Flow-Based Botnet Detection Using Ensemble Machine Learning on the CTU-13 Dataset

Harshith Siddartha Kutumbaka
Department of Computer Science
University of North Carolina at Charlotte
Charlotte, NC, USA
Email: your-email@uncc.edu

***Abstract*—Botnets remain a major cybersecurity threat due to their ability to launch large-scale attacks such as distributed denial-of-service, spam distribution, and data exfiltration. As network traffic becomes increasingly encrypted, traditional payload-based detection techniques are no longer sufficient. This paper presents a flow-based botnet detection framework using ensemble machine learning on the CTU-13 dataset. Network flow features are extracted, preprocessed, and used to train Random Forest and Gradient Boosting classifiers under highly imbalanced class conditions. Experimental results demonstrate that the Random Forest model achieves superior performance with an accuracy of 99.46%, a precision of 0.88, a recall of 0.73, an F1-score of 0.80, and a ROC-AUC of 0.995. Feature importance analysis further reveals that flow duration and byte-level characteristics are the most discriminative indicators of botnet activity. These results confirm the effectiveness of ensemble learning for scalable, interpretable, and high-accuracy botnet detection in modern encrypted network environments.**

***Index Terms*—Botnet Detection, Network Security, Machine Learning, Random Forest, Gradient Boosting, CTU-13, Explainable AI**

I. Introduction

Botnets are networks of compromised machines that are remotely controlled by attackers to perform coordinated malicious activities such as distributed denial-of-service (DDoS) attacks, phishing campaigns, spam propagation, malware distribution, and data theft. With the rapid growth of Internet-connected devices and cloud infrastructures, botnets have evolved into highly scalable and stealthy platforms for cybercrime. Modern botnets increasingly utilize encryption and polymorphic communication strategies, making traditional signature-based intrusion detection systems ineffective.

To overcome these limitations, machine learning-based network intrusion detection systems have emerged as a promising alternative. In particular, flow-based detection methods analyze statistical characteristics of network connections without relying on packet payload inspection, making them well-suited for encrypted environments. By leveraging temporal and volumetric flow attributes, machine learning models can distinguish malicious traffic patterns from benign behavior with high accuracy.

This work presents a flow-based botnet detection framework using ensemble machine learning on the widely adopted CTU-13 dataset. Two ensemble classifiers, Random Forest and Gradient Boosting, are evaluated to determine their effectiveness in detecting botnet traffic under extreme class imbalance conditions. In addition to detection performance, this study emphasizes explainability through feature importance analysis, which is critical for deploying machine learning systems in operational security environments.

The main contributions of this paper are summarized as follows:

- A large-scale experimental evaluation of ensemble machine learning models on real-world botnet traffic from the CTU-13 dataset.
- A detailed comparative analysis between Random Forest and Gradient Boosting classifiers using accuracy, precision, recall, F1-score, and ROC-AUC.
- An explainable detection framework based on feature importance ranking, highlighting which flow-level features contribute most to botnet detection.
- A discussion of a realistic network threat model and directions toward deep-learning-based temporal modeling.

II. Related Work

Machine learning has been widely explored for intrusion detection and botnet detection [4], [5], [15]. Flow-based approaches that operate on statistical features instead of payloads are particularly attractive in encrypted environments [6], [10]. Ensemble methods such as Random Forests and boosting algorithms have demonstrated strong performance due to their robustness and ability to model complex decision boundaries [2], [3]. The CTU-13 dataset [1] has become a standard benchmark for comparing botnet detection methods. Recent work has also emphasized explainable AI [7] and handling class imbalance [11] in security applications, as well as deep-learning-based intrusion detection using recurrent and convolutional architectures [9], [13].

III. Dataset Description

The CTU-13 dataset [1] is a publicly available benchmark consisting of thirteen different malware capture scenarios that combine botnet traffic with normal background traffic and unknown traffic. In this work, a single capture containing approximately 2.8 million network flows is used. Each flow aggregates packets belonging to the same communication and is labeled as either benign or botnet traffic.

The dataset is highly imbalanced: the majority of flows correspond to benign activity, while only a small fraction corresponds to botnet command-and-control communications. This class imbalance makes the detection task challenging and motivates the use of ensemble classifiers that can better handle skewed distributions.

IV. Preprocessing and Feature Engineering

Raw CTU-13 flow records contain a mixture of numeric and categorical fields, including IP addresses, ports, protocol identifiers, byte and packet counts, and timing information. In this work, all non-numeric attributes (e.g., IP addresses) are discarded, and only numeric flow-level features are retained.

