DS Journal of Cyber Security
Volume 1 Issue 1, 19-27, Jul - Sep 2023
ISSN: 2584-0665 / https://doi.org/10.59232/CYS-V1I1P102

Original Article

DNS-DoS Detection System using Hybrid Domain Features-Based Support Vector Machine

S. Veerapandi

Department of Computer Science, Mannar Thirumalai Naicker College, Tamilnadu, India.

viruyamini@gmail.com

Received: 16 May 2023; Revised: 26 May 2023; Accepted: 8 June 2023; Published: 6 July 2023;

Abstract - The Domain Name System (DNS) changes IP addresses into memorable domain names and the other way around in the internet ecosystem. In order to attack DNS, the malicious user makes use of DNS issues. DNS amplification attacks based on Distributed Denial of Service (DDoS) and an attack vector focusing on DNS tunneling are one of the most refined types of DNS attacks. It is a system that detects intrusions that analyses traffic for network intrusion, although it does not just monitor for DNS intrusion. In this research, a Support Vector Machine is utilised in conjunction with DNS-DoS detection to identify critical DNS-related attacks. Hybrid domain features-based Support Vector Machines with DNS-DoS detection systems are proposed to detect DNS attacks. The features of DNS-DoS detection systems are classified into two types, namely, payload tunneling features and domain host features. Using the Support Vector Machine (SVM) and a DNS attacker, determine if a DNS-DoS attack occurred or not. Sensitivity, accuracy, and specificity are the factors taken into account when evaluating the efficiency of the suggested model. In comparison to Decision Tree (DT), Support Vector Machine (SVM), Naive Bayes (NB) and Random Forest (RF), the technique enhances efficiency by 3.9%, 1.6%, and 0.41%, respectively.

Keywords - Domain Name System, Detection, DoS, Hybrid, Support Vector Machine, Decision Tree, Naïve Bayes, Random Forest.

1. Introduction

The database of the website is a Domain Name System (DNS). To obtain information on the web, people use websites like espn.com or the New York Times. Web pages interact using addresses assigned to Internet Protocol (IP). To access resources on the Internet, DNS converts domain names into IP addresses. Each internet-connected gadget has a different IP address. An operating system that operates off a disc drive is known as Disc Operating System (DOS). Furthermore, the word can be used to describe a particular group of disc operating systems, namely the Microsoft DOS or series MS-DOS. An Operating System is a software that controls a computer's hardware and devices and permits the use of additional software programmes (OS).

Networks, software, and hardware must be protected from multiple threats by implementing cybersecurity. Cyber-attacks, such as systems for detecting, preventing, and mitigating threats, and firewall systems, are executed across numerous network infrastructure components to prevent multiple layers of security. Among the security solutions listed, Intrusion Detection Systems (IDS) are the most cost-effective and widely used by web users. As the Internet expands, In addition to DNSEXT, the Internet Standards Organization also has a DNS Extension Working Group (DNSEXT) that has classified DNS as a “Critical Infrastructure”. There are a number of

distributed systems that use DNS services to resolve addresses, such as internet services, email services, file transfer protocols, security tokens, etc.

A wide range of security issues, such as cache poisoning, DNS hijacking, NXDOMAIN/NSNX, phantom domain assaults, DNS spoofing, botnet-based DDoS, DNS floods, amplification attacks, and tunneling, can make DNS vulnerable. The internet may become unreachable to a significant number of users due to DNS attacks, or users may be sent to fraudulent or duplicate websites where they can be tricked into providing personal and sensitive financial information. DNS server participation in data exfiltration and Command and Control (C&C) communication can be affected through DNS attacks. For a resilient DNS infrastructure, the following DNS assaults should be quickly addressed: Defending against DoS/DDoS through DNS reflection & amplification tunneling and BOTNET.

