Journal of Engineering, Mechanics and Modern Architecture
Vol. 2, No. 11, 2023 ISSN: 2181-4384

IMPROVING THE EFFICIENCY OF BRUTE-FORCE ATTACK DETECTION USING DECISION TREES: AN ANALYSIS STUDY

Annotation:

Brute-force attacks are a common type of cyber attack in which an attacker repeatedly tries to guess a user's password or other login credentials. These attacks can be very time-consuming, but they can also be very successful, especially if the attacker is able to guess the credentials correctly.

One way to improve the efficiency of brute-force attack detection is to use decision trees. Decision trees are a type of machine learning algorithm that can be used to classify data. In the context of brute-force attack detection, decision trees can be used to identify patterns in login attempts that are indicative of a brute-force attack.

This paper presents an analysis of the use of decision trees for improving the efficiency of brute-force attack detection. The paper first reviews the literature on brute-force attacks and decision trees. Then, the paper presents the results of an experimental study that compares the performance of a decision tree-based brute-force attack detection system to a traditional rule-based system.

The results of the experimental study show that the decision tree-based system is more efficient than the rule-based system. The decision tree-based system was able to detect brute-force attacks with a higher accuracy and a lower false positive rate.

The findings of this paper suggest that decision trees can be an effective tool for improving the efficiency of brute-force attack detection. Decision trees are easy to implement and can be used to detect a wide range of brute-force attack patterns.

Keywords: *brute-force attack, decision tree, machine learning, classification, security.*

Information about the authors

Nebras Jalel Ibrahim
Computer Center, University of Diyala

1. Introduction

In today's interconnected world, cybersecurity threats are constantly evolving, posing a significant risk to individuals, organizations, and critical infrastructure[1]. Among these threats, brute-force attacks stand as a prevalent and persistent menace. These attacks involve the systematic and repetitive guessing of login credentials, such as usernames and passwords, to gain unauthorized access to systems or accounts.

Brute-force attacks can be particularly damaging due to their simplicity and effectiveness[2]. By employing automated scripts or dedicated tools, attackers can generate a vast number of credential combinations, overwhelming security defenses and potentially breaching systems. This highlights the urgent need for robust and efficient methods to detect and prevent brute-force attacks.

Traditional approaches to brute-force attack detection often rely on predefined rules and thresholds[3]. While these methods can be effective in identifying obvious attack patterns, they may struggle to adapt to the evolving tactics and techniques employed by attackers. Additionally, rule-based systems can generate a high volume of false positives, unnecessarily burdening security personnel and increasing operational costs.

In addressing these challenges, machine learning algorithms have emerged as promising tools for enhancing brute-force attack detection[4]. Decision trees, a specific type of machine learning algorithm, offer several advantages in this context. Their ability to learn from data and identify complex patterns makes them well-suited for detecting subtle anomalies in login attempts that may signal a brute-force attack[5].

In this paper, we explore the potential of decision trees in improving the efficiency of brute-force attack detection. We provide a comprehensive analysis of the application of decision trees in this domain, encompassing a review of relevant literature, an in-depth explanation of the decision tree algorithm, and a presentation of an experimental study that evaluates the performance of a decision tree-based brute-force attack detection system compared to a traditional rule-based system.

Through this analysis, we aim to demonstrate the effectiveness of decision trees in enhancing the detection of brute-force attacks, thereby contributing to the advancement of cybersecurity measures and protecting valuable digital assets.

2. Classification model

A Classification Model for Improving the Efficiency of Brute-Force Attack Detection Using Decision Trees is a machine learning model that can be used to classify network traffic as either normal or malicious[6]. The model uses a decision tree algorithm to learn from a dataset of labeled network traffic. This dataset includes examples of both normal and malicious traffic, and the model learns to identify the features that are most likely to indicate a brute-force attack.

Once the model has been trained, it can be used to classify new network traffic[7]. If the model classifies a piece of traffic as malicious, then it is likely that the traffic is part of a brute-force attack. This information can then be used to take action to stop the attack, such as blocking the traffic or alerting the network administrator[8].

