A Review of Machine Learning Methods for Detecting DGA Botnets

|  |  |  |
| --- | --- | --- |
|  |  |  |
| line 1: Amir Kurbanov line 2: *Department of Cybersecurity* line 3: *UNITEN*  line 4: Putrajaya, Malaysia line 5:  GEB01085858@student.uniten.edu.my | line 1: Arslan Shushayev  line 2: *Department of Cybersecurity* line 3: *UNITEN*  line 4: Putrajaya, Malaysia line 5:  GEB01085859@student.uniten.edu.my | line 1: Aiya Kurzhangulova  line 2: *Department of Cybersecurity* line 3: *UNITEN*  line 4: Putrajaya, Malaysia line 5:  GEB01085857@student.uniten.edu.my |

***Abstract* —** ***Domain Generation Algorithm (DGA) botnets pose a significant cybersecurity threat by generating vast numbers of domain names to evade detection, enabling malicious activities like data exfiltration and distributed denial-of-service attacks. Traditional detection methods, such as signature-based analysis and DNS filtering, struggle to keep pace with evolving DGA techniques. Machine learning (ML) offers a promising alternative by identifying patterns in domain names and network traffic. This paper reviews current strategies for DGA botnet detection, focusing on ML approaches. Six ML models—Random Forest, Support Vector Machine, Convolutional Neural Network, Long Short-Term Memory, Decision Tree, and Naive Bayes—are analyzed for their principles, strengths, and limitations. A comparative table evaluates their accuracy, training time, and resilience to novel DGA domains. Challenges, including data quality, overfitting, and model interpretability, are discussed. Findings suggest that while ML enhances detection capabilities, limitations in real-world deployment necessitate further research. This review provides insights into the practical application of ML for combating DGA botnets, emphasizing ethical hacking perspectives.***

**Keywords —** ***DGA botnets, machine learning, cybersecurity, botnet detection, ethical hacking***

# I. INTRODUCTION

Domain Generation Algorithm (DGA) botnets represent a sophisticated class of malware that leverages algorithms to generate thousands of domain names daily, enabling communication with command-and-control (C2) servers while evading traditional detection. Unlike static malicious domains, DGA botnets dynamically create pseudo-random domain names, such as “x7b9k2p.com” or “qwe12zxc.net,” making it challenging to block them manually or with static rules. These botnets facilitate crimes like ransomware, data theft, and distributed denial-of-service (DDoS) attacks, impacting organizations globally. For instance, the 2021 Mirai botnet variant used DGA to orchestrate large-scale IoT attacks, highlighting their real-world threat [1].

|  |
| --- |
|  |

The operation of DGA botnets involves infected devices running an algorithm seeded with parameters like date or random numbers, producing domain names that the bot queries via DNS to locate C2 servers. Only a subset of these domains is registered by attackers, reducing the chance of detection. The sheer volume and randomness of generated domains overwhelm traditional defenses, as blacklists cannot scale to cover millions of potential domains.

Detection is further complicated by the adaptability of DGAs, which evolve to produce human-like domains or mimic legitimate traffic. Ethical hacking demands proactive strategies to counter these threats, as reactive measures fail to address the dynamic nature of DGAs. Machine learning (ML) has emerged as a viable solution, leveraging pattern recognition to identify malicious domains without relying on static signatures. This paper explores traditional and MLbased detection strategies, evaluates six ML models, and discusses their practical challenges in real-world environments, providing a foundation for advancing ethical hacking techniques.

# II. CURRENT STRATEGIES FOR DGA BOTNET DETECTION

Traditional methods for detecting DGA botnets include signature-based analysis, rule-based filtering, and DNS traffic analysis. Each approach has distinct advantages and limitations.

Signature-based analysis relies on predefined patterns of known DGA domains stored in databases. Tools like Snort use these signatures to flag malicious traffic. The strength lies in high accuracy for known threats, but the approach fails against new or mutated DGAs, as signatures require constant updates. A 2022 study noted that signature-based systems missed 40% of novel DGA domains in real-time tests [2].

Rule-based filtering applies heuristic rules to identify suspicious domain characteristics, such as high entropy or unusual character combinations (e.g., “x9z2k”). Systems like Bro implement these rules to block domains. This method is lightweight and fast but struggles with false positives, as legitimate domains like CDNs may share similar traits. Additionally, modern DGAs generate humanreadable domains, reducing rule effectiveness [3].

DNS traffic analysis has proven effective in detecting botnet behavior by examining abnormal patterns in domain queries.

For example, a study by Truong et al. analyzed monitored DNS traffic to identify infected hosts based on periodic query intervals and statistical features. Their approach combined behavioral DNS features with machine learning classifiers such as J48 and Random Forest, achieving up to 98.5% detection accuracy for DGA-based botnets in controlled environments [4]. This approach excels in identifying network-level anomalies but requires significant computational resources and may miss low-volume DGA activity. Hybrid systems combining these methods improve detection but remain reactive, unable to anticipate new DGA variants.

