., Buu, S.J., & Cho, S.B. (2026). *LLM-structured expert feedback for knowledge calibration of deep learning model to detect phishing URL*. Applied Soft Computing, 197: 115234..<https://doi.org/10.1016/j.asoc.2026.115234>
- [9] Sruthi, K., & Manohar Naik, S. (2025). *A novel framework for effective phishing URL detection using an LSTM-based siamese network*. Knowledge-Based Systems, 310: 113900..<https://doi.org/10.1016/j.knosys.2025.113900>
- [10] Sruthi, K., & Manohar Naik, S. (2025). *A novel framework for phishing attack detection using*

- domain-adapted GloVe embeddings and attention-enhanced neural sequence model*. *Applied Soft Computing*, 188: 114441. <https://doi.org/10.1016/j.asoc.2025.114441>
- [11] Li, C., Wang, X., & Song, L. (2026). *Concept drift handling in security applications: A survey across unimodal to multimodal data*. *Neurocomputing*, 701: 134490.. <https://doi.org/10.1016/j.neucom.2026.134490>
- [12] van Geest, R.J., Cascavilla, G., Hulstijn, J., & Zannone, N. (2024). *The applicability of a hybrid framework for automated phishing detection*. *Computers & Security*, 139: 103736. <https://doi.org/10.1016/j.cose.2024.103736>
- [13] Dhar, S., & Bose, I. (2024). *An ensemble deep learning model for fast classification of Twitter spam*. *Information & Management*, 61(3): 104052. <https://doi.org/10.1016/j.im.2024.104052>
- [14] Osmanoglu, M., Gupta, D., Ozkan, M., Ar, Y., & Aslan, O. (2026). *A comprehensive review of malicious URLs: Detection techniques, features and datasets*. *Computers and Electrical Engineering*, 136: 111186. <https://doi.org/10.1016/j.compeleng.2026.111186>
- [15] Kale, G., Bostanci, G.E., & Celebi, F.V. (2024). *Evolutionary feature selection for machine learning based malware classification*. *Journal of King Saud University - Computer and Information Sciences*. <https://doi.org/10.1016/j.jksuci.2024.101920>
- [16] Rani, N., Saha, B., & Shukla, S.K. (2025). *A comprehensive survey of automated Advanced Persistent Threat attribution: Taxonomy, methods, challenges and open research problems*. *Journal of Information Security and Applications*, 92: 104076. <https://doi.org/10.1016/j.jisa.2025.104076>
- [17] Al Saidat, M.R., Yerima, S.Y., & Shaalan, K. (2026). *A hybrid deep learning framework for Arabic smishing detection: Dataset creation, model design, and explainability*. <https://doi.org/10.1016/j.cose.2026.104700>
- [18] Prasad, N., Diro, A., Warren, M., & Fernando, M. (2025). *A survey of cyber threat attribution: Challenges, techniques, and future directions*. *Computers & Security*, 157: 104606..

- <https://doi.org/10.1016/j.cose.2025.104606>
- [19] Chen, Y., Cui, M., Wang, D., Cao, Y., Yang, P., Jiang, B., Lu, Z., & Liu, B. (2024). *A survey of large language models for cyber threat detection*. *Computers & Security*, 145: 104016.. <https://doi.org/10.1016/j.cose.2024.104016>
- [20] Maniriho, P., Mahmood, A.N., & Chowdhury, M.J.M. (2024). *A systematic literature review on Windows malware detection: Techniques, research issues, and future directions*. *Journal of Systems and Software*, 209: 111921. <https://doi.org/10.1016/j.jss.2023.111921>
- [21] Yeboah, P.N., Kayes, A.S.M., Rahayu, W., Pardede, E., & Mahbub, S. (2026). A systematic literature survey of machine learning approaches to cyber data breach detection. <https://doi.org/10.1016/j.cose.2026.104650>
- [22] Wang, Y., Liu, S., Wang, W., Zhou, C., Zhang, C., Jin, J., & Zhu, C. (2025). *A unified modeling framework for automated penetration testing*. *Computers & Security*, 159: 104787. <https://doi.org/10.1016/j.cose.2025.104787>
- [23] Bui, H.T., Aboutorab, H., Mahboubi, A., Gao, Y., Sultan, N.H., Parvez, M.Z., Bewong, M., & Islam, R. (2024). *Agriculture 4.0 and beyond: Evaluating cyber threat intelligence sources and techniques in smart farming ecosystems*. *Computers & Security*, 140: 103754.. <https://doi.org/10.1016/j.cose.2024.103754>
- [24] Patsakis, C., Casino, F., & Lykousas, N. (2024). *Assessing LLMs in malicious code deobfuscation of real-world malware campaigns*. arXiv preprint, arXiv:2404.19715.. <https://doi.org/10.48550/arXiv.2404.19715>
- [25] Li, Y., Liu, Y., Li, P., Jia, Y., & Wang, Y. (2025). *Continuous multi-task pre-training for malicious URL detection and webpage classification*. *Computer Networks*, 268: 111513.. <https://doi.org/10.1016/j.comnet.2025.111513>
- [26] Zhao, X., Wang, Y., Zhang, L., & Liu, H. (2025). *Exploring the ransomware ecosystem and the activities of ransomware groups*. *Computers & Security*, 151: 104320. <https://doi.org/10.1016/j.cose.2025.104320>
- [27] Kharnotia, S., Arora, B., & Kour, R. (2026). *Feature-driven static*