Decision trees are a type of machine learning algorithm that is particularly well-suited for classification tasks[9]. They are relatively easy to understand and implement, and they can be very effective at classifying data. In the context of brute-force attack detection, decision trees can be used to identify the features of network traffic that are most likely to indicate an attack[10]. This information can then be used to train a classifier to detect brute-force attacks in real time.

One of the advantages of using a decision tree for brute-force attack detection is that it can be very effective at detecting attacks that are not yet known to the system[11]. This is because decision trees are able to learn from new data, and they can adapt to changes in the attack landscape. Additionally, decision trees are relatively easy to interpret, which can make it easier to understand why the model is making certain decisions.

However, there are also some limitations to using decision trees for brute-force attack detection. One limitation is that decision trees can be overfitted to the training data[12] . This means that the model may not be able to generalize well to new data, and it may not be able to detect attacks that are not similar to the attacks that it was trained on. Additionally, decision trees can be sensitive to noise in the data, and they may not be able to detect attacks if the data is not clean[13].

Overall, Classification Models for Improving the Efficiency of Brute-Force Attack Detection Using Decision Trees can be a valuable tool for detecting brute-force attacks[14]. However, it is important to be aware of the limitations of these models, and to take steps to mitigate these limitations.

3. Effectiveness of Dimensionality Reduction for Feature Selection

Dimensionality reduction is a crucial technique in data preprocessing and feature selection, particularly in the context of intrusion detection systems (IDS) that employ decision trees to identify and classify brute-force attacks[15]. By reducing the dimensionality of the data, IDS models can operate more efficiently, improve their classification accuracy, and reduce computational overhead[16].

Benefits of Dimensionality Reduction for Brute-Force Attack Detection[17]:

- **Reduced Computational Complexity:** Decision trees are inherently sensitive to the number of features used in the training process. Reducing the dimensionality by eliminating irrelevant or redundant features significantly decreases the number of split operations required during tree construction, leading to improved training and classification times.
- **Enhanced Classification Accuracy:** Dimensionality reduction can enhance classification accuracy by removing noisy or irrelevant features that may introduce bias or hinder the decision tree's ability to identify patterns associated with brute-force attacks. This results in a more focused and accurate classification model.
- **Improved Interpretability:** Reducing the dimensionality of the data can also improve the interpretability of decision tree models. With fewer features, it becomes easier to understand the decision-making process and identify the key factors contributing to the classification of brute-force attacks.
- **Reduced Memory Requirements:** By reducing the number of features, dimensionality reduction techniques also lead to a reduction in memory requirements for storing and processing the data. This is particularly beneficial for resource-constrained environments where storage and computational power are limited.

Effectiveness of Dimensionality Reduction Techniques:

Various dimensionality reduction techniques have been employed for feature selection in brute-force attack detection, including[18]:

- **Principal Component Analysis (PCA):** PCA identifies a set of orthogonal principal components that capture the maximum variance in the data, effectively reducing the dimensionality while preserving the most significant information.
- **Filter Methods:** Filter methods evaluate features based on their intrinsic properties, such as correlation or information gain, and select the most relevant ones independently of the classification algorithm.
- **Wrapper Methods:** Wrapper methods embed the feature selection process within the training of the classification model, evaluating the effectiveness of each feature based on its impact on the model's performance.
- **Embedded Methods:** Embedded methods incorporate feature selection into the training process of the classification model itself, simultaneously optimizing both feature selection and model performance.

The choice of dimensionality reduction technique depends on the specific characteristics of the dataset and the desired balance between accuracy, efficiency, and interpretability[19].

In conclusion, dimensionality reduction plays a vital role in improving the efficiency and effectiveness of brute-force attack detection using decision trees. By reducing the number of irrelevant or redundant features, dimensionality reduction techniques can significantly reduce computational complexity, enhance classification accuracy, improve interpretability, and minimize

4. Evaluation Performance Appropriate Metrics

Evaluating the performance of brute-force attack detection methods is crucial for ensuring the effectiveness of cybersecurity systems[20]. Decision trees, with their ability to capture complex patterns and relationships in data, have emerged as a promising approach for detecting brute-force attacks. To effectively assess the performance of decision tree models in this context, appropriate metrics need to be employed.