In summary, traditional strategies are effective against known threats but lack adaptability to the evolving DGA landscape, necessitating advanced approaches like ML..

# III. MACHINE LEARNING IN DGA BOTNET DETECTION

Machine learning enhances DGA botnet detection by analyzing domain name features and network traffic patterns without relying on static rules. ML models are trained on datasets containing legitimate and DGA-generated domains, extracting features like character frequency, n-gram distributions, entropy, and length. Network-based features, such as DNS query frequency or IP diversity, further improve detection. These models classify domains as benign or malicious, adapting to new patterns through retraining.

Supervised learning dominates DGA detection, using labeled datasets like the DGArchive for training. For instance, Random Forest models analyze domain entropy to achieve over 90% accuracy in controlled settings [5]. Unsupervised learning, such as clustering, identifies anomalies in unlabeled DNS traffic, useful for detecting unknown DGAs. Deep learning models, like Convolutional Neural Networks (CNNs), process raw domain strings as sequences, capturing subtle patterns missed by traditional methods.

A 2022 study demonstrated ML’s effectiveness in detecting the Ramnit botnet, where a Long Short-Term Memory (LSTM) model identified DGA domains with 93% precision by analyzing temporal query patterns [6]. However, ML requires high-quality datasets and computational resources, and models may struggle with adversarial DGAs designed to mimic legitimate domains. Despite these challenges, ML’s ability to generalize across DGA families makes it a cornerstone of modern ethical hacking strategies.

IV. MACHINE LEARNING MODELS FOR DGA DETECTION

Six ML models commonly used for DGA botnet detection are evaluated below, covering their principles, strengths, and limitations.

## A. Random Forest

Random Forest builds multiple decision trees, aggregating their outputs to classify domains. Features like entropy and n-grams are fed into the model. Its strength lies in high accuracy (up to 92% in 2021 tests) and robustness to noisy data [5]. However, training time increases with dataset size, and the model struggles with adversarial DGAs mimicking legitimate patterns.

## B. Support Vector Machine (SVM)

SVM maps domain features into a high-dimensional space, finding a hyperplane to separate malicious and benign domains. It excels in small datasets, achieving 90% accuracy in 2022 experiments [5]. However, SVM scales poorly with large datasets, and kernel selection impacts performance, requiring expert tuning.

## C. Convolutional Neural Network (CNN)

CNN treats domain names as image-like sequences, applying convolutional layers to extract spatial patterns. A 2023 study reported 94% accuracy on diverse DGA families [7]. CNNs handle raw data effectively but demand significant computational resources and large datasets, limiting real-time use.

## D. Long Short-Term Memory (LSTM)

LSTM, a recurrent neural network, models temporal dependencies in DNS query sequences and has been shown effective in detecting DGA domains with precision up to 93% in studies up to 2020 [6]. Its strength is capturing sequential patterns, but training is slow, and overfitting occurs with imbalanced datasets.

## E. Decision Tree

Decision Tree splits domain features into branches based on thresholds, classifying domains at leaf nodes. Its simplicity enables fast training and interpretability, with 90.31% accuracy in detecting harmful domains [8]. However, it overfits noisy data and struggles with complex DGA patterns. *F. Naïve Bayes*

Naive Bayes calculates probabilities of domain features belonging to malicious or benign classes, assuming feature independence. It is computationally efficient and suitable for real-time detection. A 2023 study demonstrated Naive Bayes achieving up to 97.5% accuracy in detecting algorithmically generated domains. However, the independence assumption can reduce accuracy when features are correlated [9].

The table below compares these models based on accuracy, training time, and resilience to new DGA domains.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Accuracy (%)** | **Training Time** | **Resilience to New**  **DGAs** |
| Random  Forest | 92 | Moderate | High |
| SVM | 90 | High | Moderate |
| CNN | 94 | Very High | High |
| LSTM | 93 | Very High | High |
| Decision Tree | 88 | Low | Low |
| Naive Bayes | 97.5 | Low | Moderate |

V. DISCUSSION AND ANALYSIS Machine learning significantly improves DGA botnet detection, but real-world challenges persist. Data quality is a primary concern, as datasets like DGArchive may lack diversity or contain outdated DGA families, reducing model generalizability. A 2023 study noted that models trained on synthetic data missed 30% of real-world DGAs [9]. Imbalanced datasets, with fewer malicious samples, lead to biased models favoring benign classifications.

Overfitting is another issue, particularly for deep learning models like CNNs and LSTMs, which memorize training data rather than generalizing. Regularization techniques, such as dropout, mitigate this but increase training complexity. Adversarial DGAs, designed to mimic legitimate domains, further challenge ML models. Adversarial samples have been shown to reduce the accuracy of LSTM models in detecting malicious patterns, demonstrating vulnerabilities of deep learning systems to such attacks. This highlights the importance of developing robust defense mechanisms against adversarial inputs [10].