An additional tool for assuring a safe DNS system is a DNS firewall. The DNS firewall is an effective security technique for filtering unreliable DNS traffic and defending against numerous DNS threats. Certain DNS requests are limited to known dangerous websites. Malware protection and Response Policy Zones (RPZ) capabilities are features that the majority of DNS firewalls offer. Based on attack signatures, it can identify the majority of DNS attacks. Provide defence by throwing packets based on matching signatures. It can keep track of real-time DNS activity and block certain domains.

The cryptographic hash function and symmetric keys are used in Transactional Signatures (TSIG), which guarantee secure communication between secondary and primary domain controllers. Additionally, it ensures the accuracy and integrity of the information received during zone transfers as well as the validity of the DNS answer. An update to a Resource Record (RR) between a master and slave server occurs when IP spoofing is dealt with using the TSIG technique. It is a method for verifying changes made to a DNS database. The only attacks prevented by TSIG's authenticated zone transfers between DNS servers are “spoofing master” attacks.

With the aim of protecting DNS infrastructure, numerous security solutions and protocols have been developed as a result of significant research. However, none succeeded in eradicating all DNS-related dangers, and DNS still has numerous security flaws. Our study closes this space by incorporating a well-known open-source IDS SNORT method to fight against all conceivable DNS-related threats. Researchers also developed DNS tunneling, DoS, and DNS amplification attack signatures. This work describes an intrusion detection system based on Snort for discovering DNS protocol anomalies.

This is an overview of this paper's main contributions;

- In this paper, a novel hybrid domain features-based Support Vector Machine with DNS-DoS detection systems developed to detect DNS attacks has been proposed.
- An Intrusion Detection System (IDS) is a device that analyses traffic for network intrusion, although it does not just monitor for DNS intrusion.
- The features of DNS-DoS detection systems are classified into two types, namely, payload tunneling features and domain host features.
- The Support Vector Machine (SVM) and a DNS attacker determine if a DNS-DoS attack occurred or not.
- As a result, sensitivity, accuracy, and specificity are the factors taken into account when evaluating the efficiency of the suggested model.

2. Literature Survey

In 2000 S. Cheung, K.N. Levitt., et al. [1] proposed an explicit intrusion detection technique that employs a formal specification, modelling, and proof and formal analysis to improve DNS IDS level of certainty. Monitoring malicious DNS traffic was done using DNS wrappers. Anomalies were identified as DNS by analysing DNS traffic to its specifications. The authors also pointed out that this event was marked as potentially malicious a deviation from the authoritative response is observed in the monitored traffic.

In 2009 S. Rastegari, M.I. Saripan., et al. [2] developed an IDS based on machine learning that was proposed for DNS DoS attacks that aimed to exploit neural networks' learning capacity to recognise DNS-DoS attacks. Three alternative self-organising maps neural networks-backpropagation and radial basis function, were used to assess the proposed IDS. On the NS-2 simulator, the complete experiment was simulated. It was found that BP neural networks far outperformed other types of networks, detecting attacks 99 percent of the time and generating fewer false alarms than other types of networks.

In 2016 Hock, Kortis., et al. [3] developed the idea for a DNS security firewall solution that manages transport and network-level DNS security procedures. The cornerstone of the suggested security strategy consists of prioritisation, flow filtering, and traffic shaping. They talked about how a firewall may modify traffic on the fly to ensure packet delivery even if the DNS server is being attacked.

In 2019 S. Spacek, M. Lastovicka., et al. [4] developed a DNS firewall with zones for DNS response policies that are open-source and free. A recommended system was designed with zone limitation, customer alerting, activity logging, and domain blacklist sharing. According to research, the DNS firewall is unable to recognise or fight against the bulk of DNS threats, even though it can be used to restrict access to and prevent identification of C&C and data exfiltration domains that can be used in DNS tunneling attacks.

In 2019 T. Ghosh, E. El-Sheikh., et al. [5] researched many ways to recognise Command and Control (C&C) systems that have been proposed, and one of them uses pattern comparison to recognise a bot's handshake with the botnet's Command and Control system is via SSH. This method is used to identify botnet-based DNS tunneling. The authors provided a multi-step process for keeping track of botnet communications using Fast Flux, DNS tunneling, and the Domain Generation Algorithm. A Bro network security monitoring program was implemented to detect this behaviour, and rules were developed based on anomalies and signatures observed on the network.

