Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/18/20230967

Research on botnet detection technology in network security

Haodong Yu

Maumee Valley Country Day School, Toledo, 43614, U.S.

24hyu@mvcds.org

Abstract. In the current world of consistent cybersecurity threats, the priority of protecting precious data from malicious activities has never come this high. A network of infected computers that are under the control of bad actors is known as a botnet. These networks may be used for a variety of things, including spam distribution, distributed denial of service (DDoS) assaults, identity theft, and malware distribution. A botnet's constituent computers are frequently referred to as "bots" or "zombies.". And there have been appalling statistics of a 100% increase in DDoS attacks from 2021 to 2022, and attackers have been consistently evolving, implementing smaller, yet more persistent attacks. Fortunately, the measures for protecting computers from botnet attacks have also been evolving. The very first step to defending against botnet attacks is to spot suspicious requests, and in this paper, the machine learning method is utilized to help pinpoint the potential attacks. First, a comprehensive dataset is found and used to train the model. This is a dataset consisting of source IPs, protocols, bidirectional flows, packets and a total of 33 features of internet flows with a mix of normal and malicious internet flow. As for the models performed, random forest and logistic regression were chosen and run with 80 percent of the data from the dataset as a training set and 20 percent as a testing set. Overall, the two models perform greatly with the given dataset. It is a very basic study on the prevention of botnet detection, yet certainly, it provides insights and contributions into further developments in cybersecurity.

Keywords: botnet, random forest, logistic regression, internet flow

1. Introduction

1.1. Background

A botnet is a network of compromised computers under the control of malicious actors. These networks can be used for various purposes, including sending spam, launching distributed denial of service (DDoS) attacks, stealing identity, and spreading malware. The computers that make up a botnet are often called "bots" or "zombies". Detecting botnets is a difficult task as they operate covertly and are designed to evade detection. However, botnet detection is important to protect individuals and organizations from the harmful effects of these networks [1]. Botnets can cause significant financial loss, reputational damage, and compromise of confidential information. Multiple approaches to detecting botnets include signature-based detection, behavior-based detection, and anomaly detection. Signature-based detection looks for specific patterns or signatures associated with known botnet activity. Behavior-based detection analyzes network traffic behavior to identify activity consistent with botnet activity. Anomaly detection identifies deviations from normal network traffic patterns that may indicate botnet activity. Detecting

DOI: 10.54254/2755-2721/18/20230967

botnets is an ongoing challenge, as botnet authors constantly develop new techniques to evade detection [2]. However, continued research and development of detection methods can help reduce the risks associated with botnets and improve the security of computer networks.

1.2. Definition of botnet

A botnet is a network of infected computers controlled for malicious purposes by a central authority called a botmaster. Also known as zombies or bots, these infected computers can be used to carry out various types of cyber-attacks such as distributed denial-of-service (DDoS) attacks, spamming, phishing, and malware distribution.

1.3. Challenges in botnet detection

However, due to the sophisticated and ever-evolving nature of botnets, they can be difficult to detect. Botnets use techniques such as encryption, obfuscation, and polymorphism to evade detection and can operate on a global scale, making it difficult to track and identify their source. Additionally, botnets can be decentralized, making it difficult to identify botmasters or shut down entire networks [3].

2. Botnet detection techniques

There are several techniques that cybersecurity professionals can use to detect botnets, each with its own advantages and limitations

2.1. Signature-based detection

One of the most common botnet detection techniques is signature-based detection, which looks for specific code or behavioral patterns that are characteristic of known botnets. Signature-based detection is a reliable and efficient technique, but it can be easily bypassed by botnets using encryption or other obfuscation techniques.

2.2. Behavior-based detection

Another technique used to detect botnets is behavior-based detection, which monitors network traffic for unusual or suspicious behavior, such as heavy outbound traffic or communication with known botnet command and control servers. Behavior-based detection is effective at detecting new and unknown botnets, but it can also generate false positives when legitimate traffic is misclassified as botnet-related.

2.3. Machine learning

As an increasingly popular measure taken by professionals, machine learning algorithms can be trained to recognize botnet-related patterns and behaviors by analyzing large amounts of network traffic data. Hence signature-based detection and behavior-based detection have been built into the detection method [4]. Machine learning has the advantage of being adaptable to new and evolving botnets, but it can also be resource-intensive and require large amounts of training data.