- analysis for learning-based android malware detection: A review.* ICT Express, 12(1): 186-198..
<https://doi.org/10.1016/j.ict.2025.11.002>
- [28] Ferrag, M.A., Shu, L., Yang, X., & Derhab, A. (2025). *Generative AI in cybersecurity: A comprehensive survey.* IEEE Access, 13: 21045-21072..<https://doi.org/10.1109/ACCESS.2025.3365349>
- [29] Manguli, K., Kondaiah, C., Pais, A.R., & Rao, R.S. (2025). *GraPhish: A graph-based approach for phishing detection from encrypted TLS traffic.* Journal of Information Security and Applications..<https://doi.org/10.1016/j.jisa.2025.104216>
- [30] Vennela, P., Rao, K.S., Kumar, A., & Sharma, R. (2026). *Intelligent cybersecurity systems for phishing detection.* Computer Communications, 215: 108-121..
<https://doi.org/10.1016/j.comcom.2025.10.015>
- [31] Alageel, A., & Maffeis, S. (2026). *Investigation of advanced persistent threats network-based tactics, techniques and procedures.* Computer Networks, 278: 112069..
<https://doi.org/10.1016/j.comnet.2025.112069>
- [32] Arazzi, M., Arikkat, D.R., Nicolazzo, S., Nocera, A., Rehiman, R.K.A., Vinod, P., & Conti, M. (2025). *NLP-based techniques for Cyber Threat Intelligence.* Computer Science Review, 58: 100765..
<https://doi.org/10.1016/j.cosrev.2025.100765>
- [33] Alsirhani, A., Ali, S., Humayun, M., Rehman, S.U., & Alzahrani, M.S. (2026). *NLP in cybersecurity: Advances, applications, and future directions for threat analysis and dark web intelligence.* Neurocomputing, 699:134365..<https://doi.org/10.1016/j.neucom.2026.134365>
- [34] Prasad, Y.B., Dondeti, V., & Dondeti, R. (2026). *Phishing URL Detection in Real-Time: Machine Learning and Deep Learning Approaches on a Custom Threat Intelligence Dataset.* Procedia Computer Science, 283: 4859-4879..<https://doi.org/10.1016/j.procs.2026.06.541>
- [35] Alanazi, M.M., Wahab, A.W.A., Idris, M.Y.I., & Nandy, T. (2027). *Profiling advanced persistent threat groups: a systematic review, activity classification, and*

- countermeasure taxonomy*. Computer Standards & Interfaces, 99: 104190.. <https://doi.org/10.1016/j.csi.2026.104190>
- [36] Al-rimy, B.A.S., Maarof, M.A., Shaid, S.Z.M., & Doss, R. (2024). *Ransomware early detection: A survey*. Computer Networks, 239: 110138. <https://doi.org/10.1016/j.comnet.2023.110138>
- [37] Yigit, Y., Buchanan, W.J., Tehrani, M.G., & Maglaras, L. (2025). *Review of Generative AI methods in cybersecurity*. Internet of Things and Cyber-Physical Systems, 5: 241-261.. <https://doi.org/10.1016/j.iotcps.2025.01.002>
- [38] Abdullah, S., & Faye, I. (2024). *Securing the Industrial Internet of Things against ransomware attacks: a comprehensive analysis of the emerging threat landscape and detection mechanisms*. Journal of Network and Computer Applications, 224:103809.<https://doi.org/10.1016/j.jnca.2024.103809>
- [39] Li, Q., & Fung, B. (2025). *Security concerns for Large Language Models: A survey*. <https://doi.org/10.1016/j.jisa.2025.104284>
- [40] Prasad, A., Yadav, V., Solanki, C., Goswami, H., Jha, T., & Nagal, D. (2026). *StealthPhisher: A defensive framework against phishing attack using hybrid deep learning and GenAI*. Expert Systems with Applications, 299: 130205.. <https://doi.org/10.1016/j.eswa.2025.130205>
- [41] Alam, M.T., Bhusal, D., Park, Y., & Rastogi, N. (2025). *Trustworthy and explainable malware threat intelligence through STIX-based CTI and nature-inspired optimization*. Computer Security & Intelligence, 45: 102190..<https://doi.org/10.1016/j.csi.2025.102190>
- [42] Gagniuc, P. (2026). *Why educated users still click: a review on phishing and the limits of human judgment*. Computer Science Review, 59: 100830..<https://doi.org/10.1016/j.cosrev.2026.100830>