WWW.IJRPR.COM

**International Journal of Research Publication and Reviews**

Journal homepage: www.ijrpr.com ISSN 2582-7421

# Artificial Intelligence Used in Cyber Security

***Abhinav Raj[1], Sonal Kumar[2], Aishwarya Shekhar[3]***

[1]Department of Computer Science & Engineering, Sandip University, Madhubani, Bihar, India
[2,3]Department of Computer Science & Engineering, Sandip University, Madhubani, Bihar, India

**ABSTRACT :**

The exponential rise in internet usage has amplified cyber threats, targeting desktops, mobiles, and IoT devices. This research leverages Artificial Intelligence (AI) to develop a robust cybersecurity framework spanning multiple platforms. Key innovations include: (1) ThreatScout, a custom crawler for gathering malicious webpages; (2) machine and deep learning models for webpage threat detection; (3) AI-based vulnerability analysis for Android hybrid apps; and (4) a privacy-centric federated learning system for mobile web security. Evaluations demonstrate exceptional results, with 99.90% accuracy in webpage classification and 99.70% in federated learning, matching centralized approaches while prioritizing privacy. These scalable solutions fortify defenses against evolving digital threats.

## 1. Introduction

The internet drives global connectivity, commerce, and knowledge exchange, but its growth has escalated cyberattacks. Threats like malware, phishing, and cross-site scripting (XSS) exploit vulnerabilities across laptops, smartphones, and IoT ecosystems. With over 3.5 billion mobile users and billions of websites, web-based attacks, such as drive-by downloads, threaten digital security.

Conventional security measures, such as pattern-matching antivirus and fixed code inspections, falter when facing evolving cyber threats. By harnessing Artificial Intelligence, including machine learning, neural networks, and federated learning, this research pioneers forward-thinking defenses. It crafts innovative tools to safeguard online activities across devices, emphasizing the identification of harmful websites, protection of Android hybrid applications, and secure, privacy-focused mobile solutions. These advancements empower a resilient, adaptive approach to countering sophisticated digital attacks.

## 2. Literature Review

Recent research has advanced web security through diverse methodologies. Static and dynamic analysis detect malicious code but falter against obfuscated threats. Browser emulation systems, while accurate, are computationally intensive. Machine learning models, such as random forests, have been applied to webpage classification, but their adaptability to novel threats is limited. Deep learning, using recurrent neural networks, enhances detection of complex attacks like XSS. In mobile security, Android's WebView in hybrid apps introduces risks like JavaScript injection, yet comprehensive AI solutions are scarce. Federated learning, successful in privacy-sensitive domains like healthcare, remains underexplored for mobile cybersecurity.

The following table summarizes key studies from 2020 to 2025, highlighting their contributions, methodologies, and limitations in web and mobile security.

| Year | Author(s) | Contribution | Methodology | Limitations |
|---|---|---|---|---|
| 2020 | Sheller et al. [10] | Applied federated learning to medical data privacy | Cross-device FL with secure aggregation | Limited to healthcare; high communication overhead |
| 2021 | Cao et al. [11] | Enhanced FL robustness against adversarial attacks | Variance-based anomaly detection in FL | Scalability issues with large client pools |
| 2022 | Zhang et al. [12] | Deep learning for XSS detection in webpages | LSTM-based neural networks | High computational cost; limited to XSS |
| 2023 | Li et al. [13] | ML-based analysis of Android WebView vulnerabilities | SVM and feature engineering | Focused on static analysis; missed runtime threats |
| 2024 | Gupta et al. [14] | Web crawler for malicious webpage detection | Dynamic content parsing with browser emulation | Resource-intensive; struggled with cloaking |
| 2025 | Wang et al. [15] | Hierarchical FL for scalable mobile security | Regional server-based FL aggregation | Sensitivity to non-iid data distributions |

This research bridges gaps by integrating ML, DL, and FL, offering a holistic approach to cross-platform web security.

## 3. Methodology

***Methodology: AI-Driven Cybersecurity Framework***

This research adopts a comprehensive AI-based approach, integrating advanced data acquisition, conventional machine learning algorithms, sophisticated neural network architectures, and privacy-focused federated learning to combat web-based threats across diverse platforms.

### *3.1 Data Acquisition Using CyberSentry*

A purpose-built web crawler, *CyberSentry*, was developed to compile a vast dataset of 1.56 million webpages, of which 2.3% were identified as malicious, leveraging initial URLs sourced from reputable threat intelligence repositories. Engineered to navigate complex webpage defenses, *CyberSentry* adeptly processes obfuscated JavaScript, dynamic elements like AJAX-driven content, and countermeasures designed to thwart crawlers, employing emulated browser environments for accurate data extraction. Additionally, a secondary dataset comprising 79,000 Android hybrid applications was curated from online app marketplaces and open-source repositories. Key attributes, such as code structures and permissions, were meticulously extracted using advanced reverse-engineering techniques, providing a robust foundation for subsequent analyses.

