DETECTING MALICIOUS URLS: TRENDS, CHALLENGES AND THE ROLE OF BROWSER EXTENSIONS

Homam Eltaj and Maria Ajaj

Department of Cybersecurity, Dar Al-Hekma University, Jeddah, Saudi Arabia

ABSTRACT

Malicious URLs continue to pose a significant cybersecurity threat, frequently bypassing conventional detection systems through URL masking and rapid domain switching. This paper proposes a conceptual framework for a lightweight, real-time browser extension designed to block harmful URLs using a dynamically updated blacklist. Beyond detection, the system integrates a user-awareness module offering contextual security guidance and regulatory resources, promoting safer online behavior. The proposed extension aims to address key limitations in existing tools by offering client-side protection with minimal performance overhead and a communitydriven reporting mechanism. Although the current design employs deterministic logic, future development will incorporate machine learning to enhance adaptability and improve classification of emerging threats. By combining real-time detection, user education, and planned AI integration, the proposed solution contributes a practical and forward-looking approach to strengthening browserlevel web security.

KEYWORDS

Malicious URL Detection, Browser Extensions, Phishing Prevention, AI-Based Security, Cybersecurity Awareness

1. INTRODUCTION

Malicious URLs serve as a primary vector for cyberattacks such as phishing, malware distribution, and credential theft. As threat actors continuously evolve their tacticsusing techniques like URL masking, fast flux domains, and link shortening, traditional detection systems, particularly those based on static blacklists or heuristic rules, often fail to deliver timely and accurate protection [1], [2].

Browser extensions have emerged as a practical client-side defense, enabling real-time link scanning and warning systems. However, existing tools are limited by static detection logic, poor adaptability, inconsistent update mechanisms, and a lack of user-facing educational support. These gaps create opportunities for designing more responsive, informative, and user-centric browser-based security solutions.

This paper presents a conceptual model for a browser extension that addresses these limitations through an integrated approach to detection and awareness. The proposed system is designed to be lightweight, efficient, and scalable, with built-in mechanisms for real-time threat mitigation and user guidance.

The main contributions of this articleinclude the design of a lightweight browser extension capable of real time malicious URL detection, the integration of a dynamic blacklist that receives

continuous updates from trusted threat intelligence sources, and the addition of a built-in educational component aimed at enhancing user cybersecurity awareness. Furthermore, the proposed system features a flexible architecture that supports future integration of artificial intelligence and machine learning models to improve adaptability and detection accuracy.
This article is structured as follows:

- **Section 2**reviews malicious URL threats, attack techniques, and existing detection approaches.
- **Section 3**introduces the proposed detection system, outlining its technical components and operational workflow.
- **Section 4**presents a discussion of limitations and potential advancements, including AI integration and mobile deployment.
- **Section 5** concludes the paper by summarizing findings and identifying directions for future cybersecurity research.

2. LITERATURE REVIEW

Detecting malicious URLs remains essential to modern cybersecurity. As attackers adopt increasingly deceptive techniques ranging from URL masking to rapid URL churn, security systems must move beyond static lists towards more adaptive and intelligent solutions. This section provides a structured overview of detection paradigms, categorized by methodology, and assesses their effectiveness in real-time threat mitigation.

2.1. Signature-Based Detection

Signature-based detection is among the earliest and simplest forms of malicious URL filtering. It relies on exact pattern matching against known blacklists or domain reputation databases. Tools such as Google Safe Browsing, Spamhaus, and SURBL maintain repositories of flagged domains and IPs that are regularly updated and widely integrated into browser-level or email security systems[3, p. 202].

While efficient in terms of speed and computational load, signature-based methods struggle to detect novel or masked URLs, making them largely ineffective against zero-day attacks and rapidly evolving threat actors [2]. The static nature of these systems also increases reliance on third-party update frequency and blacklist scopes.