3. Implementation of botnet detection methods

3.1. Evaluation metrics

Evaluation metrics are used to assess the performance of a botnet detection model. The choice of metrics depends on the specific objectives of the model and the nature of the data being analyzed [5]. The evaluation metrics for the botnet detection models discussed here include:

3.1.1. Detection accuracy

Precision measures the proportion of true positives among all the instances that the model classified as positive. A high precision indicates that the model has a low false positive rate and is accurately identifying botnet traffic.

DOI: 10.54254/2755-2721/18/20230967

3.1.2. Recall

Recall measures the proportion of true positives among all the instances of botnet traffic in the data set. A high recall indicates that the model is detecting most of the botnet traffic in the data set.

3.2. Logistic regression

Logistic regression is a statistical technique used to analyze the relationship between a binary dependent variable and one or more independent variables. It is commonly used in various fields, including finance, healthcare, marketing, and social sciences, to make predictions or classify observations based on certain characteristics [6].

The dependent variable in logistic regression is binary, meaning it takes on only two possible values, typically 0 or 1. Examples of binary dependent variables include whether a customer will purchase a product or not, whether a patient has a disease or not, or whether an individual will default on a loan or not.The independent variables, also known as predictors or covariates, can be either categorical or continuous. Categorical variables are variables that take on a limited number of discrete values, such as gender or marital status. Continuous variables, on the other hand, can take on a range of values, such as age or income. The goal of logistic regression is to estimate the probability of the dependent variable taking on a particular value, given the values of the independent variables. The logistic function, also known as the sigmoid function, is used to transform the linear combination of the independent variables into a probability value between 0 and 1. Logistic regression is a popular technique because it is relatively simple to implement, interpretable, and robust to outliers [7]. It also allows for the identification of important predictors and the assessment of their impact on the dependent variable.

3.2.1. Advantages

Logistic regression has several advantages that make it a popular technique for analyzing binary dependent variables. Some of the key advantages of logistic regression are

- Simplicity: Logistic regression is a relatively simple and easy-to-understand statistical technique, making it accessible to researchers and practitioners with limited statistical training. It can be easily implemented in most statistical software packages and can be interpreted without much difficulty.
- Interpretability: The coefficients of logistic regression models are easy to interpret, providing insight into the relationship between the independent variables and the probability of the dependent variable. The coefficients can be used to identify important predictors and assess their impact on the dependent variable.
- Flexibility: Logistic regression can be used with both categorical and continuous independent variables, providing flexibility in the types of data that can be analyzed. Additionally, logistic regression can be used to analyze multiple independent variables, allowing for the exploration of complex relationships between variables.
- Robustness: Logistic regression is robust to outliers and can handle missing data, making it a reliable technique for analyzing real-world data sets.
- Model validation: Logistic regression allows for the validation of the model using various techniques, such as cross-validation or goodness-of-fit tests, to assess the model's performance and generalizability.

In summary, logistic regression has several advantages, including simplicity, interpretability, flexibility, robustness, good performance, and model validation. These advantages make it a popular and reliable statistical technique for analyzing binary dependent variables.

3.2.2. Limitations

While logistic regression has several advantages, it also has some limitations that should be considered when using this statistical technique. Some of the key limitations of logistic regression are.

DOI: 10.54254/2755-2721/18/20230967

- Linearity assumption: Logistic regression assumes that the relationship between the independent variables and the dependent variable is linear. If the relationship is not linear, the model may not fit the data well, leading to biased or incorrect predictions.
- Independence assumption: Logistic regression assumes that the observations are independent of each other. If there is correlation or clustering among the observations, the model may not fit the data well, leading to biased or incorrect predictions.
- Limited outcome variable: Logistic regression is only applicable to binary dependent variables. If the dependent variable has more than two categories, alternative methods such as multinomial logistic regression or ordinal logistic regression may be more appropriate.
- Lack of robustness: While logistic regression is generally robust to outliers, extreme outliers can still have a significant impact on the model's predictions. Additionally, logistic regression may not perform well if the data is imbalanced or if there are rare events in the dependent variable.
- Overfitting: Logistic regression can be prone to overfitting if the model is too complex or if there are too many independent variables relative to the sample size. Overfitting can lead to a model that fits the training data well but does not generalize well to new data.
- Assumption of linearity in the logit: Logistic regression assumes that the relationship between the logit of the dependent variable and the independent variables is linear. If this assumption is violated, the model may not fit the data well, leading to biased or incorrect predictions.

In summary, while logistic regression has several advantages, it also has limitations related to its assumptions and the nature of the data being analyzed. These limitations should be carefully considered when choosing and interpreting logistic regression models.