The preprocessing pipeline performs the following steps:

- Column names are normalized to lower case and stripped of whitespace.
- The label field is converted into a binary indicator where 1 represents botnet traffic and 0 represents benign traffic.
- Infinite values are replaced with NaNs, and all missing values are imputed with zero.
- Only numeric columns, including flow duration (`dur`), total bytes (`totbytes`), source bytes (`srcbytes`), total packets (`totpkts`), and header/state fields (e.g., `dtos`, `stos`), are retained.

The final dataset is split into 80% training and 20% testing using stratified sampling to preserve the class distribution.

V. Machine Learning Models

A. Random Forest

Random Forest [2] is an ensemble learning method composed of multiple decision trees trained using bootstrap aggregation (bagging). Each tree is trained on a random subset of the training data and a random subset of features. The final prediction is obtained by majority voting across trees. Random Forests are robust to overfitting, can model complex non-linear decision boundaries, and perform well on imbalanced datasets.

B. Gradient Boosting

Gradient Boosting [3] builds an ensemble of decision trees sequentially, where each new tree attempts to correct the residual errors of the previous trees. It optimizes a differentiable loss function using gradient descent in function space. Gradient Boosting often achieves strong performance on tabular data but can be more sensitive to noise and class imbalance than bagging-based methods.

VI. Experimental Setup

All experiments were conducted using Python 3.10 with the `pandas`, `numpy`, and `scikit-learn` libraries. The models were trained on an 80–20 stratified train-test split of the CTU-13 flow dataset.

The Random Forest classifier was configured with 200 trees, a maximum depth of 20, and all available CPU cores. Gradient Boosting was trained with 200 estimators, a learning rate of 0.1, and a maximum tree depth of 3. Due to the large dataset size, Gradient Boosting was trained on a stratified subsample of 300,000 flows, while Random Forest was trained on the full training set.

Performance was evaluated using accuracy, precision, recall, F1-score, and the area under the ROC curve (ROC-AUC). These metrics provide complementary views of classifier behavior under severe class imbalance.

VII. Results

TABLE I
Model Performance Comparison on CTU-13

Model	Accuracy	Precision	Recall	F1	ROC-AUC
Random Forest	0.9946	0.8800	0.7299	0.7979	0.9953
Gradient Boosting	0.9918	0.8428	0.5374	0.6563	0.9827

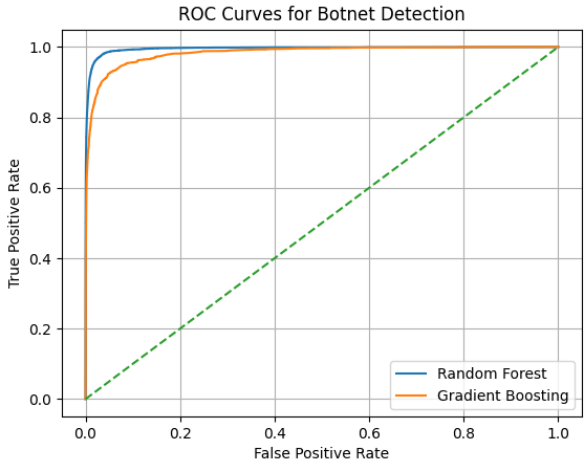

Fig. 1. ROC curves for Random Forest and Gradient Boosting classifiers on the CTU-13 dataset.

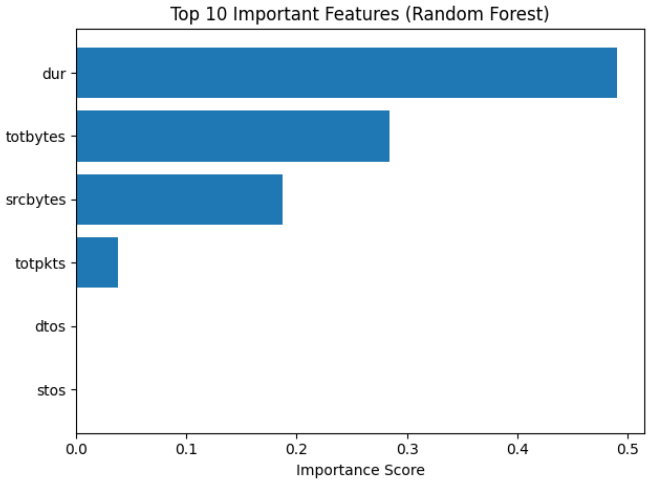

Fig. 2. Top 10 most important features according to the Random Forest classifier.

The Random Forest model clearly outperforms Gradient Boosting across all metrics. In particular, Random Forest achieves substantially higher recall and F1-score, which are critical for minimizing missed detections in security applications. The ROC curves in Fig. 1 show that Random Forest consistently dominates Gradient Boosting across all classification thresholds.

VIII. Feature Importance and Explainability

To better understand the decisions made by the Random Forest classifier, feature importance scores were computed based on the mean decrease in impurity. Fig. 2 shows the top ten most important features. Flow duration (`dur`) is the most influential feature, followed by total byte volume (`totbytes`) and source-side bytes (`srcbytes`). Total packet count (`totpkts`) and header/state features (e.g., `dtos`, `stos`) also contribute to the model's decisions.