2.2. Heuristic-Based Detection

Heuristic methods enhance static blacklists by incorporating lightweight rule-based logic and contextual feature analysis. These systems evaluate characteristics such as URL length, token count, special character usage, and domain registration age. Notable tools like PhishTank use community submissions and basic heuristics to identify suspicious URLs [4].

Heuristics offer improved adaptability over signatures but often suffer from high false positive rates due to the simplistic nature of their rules. Their effectiveness is further constrained by their inability to generalize well across diverse and evolving attack strategies [5].

2.3. Machine Learning-Based Detection

Machine learning (ML) techniques provide a more robust alternative by learning discriminative patterns from large-scale labeled datasets. Common algorithms used in URL detection include

Support Vector Machines (SVM), Random Forests, and Naïve Bayes classifiers. These models rely on engineered features such as n-gram token distributions, lexical statistics, and domain-related metadata [6].

Public datasets such as PhishTank, OpenPhish, and URLNet serve as the foundation for model training and benchmarking [7]. Despite their promise, ML-based solutions require extensive feature engineering and frequent retraining to adapt to evolving threats. They may also be challenged by data imbalance and the presence of adversarial input [8].

2.4. Deep Learning and AI-Based Detection

Recent advances in deep learning eliminate the need for manual feature design by leveraging raw URL input and content metadata. Architectures like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and transformer-based models have shown success in classifying URLs as benign or malicious based solely on character sequences or domain structure [9], [10].

These models can incorporate embeddings, behavioral analysis, and even NLP techniques to interpret landing page content. While deep learning enhances detection accuracy and generalization, it introduces challenges in explainability, resource requirements, and dataset dependency [11]. Additionally, their use in real-time environments is limited by latency and computational constraints.

2.5. Browser Extension-Based Detection

Browser extensions represent a client-side defense mechanism that can detect threats at the point of interaction. Tools such as Netcraft, Bitdefender TrafficLight, and Avast Online Security operate by scanning accessed URLs and comparing them to blacklists or heuristic rules in real time [12].

These extensions are especially valuable for their user-facing immediacy and integration with user behavior. However, their reliance on limited detection logic often leaves them vulnerable to evasion. Moreover, browser extensions raise concerns related to performance overhead, permission scope, and user privacyespecially when integrated with third-party data sources or analytics engines [13].

Table 1: Summary of Detection Approaches

Approach	Examples	Strengths	Limitations	Real-time Capability
Signature-Based	Google Safe Browsing, Spamhaus	Fast, Simple	High evasion rate, outdated quickly	Yes
Heuristic-Based	PhishTank, SURBL	Lightweight, community-fed	High false positives, Limited generalization	Yes
ML-Based	SVM, RF, OpenPhish, URLNet	Adaptive, data-driven	Requires feature engineering, retraining	Conditional
Deep Learning / AI	CNN, RNN, Transformers	Featureless input, scalable	Expensive to train, limited interpretability	Potentially
Browser Extension-Based	Netcraft, Avast, TrafficLight	User-friendly, immediate	Limited AI use, privacy tradeoffs	Yes

2.6. Threat Landscape for Browser-Based Attacks

Malicious URLs are a central attack vector in modern cybercrime, used to launch phishing campaigns, distribute malware, and hijack user credentials. These threats are especially dangerous in browser environments, where deceptive links can directly manipulate user actions or inject harmful scripts. Several notable attack types illustrate the diversity and severity of these threats:

- **Phishing Attacks:** Attackers create convincing replicas of trusted websites to harvest login credentials or sensitive data. These URLs often arrive via email or social platforms and evolve quickly, rendering static blacklists ineffective [1].
- **Redirect URLs & Fake Virus Alerts:** Adversaries hijack navigation flows to redirect users to fake alerts or malware download pages. These are common on mobile devices, particularly Android, due to limited phishing protection and exposure through third-party app stores [12].
- **Clickjacking:** This technique overlays invisible malicious elements over legitimate interface components, tricking users into activating harmful actions. Detection is difficult because the underlying URL may appear legitimate [9].
- **Credential Stuffing and Account Takeovers:** Reused or weak credentials are targeted via malicious login portals or automated scripts that simulate legitimate access attempts, often hidden within redirect chains or shortened URLs [11].