3.3. Random forest

Random forest is a powerful machine learning algorithm that is widely used for classification, regression, and other types of data analysis. It is a type of ensemble learning algorithm that combines multiple decision trees to make predictions. Each decision tree in the random forest is built using a subset of the available data and a random selection of the independent variables, ensuring that the trees are diverse and not overly dependent on any one variable [8].

The random forest algorithm works by aggregating the predictions of the individual decision trees. For classification problems, the final prediction is typically made by taking a majority vote among the trees, while for regression problems, the final prediction is typically made by taking the mean or median of the predictions from the individual trees. By aggregating the results of multiple decision trees, the random forest can capture complex relationships between the independent and dependent variables, resulting in high accuracy even with noisy or imbalanced data. One of the key advantages of random forest is its ability to handle a wide variety of data types, including both categorical and continuous data. It is also robust to outliers and missing data, making it a reliable algorithm for real-world data sets [9]. In addition, the random forest provides a measure of feature importance, allowing users to identify the most important predictors for the dependent variable. This can be useful for feature selection, dimensionality reduction, and understanding the underlying relationships in the data. Random forest is a scalable algorithm that can handle large data sets with thousands of observations and hundreds of independent variables. It can also be easily parallelized, allowing for efficient computation on clusters or distributed systems. These advantages have made the random forest a popular and effective machine-learning algorithm for a wide range of applications, including predicting customer behavior, identifying fraud, and analyzing medical data [10].

3.3.1. Advantages

Random forest is a popular machine learning algorithm that is used for classification, regression, and other types of data analysis. Some of the key advantages of using random forest are.

- Accurate: Random forest can provide highly accurate predictions by aggregating the results of multiple decision trees. The algorithm can capture complex relationships between the

DOI: 10.54254/2755-2721/18/20230967

independent and dependent variables, resulting in high accuracy even with noisy or imbalanced data.

- Robust: Random forest is robust to outliers and missing data, making it a reliable algorithm for real-world data sets. It can handle a large number of independent variables, as well as both categorical and continuous data.
- Non-parametric: Random forest is a non-parametric algorithm, meaning that it does not make any assumptions about the underlying distribution of the data. This makes it flexible and able to handle a wide variety of data types.
- Feature importance: Random forest provides a measure of feature importance, allowing users to identify the most important predictors for the dependent variable. This can be useful for feature selection, dimensionality reduction, and understanding the underlying relationships in the data.
- Scalable: Random forest is a scalable algorithm that can handle large data sets with thousands of observations and hundreds of independent variables. It can also be easily parallelized, allowing for efficient computation on clusters or distributed systems.
- Low bias: Random forest has a low bias, meaning that it can capture complex relationships between the independent and dependent variables, even when they are nonlinear or involve interactions between multiple predictors.
- In summary, the random forest has several advantages, including high accuracy, robustness, non-parametric flexibility, feature importance, scalability, and low bias. These advantages make it a popular and effective machine learning algorithm for a wide range of applications, including classification, regression, and other types of data analysis.

3.3.2. Limitations

Despite its many advantages, there are also some limitations to using random forest. Some of the key limitations are.

- Interpretability: While random forest provides a measure of feature importance, it can be difficult to interpret the individual decision trees that make up the ensemble. This can make it challenging to understand the underlying relationships in the data and to communicate the results to stakeholders.
- Overfitting: Random forests can be prone to overfitting when the number of trees in the ensemble is too large or when the trees are too deep. This can lead to high variance and poor generalization performance on new data.
- Computationally intensive: Random forests can be computationally intensive, especially when the number of trees and the number of independent variables are large. This can make it difficult to train and deploy the model in real-time applications.
- Imbalanced data: Random forests can struggle with imbalanced data sets, where one class is much less common than the others. This is because the majority class can dominate the predictions, leading to poor performance of the minority class.
- Correlated features: Random forests can struggle with highly correlated features, as they can dominate the feature importance rankings and lead to overfitting. Feature selection techniques may be needed to address this issue.

In summary, the random forest has several limitations, including interpretability, overfitting, computational intensity, imbalanced data, and correlated features. These limitations should be carefully considered when choosing an appropriate machine-learning algorithm for a given application. However, despite these limitations, the random forest remains a powerful and widely used algorithm for classification, regression, and other types of data analysis.

DOI: 10.54254/2755-2721/18/20230967

4. Results

Table 1. Comparison of classification results.

Classifiers	Accuracy (%)	Precision (%)	Recall (%)
Logistic Regression	83.7	78.3	90
Random Forest	99.8	99.9	99.8

Table 2. Comparison of experimental results under different characteristics.