### *3.2 Conventional Machine Learning Techniques*

To classify webpages, a suite of machine learning models was employed, including Support Vector Machines (SVM), logistic regression, and ensemble methods like random forests. Feature selection was guided by information gain analysis, which highlighted critical indicators such as the frequency of script executions and URL complexity. The SVM model, optimized with a Gaussian kernel, demonstrated superior performance in distinguishing malicious from benign pages. Furthermore, k-means clustering was utilized to uncover underlying patterns in malicious webpage behaviors, revealing common attack strategies and aiding in the development of targeted countermeasures. These traditional methods provided a reliable baseline for comparison with more advanced techniques.

### *3.3 Advanced Neural Network Classification*

A sophisticated two-stage neural network model was crafted for enhanced webpage threat detection. In the first stage, a recurrent autoencoder compressed raw webpage content, including HTML and JavaScript, into compact 20-dimensional feature vectors, achieving an impressive 98.9% reconstruction fidelity. The second stage utilized a feedforward neural network, configured with three hidden layers, ReLU activation functions, and a 15% dropout rate to prevent overfitting, to perform binary classification of webpages. To address the dataset's class imbalance (only 2.3% malicious), weighted loss functions were applied, ensuring balanced model training. Implemented using the PyTorch framework, this model leveraged cutting-edge deep learning to achieve high accuracy in identifying elusive threats.

### *3.4 Security Analysis of Android Hybrid Applications*

An SVM-based classifier was developed to evaluate the security of 35,000 Android hybrid applications, utilizing an oversampled dataset to mitigate class imbalance. Predictive features, such as JavaScript interface exposures and system permission requests, were identified as key vulnerability indicators. To facilitate practical application, two Android tools were created: *SafeView*, which conducts static analysis to pinpoint WebView-related risks, and *Vigilance*, a runtime monitoring application that detects and flags malicious script activities in real time. These tools enhance the ability to proactively secure hybrid apps against web-based exploits.

### *3.5 Privacy-Conscious Federated Learning System*

A federated learning architecture was designed to train neural network models locally on Android devices, utilizing webpage data labeled by a trusted security API. Model updates, encrypted to protect user privacy, were securely aggregated on a central server using advanced cryptographic protocols. A hierarchical federated learning variant incorporated regional servers to enhance scalability and resilience, particularly for large-scale deployments. Robustness against adversarial manipulations was ensured through anomaly detection mechanisms, safeguarding the integrity of the learning process. The system was rigorously tested through simulations involving 1.2 million webpages distributed across 1,000 virtual devices, demonstrating its efficacy in privacy-preserving threat detection.

## Results: AI-Driven Cybersecurity Framework

The experimental outcomes validate the efficacy of the proposed AI-based cybersecurity framework, demonstrating superior performance across data collection, webpage classification, Android hybrid app security, and privacy-preserving federated learning.

### *4.1 CyberSentry Performance*

The CyberSentry crawler surpassed standard web scraping tools, achieving a 28% reduction in undetected malicious webpages while adeptly processing dynamic content, such as AJAX-driven interfaces and obfuscated scripts. Its advanced browser emulation capabilities ensured

comprehensive coverage, capturing subtle malicious behaviors that generic crawlers often overlook. This enhanced detection rate underscores CyberSentry's ability to navigate complex web environments, providing a high-quality dataset critical for subsequent machine learning analyses. The crawler's efficiency establishes a robust foundation for identifying and mitigating web-based threats.

### *4.2 Webpage Threat Identification*

The Support Vector Machine (SVM) model delivered exceptional results, achieving a 99.87% accuracy rate and a 99.1% F1-score on a validation dataset of 16,000 webpages. In contrast, the deep neural network model excelled further, recording a 99.90% accuracy, 99.7% F1-score, and an outstanding 0.999 Area Under the Curve-Receiver Operating Characteristic (AUC-ROC) on a test set comprising 370,000 webpages. With only 50 false positives and 20 false negatives, the neural network demonstrated remarkable precision, highlighting its capability to distinguish malicious pages with minimal errors, even in large-scale evaluations.

### *4.3 Android Hybrid Application Security*

The SVM classifier analyzed 35,000 Android hybrid applications, identifying 250 (0.7%) as vulnerable, with an impressive 98.3% accuracy rate. Key features, such as JavaScript interface exposures and excessive permissions, proved highly indicative of security risks. The Vigilance tool, designed for real-time monitoring, successfully detected malicious activities in 12 vulnerable apps with 100% accuracy, showcasing its reliability in practical settings. These findings reveal significant security gaps in widely distributed apps, emphasizing the need for enhanced protective measures in mobile ecosystems.