These evolving techniques underscore the need for browser-based security systems that operate in real time, adapt to new URL behaviors, and integrate user education and feedback. The proposed extension addresses these requirements by combining dynamic blacklist validation with future-ready AI integration and built-in user guidance.

Table 3 summarizes key URL-based threat categories discussed in this section, highlighting their nature, impact, and challenges for detection.

Table 2: Common URL-Based Threats Targeting Browser Users and Associated Detection Challenges

Threat Type	Description	Impact	Detection Challenge
Phishing	Fake websites mimicking trusted brands to steal credentials	Identity theft, account compromise	Frequent domain changes, social engineering tactics
Redirect URLs	Links that auto-redirect to malware or scam content	Malware infection, scareware	Hard to track origin; hidden in shortened links
Clickjacking	Invisible or disguised links that hijack user actions	Unauthorized clicks, privilege misuse	Legitimate-looking interfaces
Credential Stuffing	Automated login attempts using leaked username-password pairs	Account takeovers, data breaches	URL masking and request chains
Drive-by Downloads	URLs triggering automatic download of malware or spyware	Device compromise	Often bypass antivirus if user doesn't click
Fast-Flux Hosting	Rapid DNS changes to evade detection	Persistent phishing, botnets	Breaks static blacklist mechanisms

3. CONCEPTUAL PROPOSAL: AI-ENHANCED BROWSER EXTENSION

This section outlines a conceptual browser extension designed to detect and block malicious URLs in real time. Unlike conventional blacklist-based systems, the proposed solution integrates AI capabilities for adaptive detection and cybersecurity education.

3.1. Scope of Work

The proposed security system is designed as a browser extension for Google Chrome, selected due to its widespread adoption, extensive extension APIs, and built-in support for real-time web request interception. This scope ensures broad user reach and simplifies integration with existing browsing environments.

The extension will function within the browser context only, monitoring user-accessed URLs and filtering them through a local blacklist. It will not have access to page content, user credentials, or background system processes, ensuring minimal intrusion and privacy preservation.

Chrome's extension model provides APIs such as webRequest, declarativeNetRequest, and storage, which support URL interception, decision-based blocking, and local caching of blacklist data. These components align with the proposed system's goals of lightweight deployment, client-side decision-making, and real-time response.
This scope excludes:

- OS-level or network-wide filtering.
- Full-page content analysis.
- User behavior profiling.

By leveraging Chrome's modular structure and security sandbox, the extension can be developed and iterated rapidly, supporting future AI integration and updates without compromising usability or performance.

3.2. System Overview

The proposed extension functions as a lightweight tool for Google Chrome that monitors user-accessed URLs prior to page load. The architecture includes four core components:

- URL monitoring.
- Comparison against a dynamic blacklist.
- User alert interface.
- A user-awareness module that delivers cybersecurity tips and links to relevant regulatory resources.

When a malicious URL is detected, the user is warned and provided with contextual awareness guidance. The system also allows user-submitted reports to enhance the evolving blacklist through community feedback.

3.3. Detection Workflow

The extension queries a real-time blacklist sourced from reputable cybersecurity databases. If a URL is flagged, the system presents the user with three options: close the page, view best-

practice guidelines, or access cybersecurity regulations. This encourages safe decision-making while maintaining usability.

To extend its capabilities, the system is designed to integrate AI classifiers capable of analyzing previously unseen threats by learning from URL patterns, domain reputation, and behavioral indicators.

The diagram below illustrates the automated detection and alert mechanism, showing how the extension scans URLs, verifies them against a dynamic blacklist, and responds to identified threats.