Features	Logistic Regression			Random Forest		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall
F1	77.6	71.2	92.1	84.3	76.4	99.4
F2	50.8	21.4	0.4	54.8	62.3	60
F3	68.5	59.7	88.5	72.6	66.3	92.6
F4	50.6	0	0	50.9	45.5	90
F1, F2	78.2	71.8	92.1	86.9	80	98.5
F1, F3	82.8	77.2	89.3	89.9	85.5	96.2
F1, F4	78.7	72.4	92.2	85.2	77.7	98.7
F2, F3	68.9	60	88.5	71.3	70.3	83.4
F2, F4	61.8	56.6	98.4	53.8	64.6	49.9
F3, F4	68.7	59.8	88.6	73.8	67.5	92.1
F2, F3, F4	69	60.1	88.6	75	68.3	93.7
F1, F3, F4	83.4	78	89.9	90.4	85.8	97
F1, F2, F4	79.1	72.9	92.2	87.2	80.2	99
F1, F2, F3	83.2	77.9	89.6	90.8	86.7	96.5

The results of the experiments are shown in Table 1. It clearly shows Random Forest model outperformed the logistic regression with an accuracy of ~99.6%. But in exchange, Random Forest took 30 minutes to perform, in contrast, less than 1 minute for logistic regression. Table 2 shows the results of the feature analysis. We observe that generic feature group F1 and subnet feature group F3 are contributing much more when compared to aggregate features F2 & periodic communication features F4. Also when periodic communication feature set F4 is fed alone into the logistic regression classifier, the precision, and recall are 0. This clearly indicates the feature set is not contributing to the overall classification. This might be due to the threshold of the standard deviation of IATs or the flow IATs themselves which was used as a periodic communication detector and needs to be analyzed further.

5. Conclusion

Botnet detection is a crucial aspect of maintaining cybersecurity, as botnets are a common tool used by attackers to carry out malicious activities such as Distributed Denial of Service (DDoS) attacks, spamming, and credential theft. Detecting and mitigating botnets involves analyzing network traffic, identifying anomalous behavior, and employing various techniques to block or limit communication between infected devices and their command and control servers. The above paper discussed ultilizing machine learning, using random forest and logistic regression to filter the potential malicious behavior from botnet detection, aiming to provide a proper study on the prevention of cyber attack under current trend of increasing threats of cyber security. Overall, botnet detection is a continuous process that requires constant monitoring and updating of security measures to stay ahead of evolving threats. By implementing a comprehensive security strategy that includes botnet detection and mitigation, better protection from malicious attacks can be hence achieved.

DOI: 10.54254/2755-2721/18/20230967

References

[1] Kaur N, Singh M. Botnet and botnet detection techniques in cyber realm[C]//2016 International Conference on Inventive Computation Technologies (ICICT). IEEE, 2016:1-7.

[2] Rajab MA, Zarfoss J, Monrose F, et al. Multifaceted approach to understanding the botnet phenomenon[J]. ACM Computing Surveys (CSUR), 2010, 43(3):16.

[3] Salloum SA, Alshurideh M, Elnagar A, et al. Machine learning and deep learning techniques for cybersecurity: a review[C]//Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020). Springer International Publishing, 2020:50-57.

[4] Shinan K, Alsubhi K, Alzahrani A, et al. Machine learning-based botnet detection in software-defined network: a systematic review[J]. Symmetry, 2021, 13(5):866.

[5] Muhammad A, Asad M, Javed AR. Robust early stage botnet detection using machine learning[C]//2020 International Conference on Cyber Warfare and Security (ICCWS). IEEE, 2020:1-6.

[6] Trajanovski T, Zhang N. An automated and comprehensive framework for IoT botnet detection and analysis (IoT-BDA)[J]. IEEE Access, 2021, 9:124360-124383.

[7] Capuano N, Fenza G, Loia V, et al. Explainable Artificial Intelligence in CyberSecurity: A Survey[J]. IEEE Access, 2022, 10:93575-93600.

[8] [8] Aloqaily M, Kanhere S, Bellavista P, et al. Special Issue on Cybersecurity Management in the Era of AI[J]. Journal of Network and Systems Management, 2022, 30(3):39.

[9] Wazzan M, Algazzawi D, Bamasaq O, et al. Internet of Things botnet detection approaches: Analysis and recommendations for future research[J]. Applied Sciences, 2021, 11(12):5713.

[10] Qiao H, Novikov B, Blech JO. Concept Drift Analysis by Dynamic Residual Projection for effectively Detecting Botnet Cyber-attacks in IoT scenarios[J]. IEEE Transactions on Industrial Informatics, 2021, 18(6):3692-3701.