In 2021 Y.F. Mohammed., [6] created a reliable and effective network-based prevention against DNS-based attacks, including DNS tunneling risks, using a variety of methodologies, including visualisation, machine learning, and statistical analysis. The suggested approach functions as a DNS server, identifying and blocking threats from various threat tools that involve DNS tunneling.

In 2022 R. Mitsuhashi et al. [7], A machine learning approach based on DoH traffic analysis has been suggested to discover malicious DNS tunnel tools. The suggested solution, based on hierarchical machine learning categorisation, exclusively uses DoH traffic analysis as a data source. According to the evaluation results, 98.02% of the six malicious DNS tunnel tools can be appropriately classified using their suggested method.

It can be seen from the reviews above that these methods have some shortcomings. This research proposes a DNS-DoS detection system using a hybrid domain features-based Support Vector Machine to detect DNS attacks.

3. Proposed Method

This paper uses a proposed DNS-DoS detection while using a Support Vector Machine to detect major DNS-related attacks. A novel hybrid domain features-based Support Vector Machine employing a DNS-DoS detection system detects DNS attacks. The features of DNS-DoS detection systems are classified into two types, namely, payload tunneling features and domain host features. Using the Support Vector Machine (SVM) and a DNS attacker, determine if a DNS-DoS attack occurred or not.