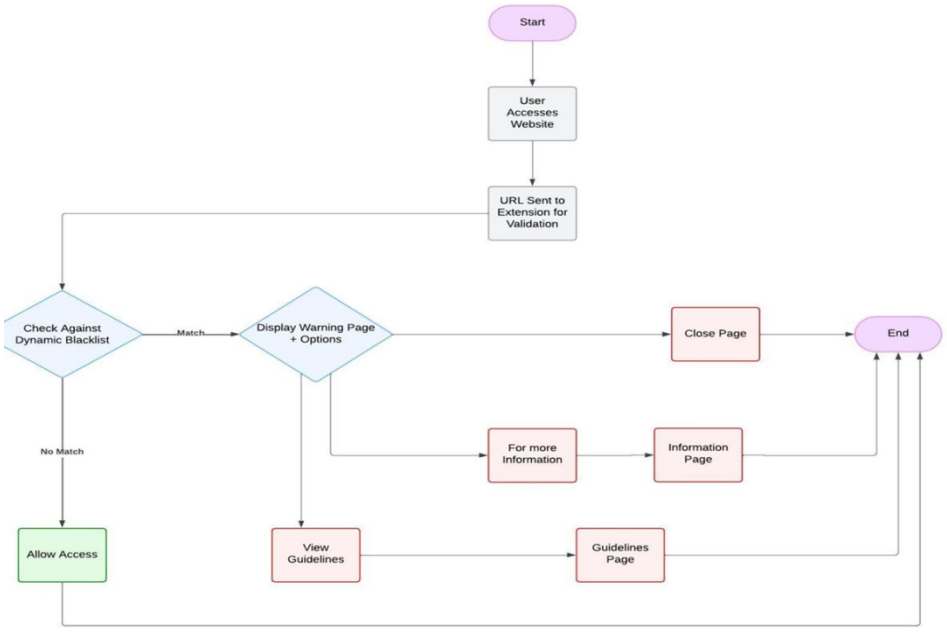

Figure 1: Flowchart of work

3.4. System Architecture Overview (Technical Detail)

The extension operates in a multi-phase sequence. Upon URL access, the system first queries a local cache of the dynamic blacklist, synchronized with trusted external sources. If the URL is flagged, the user is presented with an alert offering the option to exit or view security guidance. URLs not found on the blacklist are temporarily logged for user-feedback tagging or future AI-based classification. The proposed model supports modular upgrades, allowing the AI component to act as a secondary layer for edge-case detection. All detection occurs client-side to preserve privacy and reduce latency, with optional cloud sync only for list updates and feedback processing.

3.5. Planned AI Integration

While the initial system relies on deterministic logic and blacklists, future enhancements will introduce machine learning models, such as Random Forests or CNNs. These will assess lexical patterns, token entropy, domain attributes, and redirect behavior to improve zero-day threat detection. The AI layer is designed to actas a secondary verification layer when the blacklist returns no match, improving real-time protection against evolving attack strategies.

3.6. Novelty and Differentiation

Unlike conventional browser extensions that primarily rely on static blacklists or heuristic rules, the proposed solution introduces a dual-layer defense: real-time detection enhanced by AI (planned), and a built-in user education component that promotes proactive cybersecurity behavior.

To better illustrate its comparative advantages, the following table outlines key differentiators between existing extensions and the proposed system:

Table 3: Comparative Overview – Existing Extensions vs. Proposed Solution

Feature	Existing Browser Extensions (e.g., Netcraft, Bit defender Traffic Light, Avast Online Security)	Proposed Extension
Real-Time URL Detection	Yes	Yes
Dynamic Blacklist Integration	Often limited to static or periodic updates	Live updates planned
AI-Based Detection	Largely absent	Planned integration
User Awareness & Education	Rarely included	Built-in guidance
User Feedback Mechanism	Minimal (if any)	User-report feature
Resource Efficiency	Varies; some extensions affect performance	Lightweight design
Privacy-Preserving Approach	Often requires intrusive permissions	Non-invasive model

The comparison table above further emphasizes the conceptual advantages of the proposed system, particularly in bridging technical protection with user empowerment. By addressing key limitations in existing tools such as adaptability, user feedback, and education, the system introduces a more holistic approach to malicious URL detection.