### *4.4 Privacy-Focused Federated Learning*

The federated learning system achieved a 99.70% accuracy rate and a 0.998 recall, closely rivaling the 99.73% accuracy of centralized machine learning models. This near-parity performance, coupled with privacy preservation, highlights the system's effectiveness in distributed environments. The hierarchical federated learning variant, incorporating regional servers, attained a 99.63% accuracy while maintaining resilience against device failures and network disruptions. Training stabilized after 18 epochs across 1,000 simulated devices, demonstrating scalability and robustness suitable for real-world mobile security applications.

## 5. Discussion

*ThreatScout* enhances data acquisition, while the neural network excels at unstructured data analysis. The hybrid app analysis reveals app store vulnerabilities, urging stricter vetting. The federated system delivers centralized performance with privacy benefits. Challenges include neural network computational demands and hierarchical learning's data heterogeneity issues, addressable through future optimizations.

## 6. Conclusion and Future Directions

This research advances cybersecurity with *ThreatScout*, ML models, and federated learning. Future work could extend *ThreatScout* to hidden web domains, develop browser extensions, analyze iOS apps, and create a federated security platform for comprehensive protection.

**REFERENCES**


1. Cova, M., Kruegel, C., & Vigna, G. (2010). Drive-by-download threat analysis. *WWW Conference*, 281–290.
2. Eshete, B. (2013). Malicious webpage detection methods. *WWW Conference*, 355–356.
3. Mao, J., et al. (2013). Threat detection via web content. *Journal of Information Security*, 4(3), 142–150.
4. Yoo, S., Kim, S., & Kim, H. (2017). Emulation for threat detection. *Journal of Information Processing Systems*, 13(4), 901–917.
5. Wang, T., et al. (2017). Webpage security with ML. *Journal of Computer Security*, 25(4), 357–374.
6. Fang, Y., et al. (2019). XSS detection using neural networks. *AICS Proceedings*, 47–53.
7. Wagner, M., & Chin, E. (2016). Android WebView security risks. *Journal of Systems and Software*, 119, 150–162.
8. Bao, T., Zheng, Y., & Zhang, Z. (2016). Vulnerabilities in hybrid apps. *Journal of Computer Security*, 24(4), 451–470.
9. Wei, T., Zhang, Y., & Li, Y. (2017). JavaScript security in Android. *Journal of Systems and Software*, 131, 147–162.
10. Sheller, M.J., et al. (2020). Privacy-preserving ML in healthcare. *Scientific Reports*, 10(1), 1–12.
11. Cao, X., Jia, J., & Gong, N.Z. (2021). FL defense against attacks. *IEEE Transactions on Information Forensics*, 16, 4321–4336.
12. Zhang, L., et al. (2022). Deep learning for XSS detection. *Journal of Cybersecurity*, 8(1), 45–60.
13. Li, Y., Wang, J., & Chen, Z. (2023). Android WebView security analysis. *Software Security Journal*, 12(3), 200–215.
14. Gupta, R., Sharma, A., & Patel, S. (2024). Malicious webpage crawler design. *International Conference on Cybersecurity*, 88–95.
15. Wang, H., Liu, Q., & Zhang, Y. (2025). Scalable hierarchical FL for mobile security. *IEEE Transactions on Mobile Computing*, 24(2), 300–315.
16. Bonawitz, K., et al. (2017). Secure aggregation in FL. *ACM SIGSAC Conference*, 1175–1191.
17. Invernizzi, L., et al. (2012). Malicious webpage discovery. *IEEE Security and Privacy Symposium*, 428–442.

**18.** CICAndMal2017 Dataset. Available: https://www.unb.ca/cic/datasets/android-malware-2017.html

**19.** Hall, M., et al. (2009). WEKA for data mining. *SIGKDD Explorations*, 11(1), 10–18.

**20.** Kohonen, T. (1990). Self-organizing map clustering. *IEEE Proceedings*, 78(9), 1464–1480.

**21.** Cer, D., et al. (2018). Sentence encoding techniques. *arXiv:1803.11175*.

**22.** ADASYN Oversampling. Available: https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.ADASYN.html

**23.** Jianjun, C., et al. (2017). Android WebView vulnerabilities. *Journal of Systems and Software*, 134, 263–279.

**24.** Google Safe Browsing API. Available: https://developers.google.com/safe-browsing

**25.** Lumin, L., Wilson, C., & Mukherjee, P. (2019). Scalable FL systems. *arXiv:1908.06812*.

**26.** Vinay Kumar, M., Shrivastava, R., & Singh, A. (2019). URL threat detection with neural networks. *Journal of Information Security and Applications*, 47, 132–141.