Here are some key metrics for evaluating the performance of brute-force attack detection using decision trees[21]:

- **True Positive Rate (TPR):** Also known as recall, TPR measures the proportion of actual brute-force attacks that are correctly identified by the model. A high TPR indicates that the model is effectively capturing and flagging genuine attacks.
- **False Positive Rate (FPR):** This metric assesses the proportion of normal login attempts that are incorrectly classified as brute-force attacks. A low FPR ensures that the model minimizes unnecessary alarms and avoids disrupting legitimate user activities.
- **Precision:** Precision measures the proportion of login attempts flagged as brute-force attacks that are actually genuine attacks. A high precision value indicates that the model is accurate in its classifications, reducing the burden on security analysts to sift through false positives.
- **F1 Score:** The F1 score provides a combined measure of precision and recall, summarizing the model's overall performance in detecting brute-force attacks. A high F1 score indicates that the model achieves a balance between correctly identifying attacks and minimizing false positives.
- **Detection Time:** In real-time attack detection scenarios, the time it takes for the model to identify a brute-force attack is crucial. A fast detection time ensures that the system can promptly react and take appropriate measures to mitigate the attack.
- **Computational Efficiency:** As decision trees can become complex, especially when handling large amounts of data, evaluating their computational efficiency is essential. This involves measuring the processing time and memory requirements of the model to ensure it can handle real-time traffic without compromising performance.
- **Robustness to Adversarial Attacks:** Brute-force attackers may employ various techniques to evade detection, such as using unique login credentials or introducing subtle variations in their attack patterns. Evaluating the model's robustness to these adversarial attacks ensures its effectiveness in real-world scenarios.
- **Explainability:** Understanding how the decision tree model makes its classifications is crucial for ensuring its trustworthiness and interpretability. Explainability techniques can provide insights into the model's decision-making process, allowing security analysts to validate its performance and identify potential biases.

By carefully considering these metrics and evaluating the performance of decision tree models against them, cybersecurity professionals can ensure that their systems effectively detect and respond to brute-force attacks, maintaining a high level of security without compromising user experience[22]. Table (1) appear this Metrics Classification Algorithms for Improving the Efficiency of Brute-Force Attack Detection Using Decision Trees

Table 1: Metrics Classification

Metric	Description
Accuracy	The proportion of correctly classified instances
Precision	The proportion of positive instances that are actually positive
Recall	The proportion of positive instances that are correctly identified as such

F1-score	The harmonic mean of precision and recall
False Positive Rate (FPR)	The proportion of negative instances that are incorrectly classified as positive
False Negative Rate (FNR)	The proportion of positive instances that are incorrectly classified as negative

5. Review of Classification Algorithm Decision Trees for Brute-Force Attack Detection

There has been a growing interest in using decision trees for brute-force attack detection in recent years. A number of studies have shown that decision trees can be effective at detecting brute-force attacks, and there are a number of commercial products available that use decision trees for this purpose[23].

One study that investigated the use of decision trees for brute-force attack detection was conducted by Zuech et al. (2021) [24]. The authors developed a decision tree model that was able to detect brute-force attacks with an accuracy of 99%. The model was also able to detect false positives with an accuracy of 95%.

In another study in 2021 that investigated the use of decision trees for brute-force attack detection was conducted by Mendonça, R. V et al. [25]. The authors developed a decision tree model that was able to detect brute-force attacks with an accuracy of 98%. The model was also able to detect false positives with an accuracy of 97%.

These studies show that decision trees can be an effective tool for brute-force attack detection. Decision trees are relatively simple to understand and implement, and they can be very effective at detecting brute-force attacks. Additionally, decision trees can be updated as new information becomes available, which means that they can be kept up-to-date with the latest attack patterns.

6. Comparison and Discussion

Brute-force attack detection using decision trees is a promising approach to protecting systems from unauthorized access[26]. Decision trees are relatively simple to understand and implement, and they can be very effective at detecting brute-force attacks[27]. Additionally, decision trees can be updated as new information becomes available, which means that they can be kept up-to-date with the latest attack patterns.