In summary, the novelty of the proposed system lies in its holistic approach, combining real-time detection, privacy aware architecture, future ready AI integration, and a user centered awareness module. While existing browser extensions tend to focus narrowly on static blacklists or heuristic rules, this model addresses adaptability, user education, and feedback as core system pillars. This integrated design offers a forward-looking alternative in an increasingly complex threat landscape.

3.7. Limitations of Current Solutions

Despite the availability of multiple security tools and detection mechanisms, many existing solutions face several critical limitations.

- **Reactive rather than proactive Detection:**Most current security tools rely on static blacklists or predefined rule-based methods, which only protect users from previously identified malicious URLs. Attackers frequently alter domains or use URL obfuscation techniques to bypass these detection systems [2], [14]. This issue is exacerbated by the rapid evolution of phishing tactics and the constant creation of new fraudulent domains [1].
- **High Resource Consumption and Performance Issues:** Real-time scanning can be resource-intensive, leading to slow browser speeds, increased CPU load, and excessive

battery drain on mobile devices. These performance issues discourage users from keeping security measures active, which paradoxically increases their risk of exposure [15].

- **Inadequate Adaptability to Emerging Threats:** Traditional security solutions often struggle to keep pace with fast-evolving cyberattack techniques. Attackers use methods like domain hopping, URL shortening, and encrypted payloads to bypass static detection methods. As a result, many detection tools fail to adapt to new attack patterns and can leave users exposed to emerging threats [16], [17].

3.8. Future Directions and Advancements

To address the limitations of current detection tools, future solutions should focus on AI-driven classification, real-time URL analysis, and mobile platform integration.

AI and Machine Learning Integration

AI and machine learning models have the potential to significantly improve malicious URL detection. These technologies can learn from large datasets and dynamically classify URLs based on real-time behaviors [7]. Integrating AI-driven models with heuristic analysis could reduce false positives and negatives while providing real-time protection against emerging threats [6]. Moreover, machine learning could enable continuous updates to detection models, allowing them to adapt to new attack patterns [8].

Mobile Platform Integration

As mobile devices account for an increasing share of global web traffic [18], integrating malicious URL detection into mobile operating systems and applications becomes essential. Developing lightweight, adaptive detection tools for mobile platforms will be critical for enhancing mobile security [12].

4. FUTURE INTEGRATION OF AI-BASED DETECTION

While the current system relies on dynamic blacklists for real-time URL filtering, future versions may benefit from the integration of artificial intelligence (AI) to improve detection accuracy and adaptability.

The technical report underlying this article highlights the growing importance of machine learning techniques—such as Convolutional Neural Networks (CNNs) and other intelligent classifiers—in recognizing malicious URL patterns, especially in phishing and clickjacking contexts. These techniques offer potential improvements over static detection by learning from evolving attack behaviors and enabling real-time classification.

Although a specific AI model has not yet been implemented, the foundational structure of the system—modular design, browser extension compatibility, and real-time response mechanisms—was built with future AI integration in mind. As research and development progresses, appropriate models may be introduced to support threat detection in more dynamic and adaptive ways.

Importantly, any future enhancements will remain aligned with the project's core goals: lightweight operation, user privacy, and browser-side responsiveness.

Future Evaluation Plan: In future iterations, the proposed system may be validated using publicly available malicious URL datasets such as PhishTank, OpenPhish, and URLNet. These datasets provide labeled examples of benign and malicious URLs suitable for training and testing. Standard evaluation metrics—such as Accuracy, Precision, Recall, F1-score, and ROC-AUC—will be used to assess model performance. If implemented as a browser extension, the system may also be tested using simulated user environments to measure real-time response, false positive rates, and resource overhead.

As this study presents a conceptual design, no empirical implementation or experimental results are included at this stage.

5. CONCLUSION

The proliferation of malicious URLs remains a critical cybersecurity concern, exacerbated by the growing sophistication of evasion techniques and the limitations of conventional detection systems. This paper presented a comprehensive review of established detection methodologies, including signature-based, heuristic, machine learning, and deep learning approaches, evaluating their effectiveness within real-time security contexts.