These results indicate that botnet command-and-control flows differ from benign traffic primarily in how long connections persist and how much data they transfer, even without inspecting payload contents. This suggests that flow-based models can be effectively deployed in encrypted environments where deep packet inspection is not feasible.

IX. Threat Model

This work assumes a network-based adversary that controls one or more infected hosts participating in botnet command-and-control (C2) communication. The attacker is capable of generating encrypted and obfuscated traffic patterns in order to evade traditional signature-based intrusion detection systems. Botnet traffic may operate in low-and-slow modes, making it difficult to distinguish from benign background traffic using simple threshold-based rules.

The defender is assumed to have passive visibility at the network flow level but does not have access to packet payload contents due to encryption or privacy constraints. Detection is performed using statistical flow-level attributes such as duration, packet counts, and byte volume. The goal of the adversary is to maintain stealth while sustaining long-lived C2 channels and data exfiltration operations. The goal of the defender is to accurately identify malicious traffic with high recall while minimizing false positives. This threat model reflects realistic enterprise and ISP environments where encrypted traffic dominates.

X. Discussion

The experimental results demonstrate that Random Forest provides a robust and high-performing baseline for flow-based botnet detection on CTU-13. Its superior recall and F1-score compared to Gradient Boosting suggest that bagging-based ensembles are better suited to handling severe class imbalance and noisy network traffic in this setting.

The feature importance analysis offers interpretable evidence that long-lived and high-volume flows are strong indicators of botnet activity. These insights can assist security analysts in designing complementary rule-based detection strategies or prioritizing alerts.

XI. Limitations

This study has several limitations. First, it relies on a single CTU-13 capture scenario and does not evaluate generalization across all 13 scenarios or other datasets [14]. Second, only classical ensemble models were considered; more advanced architectures such as deep neural networks were not evaluated. Third, adversarial manipulation of flow features and concept drift in long-term deployments were not modeled.

XII. Future Work

Future work will extend the proposed framework in several directions. Deep learning models such as Long Short-Term Memory (LSTM) networks and Transformers will be used to model temporal dependencies across sequences of flows for each host, which may improve early-stage botnet detection [9]. Real-time deployment using streaming platforms and online learning algorithms will be explored to assess performance under realistic traffic loads. Finally, adversarial robustness and evasion-resistant feature engineering will be investigated, building on recent advances in adversarial machine learning [12].

XIII. Conclusion

This paper presented a flow-based botnet detection framework using ensemble machine learning models on the CTU-13 dataset. The Random Forest classifier achieved superior performance with 99.46% accuracy, 0.88 precision, 0.73 recall, 0.80 F1-score, and a ROC-AUC of 0.995, outperforming Gradient Boosting across all metrics. Feature importance analysis confirmed the effectiveness of traffic duration and byte-level attributes as discriminative indicators of botnet activity. The results demonstrate that ensemble tree-based methods provide a scalable and interpretable solution for encrypted botnet detection and form a strong foundation for future deep-learning-based extensions.

References

[1] S. Garcia, M. Grill, J. Stiborek, and A. Zunino, "An empirical comparison of botnet detection methods," *Computers & Security*, vol. 45, pp. 100–123, 2014.

[2] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[3] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[4] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *Proc. IEEE Symposium on Security and Privacy*, 2010.

[5] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cybersecurity intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[6] Y. Shapira et al., "Encrypted traffic classification using machine learning," in *Proc. IEEE INFOCOM*, 2019.

[7] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. NeurIPS*, 2017.

[8] M. Conti, A. Dehghantanha, K. Franke, and S. Watson, "Internet of Things security and forensics: Challenges and opportunities," *Future Generation Computer Systems*, vol. 78, pp. 544–546, 2018.

[9] Y. Kim, J. Kim, and H. Kim, "Long short-term memory recurrent neural network classifier for intrusion detection," in *Proc. IC-Platform*, 2016.

[10] T. T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 4, pp. 56–76, 2008.

[11] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.
[12] B. Biggio and F. Roli, "Wild patterns: Ten years after the rise of adversarial machine learning," *Pattern Recognition*, vol. 84, pp. 317–331, 2018.
[13] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," *EAI Endorsed Transactions on Security and Safety*, vol. 3, no. 9, 2016.
[14] I. Sharafaldin, A. Lashkari, and A. Ghorbani, "Toward a reliable intrusion detection benchmark dataset," *Software Networking*, vol. 2018, pp. 177–200, 2018.
[15] N. Moustafa and J. Slay, "The significant features of the UNSW-NB15 and the KDD99 data sets for network intrusion detection systems," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, 2015.