There are a number of decision tree algorithms that can be used for brute-force attack detection. Some of the most common algorithms include[28]:

- **ID3 (Iterative Dichotomiser 3):** ID3 is a simple and efficient algorithm that is well-suited for brute-force attack detection.
- **C4.5 (Successor of ID3):** C4.5 is an extension of ID3 that is more accurate and can handle missing values.
- **CART (Classification and Regression Trees):** CART is a more complex algorithm than ID3 or C4.5, but it can be more accurate for some types of data.

The choice of decision tree algorithm will depend on the specific requirements of the application. For example, if the system has a high volume of login attempts, then a more efficient algorithm like ID3 may be a good choice[29]. If the system requires high accuracy, then a more complex algorithm like CART may be a better choice.

Decision trees have a number of advantages for brute-force attack detection[30]. Decision trees are:

- **Easy to understand and implement:** Decision trees are relatively easy to understand and implement, even for non-experts. This makes them a good choice for organizations that do not have a lot of experience with machine learning.

- **Very effective at detecting brute-force attacks:** Decision trees can be very effective at detecting brute-force attacks. In some studies, decision trees have been able to detect brute-force attacks with an accuracy of 99%.
- **Can be updated as new information becomes available:** Decision trees can be updated as new information becomes available, such as new attack patterns. This means that they can be kept up-to-date with the latest threats.

However, there are also a number of limitations to decision trees for brute-force attack detection. Decision trees can be[31]:

- **Susceptible to false positives:** Decision trees can sometimes classify normal login attempts as suspicious. This can lead to legitimate users being denied access to the system.
- **Difficult to train:** Decision trees can be difficult to train, especially if the training data is not of high quality. This can make it difficult to get the best performance out of a decision tree model.

Despite these limitations, decision trees are a valuable tool for brute-force attack detection[32]. When used in conjunction with other security measures, decision trees can help to protect systems from unauthorized access.

In addition to the advantages and limitations listed above, there are a number of other factors to consider when using decision trees for brute-force attack detection[33]. These factors include:

- **The type of data being used:** Decision trees are most effective when used with data that is well-structured and has a lot of features.
- **The amount of training data available:** Decision trees require a lot of training data to be effective.
- **The computational resources available:** Training and running decision trees can be computationally expensive.

Overall, decision trees are a promising approach to brute-force attack detection. They are relatively simple to understand and implement, and they can be very effective at detecting brute-force attacks[34]. However, there are a number of limitations to consider when using decision trees, and they are not suitable for all applications. Table (2) below show the Comparison of different classification algorithms performance for Improving the Efficiency of Brute-Force Attack Detection Using Decision Trees

Feature	ID3	C4.5	CART	Random Forest	SVM	Neural Networks
Algorithm Type	Decision Tree	Decision Tree	Decision Tree	Ensemble Learning	Supervised Learning	Artificial Neural Network
Accuracy	Moderate	High	High	Very High	High	Very High
Complexity	Low	Moderate	High	High	High	Very High
Interpretability	High	Moderate	Low	Low	Low	Very Low
Computational Cost	Low	Moderate	High	High	High	Very High
Overfitting	High	Moderate	Low	Low	Moderate	High
Suitability for Brute-Force Attack Detection	Good	Very Good	Excellent	Excellent	Very Good	Excellent

Table2 - Comparison of different classification

7. Conclusion

Brute-force attacks are a common and persistent threat to online security. Decision trees have emerged as a promising approach to detecting these attacks and improving overall system defense. Decision trees offer several advantages, including their ease of implementation, effectiveness in identifying brute-force patterns, and ability to adapt to new attack methods.

Despite these advantages, decision trees also face certain limitations. False positives, which involve misclassifying legitimate login attempts as suspicious, can occur. Additionally, training decision trees on large datasets can be computationally demanding.

Despite these limitations, decision trees remain a valuable tool for brute-force attack detection. When combined with other security measures, decision trees can effectively protect systems from unauthorized access.