To address the shortcomings identified in existing browser-based solutions, the paper introduced a conceptual model for a lightweight, AI-enhanced browser extension that integrates real-time blacklist validation, adaptive detection mechanisms, and user-centered cybersecurity education. The proposed system is positioned as a significant advancement over current tools, particularly in its dual focus on technical precision and user empowerment.

In contrast to existing browser security extensions that rely primarily on static detection logic or heuristic rules, the proposed system introduces a conceptually novel framework that combines dynamic blacklist validation, planned AI-based classification, and a built-in user awareness module. This integrated approach not only enhances real-time detection but also empowers users through education and feedback. By emphasizing lightweight performance, modular adaptability, and privacy-conscious architecture, the proposed extension contributes to a forward-looking model for improving browser-level cybersecurity. Future work will focus on empirical validation using benchmark datasets and refining the AI component for deployment in real-world browsing environments.

ACKNOWLEDGEMENTS

The Vice Presidency funded this research/project for Graduate Studies, Research, and Business (GRB) at Dar Al-Hekma University, Jeddah. The authors, therefore, acknowledge GRB's technical and financial support with thanks.

REFERENCES

[1] Jain, A. Kumar, and B. B. Gupta, ‘A survey of phishing attack techniques, defence mechanisms and open research challenges: Enterprise Information Systems: Vol 16 , No 4 - Get Access’, 2021, doi: 10.1080/17517575.2021.1896786.

[2] N. Samarasinghe and M. Mannan, ‘On cloaking behaviors of malicious websites’, *Comput. Secur.*, vol. 101, p. 102114, Feb. 2021, doi: 10.1016/j.cose.2020.102114.

[3] ‘New details reveal how hackers hijacked 35 Google Chrome extensions’. Accessed: Feb. 14, 2025. [Online]. Available: https://www.bleepingcomputer.com/news/security/new-details-reveal-how-hackers-hijacked-35-google-chrome-extensions/