Explainability remains a critical limitation, especially in ethical hacking, where investigators must justify findings in legal contexts. Models like Random Forest and Decision Tree offer interpretability, but complex models like CNNs operate as “black boxes,” hindering trust. Techniques such as SHAP (SHapley Additive exPlanations) are being explored to improve explainability, though widespread adoption is still limited [11].

Scalability and real-time deployment also pose challenges. High-accuracy models like LSTMs require significant computational resources, impractical for resourceconstrained environments like IoT networks. Lightweight models like Naive Bayes sacrifice accuracy for speed, highlighting the trade-off between performance and practicality.

In summary, while ML advances DGA detection, data limitations, overfitting, and explainability issues necessitate careful model selection and ongoing research to ensure robust real-world performance.

# VI. CONCLUSION

Machine learning offers a powerful approach to detecting DGA botnets, surpassing traditional methods by adapting to evolving threats. Models like Random Forest, CNN, and LSTM achieve high accuracy, leveraging domain and network features to identify malicious patterns. However, challenges such as data quality, overfitting, and model explainability limit their real-world efficacy. Lightweight models like Naive Bayes enable fast deployment but sacrifice precision, while complex models like LSTMs demand significant resources. The comparative analysis highlights the need for balanced solutions that prioritize accuracy, speed, and resilience to new DGA variants. Future research should focus on improving dataset diversity, developing interpretable models, and addressing adversarial techniques. By integrating ML with ethical hacking practices, robust defenses against DGA botnets can be built, enhancing cybersecurity. Despite current limitations, ML’s adaptability positions it as a cornerstone for combating sophisticated botnet threats, provided ongoing refinements address practical deployment challenges.

# REFERENCES

1. Akamai, "What Are Domain Generation Algorithms?", Akamai

Technologies. [Online]. Available:

<https://www.akamai.com/glossary/what-are-dgas>

1. M. S. Islam, M. M. Rahman, and M. A. Rahman, "Detection of DGAGenerated Domain Names with TF-IDF," *Electronics*, vol. 11, no. 3, p. 414, 2022.
2. nergyLogserver, "Detection of Domain Generation Algorithms (DGA) Using Entropy and Artificial Intelligence," 2024
3. D.-T. Truong, G. Cheng, A. Jakalan, X.-J. Guo, and A.-P. Zhou, “Detecting DGA-Based Botnet with DNS Traffic Analysis in Monitored Network,” *Journal of Internet Technology*, vol. 17, no. 2, pp. 217–230, Mar. 2016.
4. X. D. Hoang and X. H. Vu, "An Improved Model for Detecting DGA Botnets Using Random Forest Algorithm," *Information Security Journal: A Global Perspective*, vol. 31, no. 4, pp. 441–450, 2022.
5. H. Yu, X. Jiang, and H. Zhang, “DGA Domain Name Classification

Method Based on Long Short-Term Memory with Attention

Mechanism,” *Applied Sciences*, vol. 9, no. 20, p. 4205, Oct. 2019. Available: https://www.mdpi.com/2076-3417/9/20/4205

1. R. L. La O, C. A. Catania, and T. Parlanti, “LLMs for Domain

Generation Algorithm Detection,” *Journal of Information Security and Applications*, vol. 80, p. 103060, Oct. 2024. Available: <https://arxiv.org/abs/2411.03307>

1. Zhao et al., “Malicious Domain Name Detection Using ML Algorithms,” *International Journal of Advanced Computer Science and Applications*, vol. 16, no. 3, pp. 48–55, 2025. Available: [https://thesai.org/Downloads/Volume16No3/Paper\_48-](https://thesai.org/Downloads/Volume16No3/Paper_48-Malicious_Domain_Name_Detection_Using_ML_Algorithms.pdf)

[Malicious\_Domain\_Name\_Detection\_Using\_ML\_Algorithms.pdf](https://thesai.org/Downloads/Volume16No3/Paper_48-Malicious_Domain_Name_Detection_Using_ML_Algorithms.pdf)

1. S. M. Abdullahi, A. Mohammed, R. Y. Ibrahim, and A. Shamsuddeen, “Detection of Algorithmically Generated Domain Names using Ensemble Machine Learning Technique,” *Proceedings of the Cyber Secure Nigeria Conference*, Nigerian Army Resource Centre (NARC), Abuja, Nigeria, 11–12 July 2023, pp. 27–34. Available: [https://cybersecurenigeria.org/conference-proceedings/volume-2-](https://cybersecurenigeria.org/conference-proceedings/volume-2-2023/)

[2023/](https://cybersecurenigeria.org/conference-proceedings/volume-2-2023/)

1. N. Papernot, P. McDaniel, and I. Goodfellow, “Practical Black-Box Attacks against Machine Learning,” *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, 2017. Available[: https://doi.org/10.1145/3052973.3053009](https://doi.org/10.1145/3052973.3053009)
2. S. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” *Advances in Neural Information Processing Systems*, vol.

30, 2017. Available[: https://arxiv.org/abs/1705.07874](https://arxiv.org/abs/1705.07874)