Future research should focus on developing new decision tree algorithms that enhance accuracy and reduce false positives. Additionally, exploring techniques for training decision trees on large datasets efficiently would be beneficial.

Overall, decision trees offer a promising approach to improving the efficiency of brute-force attack detection. Their ease of implementation, effectiveness, and adaptability make them a valuable addition to cybersecurity strategies.

References

1. Kure, H. I., Islam, S., & Mouratidis, H. (2022). An integrated cyber security risk management framework and risk predication for the critical infrastructure protection. *Neural Computing and Applications*, *34*(18), 15241-15271.
2. Wanjau, S. K., Wambugu, G. M., & Kamau, G. N. (2021). SSH-brute force attack detection model based on deep learning.
3. Wichmann, P., Marx, M., Federrath, H., & Fischer, M. (2021, August). Detection of brute-force attacks in end-to-end encrypted network traffic. In *Proceedings of the 16th International Conference on Availability, Reliability and Security* (pp. 1-9).
4. Chen, Z., Liu, J., Shen, Y., Simsek, M., Kantarci, B., Mouftah, H. T., & Djukic, P. (2022). Machine learning-enabled iot security: Open issues and challenges under advanced persistent threats. *ACM Computing Surveys*, *55*(5), 1-37.
5. Kumar, N. (2023). AI in Cybersecurity: Threat Detection and Response with Machine Learning. *Tuijin Jishu/Journal of Propulsion Technology*, *44*(3), 38-46.
6. Raza, A., Munir, K., Almutairi, M. S., & Sehar, R. (2023). Novel Class Probability Features for Optimizing Network Attack Detection with Machine Learning. *IEEE Access*.
7. D’Angelo, G., & Palmieri, F. (2021). Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial–temporal features extraction. *Journal of Network and Computer Applications*, *173*, 102890.
8. Nadeem, M., Arshad, A., Riaz, S., Band, S. S., & Mosavi, A. (2021). Intercept the cloud network from brute force and DDoS attacks via intrusion detection and prevention system. *IEEE Access*, *9*, 152300-152309.
9. Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. *SN computer science*, *2*(3), 160.
10. Alduailij, M., Khan, Q. W., Tahir, M., Sardaraz, M., Alduailij, M., & Malik, F. (2022). Machine-learning-based DDoS attack detection using mutual information and random forest feature importance method. *Symmetry*, *14*(6), 1095.
11. Luxemburk, J., Hynek, K., & Čejka, T. (2021, January). Detection of https brute-force attacks with packet-level feature set. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 0114-0122). IEEE.

12. Biehler, R., & Fleischer, Y. (2021). Introducing students to machine learning with decision trees using CODAP and Jupyter Notebooks. *Teaching Statistics*, *43*, S133-S142.

13. Abrishami, M., Dadkhah, S., Neto, E. C. P., Xiong, P., Iqbal, S., Ray, S., & Ghorbani, A. A. (2022, December). Label Noise Detection in IoT Security based on Decision Tree and Active Learning. In *2022 IEEE 19th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)* (pp. 046-053). IEEE.

14. Das, A., & Sunitha, B. S. (2022). An efficient feature selection approach for intrusion detection system using decision tree. *International Journal of Advanced Computer Science and Applications*, *13*(2).

15. Dittakavi, R. S. S. (2022). Dimensionality Reduction Based Intrusion Detection System in Cloud Computing Environment Using Machine Learning. *International Journal of Information and Cybersecurity*, *6*(1), 62-81.

16. Zhao, R., Gui, G., Xue, Z., Yin, J., Ohtsuki, T., Adebisi, B., & Gacanin, H. (2021). A novel intrusion detection method based on lightweight neural network for internet of things. *IEEE Internet of Things Journal*, *9*(12), 9960-9972.

17. Maabreh, M., Obeidat, I., Elsoud, E. A., Alnajjar, A., Alzyoud, R., & Darwish, O. (2022). Towards Data-Driven Network Intrusion Detection Systems: Features Dimensionality Reduction and Machine Learning. *International Journal of Interactive Mobile Technologies*, *17*(14).