[4] 'PhishTank | Join the fight against phishing'. Accessed: Feb. 14, 2025. [Online]. Available: https://phishtank.org/
[5] F. Callegati and M. Ramilli, 'Frightened by Links', *IEEE Secur. Priv.*, vol. 7, no. 6, pp. 72–76, Nov. 2009, doi: 10.1109/MSP.2009.177.
[6] R. S. Rao and A. R. Pais, 'Detection of phishing websites using an efficient feature-based machine learning framework', *Neural Comput. Appl.*, vol. 31, no. 8, pp. 3851–3873, Aug. 2019, doi: 10.1007/s00521-017-3305-0.
[7] M. Alsaedi, F. A. Ghaleb, F. Saeed, J. Ahmad, and M. Alasli, 'Cyber Threat Intelligence-Based Malicious URL Detection Model Using Ensemble Learning', *Sensors*, vol. 22, no. 9, Art. no. 9, Jan. 2022, doi: 10.3390/s22093373.
[8] N. Z. Gorment, A. Selamat, L. K. Cheng, and O. Krejcar, 'Machine Learning Algorithm for Malware Detection: Taxonomy, Current Challenges, and Future Directions', *IEEE Access*, vol. 11, pp. 141045–141089, 2023, doi: 10.1109/ACCESS.2023.3256979.
[9] I. Jemal, M. A. Haddar, O. Cheikhrouhou, and A. Mahfoudhi, 'Performance evaluation of Convolutional Neural Network for web security', *Comput. Commun.*, vol. 175, pp. 58–67, Jul. 2021, doi: 10.1016/j.comcom.2021.04.029.
[10] I. Ozen, K. Subramani, P. Vadrevu, and R. Perdisci, 'SENet: Visual Detection of Online Social Engineering Attack Campaigns', Jan. 10, 2024, *arXiv*: arXiv:2401.05569. doi: 10.48550/arXiv.2401.05569.
[11] X. Zhan *et al.*, 'A Systematic Assessment on Android Third-Party Library Detection Tools', *IEEE Trans. Softw. Eng.*, vol. 48, no. 11, pp. 4249–4273, Nov. 2022, doi: 10.1109/TSE.2021.3115506.
[12] 'McAfee 2023 Consumer Mobile Threat Report | McAfee Blog'. Accessed: Feb. 14, 2025. [Online]. Available: https://www.mcafee.com/blogs/internet-security/mcafee-2023-consumer-mobile-threat-report/
[13] K. Chia, J. Lee, W. Wan, R. Chua, and H. Guo, 'MalAware: A Tool for Safe Internet Browsing', in *Proceedings of the 9th IRC Conference on Science, Engineering, and Technology*, J. Lu, H. Guo, I. McLoughlin, E. G. Chekole, U. Lakshmanan, W. Meng, P. C. Wang, and N. Heng Loong Wong, Eds., Singapore: Springer Nature, 2023, pp. 303–315. doi: 10.1007/978-981-99-8369-8_30.
[14] A. S. Rafsanjani, N. Binti Kamaruddin, M. Behjati, S. Aslam, A. Sarfaraz, and A. Amphawan, 'Enhancing Malicious URL Detection: A Novel Framework Leveraging Priority Coefficient and Feature Evaluation', *IEEE Access*, vol. 12, pp. 85001–85026, 2024, doi: 10.1109/ACCESS.2024.3412331.
[15] S. Kumi, C. Lim, and S.-G. Lee, 'Malicious URL Detection Based on Associative Classification', *Entropy Basel Switz.*, vol. 23, no. 2, p. 182, Jan. 2021, doi: 10.3390/e23020182.
[16] M. Bossetta, 'A simulated cyberattack on Twitter: Assessing partisan vulnerability to spear phishing and disinformation ahead of the 2018 U.S. midterm elections | First Monday', no. Volume 23, Number 12-3 December 2018, Jan. 2018, Accessed: Feb. 14, 2025. [Online]. Available: https://firstmonday.org/ojs/index.php/fm/article/view/9540
[17] T. Yamauchi, R. Orito, K. Ebisu, and M. Sato, 'Detecting Unintended Redirects to Malicious Websites on Android Devices Based on URL-Switching Interval', *IEEE Access*, vol. 12, pp. 153285–153294, 2024, doi: 10.1109/ACCESS.2024.3478748.
[18] S. Gill, 'Internet Traffic from Mobile Devices Stats 2025', Priori Data. Accessed: Feb. 14, 2025. [Online]. Available: https://prioridata.com/data/mobile-internet-traffic/

AUTHORS

Homam El-Taj is an Assistant Professor at Dar Al-Hekma University in Saudi Arabia, where he explores the dynamic realm of cybersecurity. His research is fueled by a dedication to safeguarding digital landscapes, focusing on network security, cyber threat intelligence, and incident response. Homam is particularly captivated by the complexities of cryptography, cloud security, and IoT security, where he works to identify vulnerabilities and devise resilient solutions. His proactive approach to mitigating threats positions him as a leading figure in the ever-evolving field of cybersecurity.

Beyond his academic pursuits, Homam is known for his ability to bridge the gap between theory and practice. He actively engages with industry professionals, sharing insights that help organizations fortify

their defences against cyber threats. His work often highlights the importance of staying ahead of emerging technologies and the need for continuous learning in the face of rapidly changing security challenges.
When he's not immersed in research or teaching, Homam enjoys mentoring the next generation of cybersecurity experts, fostering a community of innovation and collaboration. His commitment to both academic excellence and practical application makes him a valuable asset in the ongoing battle to secure our digital world.

Mariah Ajaj dedicated and motivated senior student at Dar Al-Hekmah university studying bachelor cybersecurity, computer forensics, with hands-on experience in vulnerability assessment and digital forensics. Passionate about enhancing digital security and committed to making an impact in the cybersecurity field.