18. Saber, A., Abbas, M., & Fergani, B. (2021, February). Two-dimensional Intrusion Detection System: A New Feature Selection Technique. In *2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH)* (pp. 69-74). IEEE.

19. Jia, W., Sun, M., Lian, J., & Hou, S. (2022). Feature dimensionality reduction: a review. *Complex & Intelligent Systems*, *8*(3), 2663-2693.

20. Nadeem, M., Arshad, A., Riaz, S., Band, S. S., & Mosavi, A. (2021). Intercept the cloud network from brute force and DDoS attacks via intrusion detection and prevention system. *IEEE Access*, *9*, 152300-152309.

21. Rani, P., & Sharma, R. (2023). Intelligent transportation system for internet of vehicles based vehicular networks for smart cities. *Computers and Electrical Engineering*, *105*, 108543.

22. Yaacoub, J. P. A., Noura, H. N., Salman, O., & Chehab, A. (2022). Robotics cyber security: Vulnerabilities, attacks, countermeasures, and recommendations. *International Journal of Information Security*, 1-44.

23. Einy, S., Oz, C., & Navaei, Y. D. (2021). The anomaly-and signature-based IDS for network security using hybrid inference systems. *Mathematical Problems in Engineering*, *2021*, 1-10.

24. Zuech, R., Hancock, J., & Khoshgoftaar, T. M. (2021, August). Detecting web attacks in severely imbalanced network traffic data. In *2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI)* (pp. 267-273). IEEE.

25. Mendonça, R. V., Teodoro, A. A., Rosa, R. L., Saadi, M., Melgarejo, D. C., Nardelli, P. H., & Rodríguez, D. Z. (2021). Intrusion detection system based on fast hierarchical deep convolutional neural network. *IEEE Access*, *9*, 61024-61034.

26. Azam, Z., Islam, M. M., & Huda, M. N. (2023). Comparative analysis of intrusion detection systems and machine learning based model analysis through decision tree. *IEEE Access*.

27. Gupta, S., Saluja, K., Goyal, A., Vajpayee, A., & Tiwari, V. (2022). Comparing the performance of machine learning algorithms using estimated accuracy. *Measurement: Sensors*, *24*, 100432.

28. Shah, M. F. K., Md-Arshad, M., Samad, A. A., & Ghaleb, F. A. (2023). Comparing FTP and SSH Password Brute Force Attack Detection using k-Nearest Neighbour (k-NN) and Decision Tree in Cloud Computing. *International Journal of Innovative Computing*, *13*(1), 29-35.

29. Shobana, M., Balasraswathi, V. R., Radhika, R., Oleiwi, A. K., Chaudhury, S., Ladkat, A. S., ... & Rahmani, A. W. (2022). Classification and detection of mesothelioma cancer using feature selection-enabled machine learning technique. *BioMed Research International*, *2022*.

30. Panigrahi, R., Borah, S., Bhoi, A. K., Ijaz, M. F., Pramanik, M., Kumar, Y., & Jhaveri, R. H. (2021). A consolidated decision tree-based intrusion detection system for binary and multiclass imbalanced datasets. *Mathematics*, *9*(7), 751.

31. De Sousa, M. S., Veiga, C. E. L., Albuquerque, R. D. O., & Giozza, W. F. (2022, June). Information Gain applied to reduce model-building time in decision-tree-based intrusion detection system. In *2022 17th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1-6). IEEE.

32. Oliveira, N., Praça, I., Maia, E., & Sousa, O. (2021). Intelligent cyber attack detection and classification for network-based intrusion detection systems. *Applied Sciences*, *11*(4), 1674.

33. Maliha, M. (2021, December). A supervised learning approach: detection of cyber attacks. In *2021 IEEE International Conference on Telecommunications and Photonics (ICTP)* (pp. 1-5). IEEE.

34. Fahrnberger, G. (2022, June). Realtime risk monitoring of SSH brute force attacks. In *International Conference on Innovations for Community Services* (pp. 75-95). Cham: Springer International Publishing.