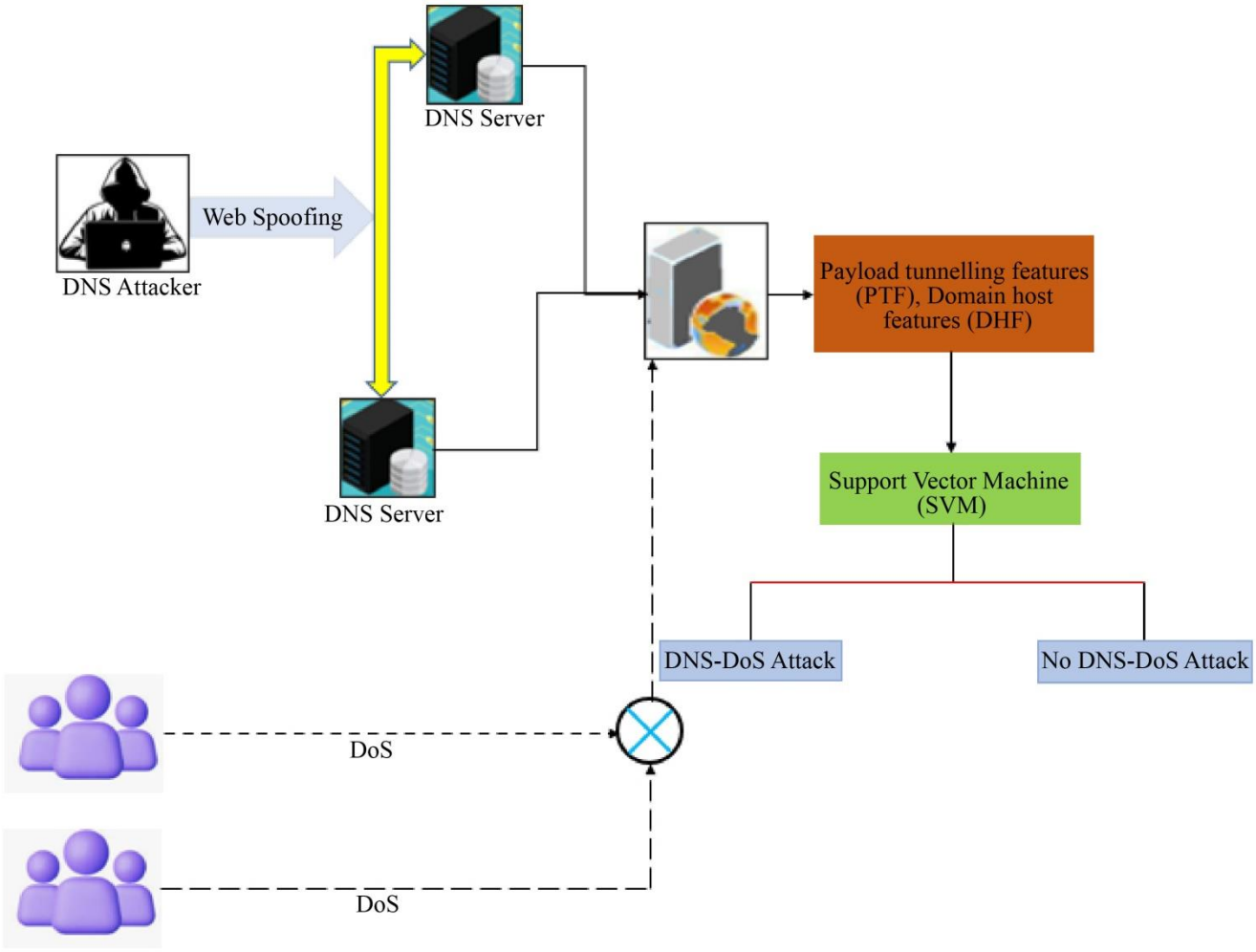


Fig. 1 Proposed DNS -DOS attack

A DNS amplification attack involves flooding a victim's system with a lot of DNS response traffic using publicly available, globally accessible recursive resolvers. This eventually results in a DDoS attack on the target. Here, the attacker spoofs the victim's IP address via DNS queries sent to open recursive resolvers. The DNS replies are subsequently sent back to the target system by the recursive resolvers, as illustrated in Figure 1. The DNS query's "any" option is used by the attacker to expand the attack surface and deliver as much zone data as possible. The attack will then be enhanced since the open DNS servers will respond with all the information they have about a DNS zone. The EDNS extension allows attackers to send large DNS packets to enhance the DNS attack. The DNS packet size would increase if the attacker also utilised DNSSEC protocols. There is a possibility that the 37 bytes of a DNS query packet can provide a response of even more than 3256 bytes. The proposed DNS -DOS reflection attack is shown in Figure 1.

3.1. Payload Tunneling Features (PTF)

The payload tunneling features are size of request and response, entropy of hostnames, statistical analysis, uncommon record types, policy violation, and specific signatures.

3.1.1. Size of Request and Response

Comparing both request and response sizes is one approach. The size of the request and response can be determined by examining them. DNS tunneling utilities generally strive to be as detailed as possible in their recommendations and responses. Due to this, tunneling requests may have lengthy labels and overall names (up to 255 characters).

3.1.2. Entropy of Hostnames

The entropy of a hostname request can be used to identify DNS tunnels. Dictionary words or other words that appear to have significance frequently occur in legitimate DNS names. The use of the character set is distributed more evenly and has higher entropy in encoded names. However, there are instances where DNS names represent specific types of information. By scanning for DNS domains with high entropy, tunneling can be found.

3.1.3. Statistical Analysis

DNS name configuration analysis involves analysing the exact characters in names, an alternate technique for detecting tunneling. Legitimate DNS names often have fewer numbers than encoded names. It has been suggested that domain names are frequently dominated by numerical characters.

3.1.4. Uncommon Record Types

Another option is to find records not normally used by a typical client, like TXT information.

3.1.5. Policy Violation

DNS lookups performed through an internal DNS server may be violated if the policy requires all queries to be made through it. Identifying direct DNS requests on the internet can be done using traffic. In most DNS tunneling methods, requests are routed through an internal DNS server to function.

3.1.6. Specific Signatures

The DNS tunneling techniques used by researchers have often been signed. Specific attributes can be checked in a DNS header using a signature, and specific contents in a payload using a signature. For detecting NSTX DNS tunneling, for example, a Snort signature was created.

3.2. Domain Host Features (DHF)

Analysing traffic involves analysing various requests and responses over time. The quantity and regularity of requests can be widely used to detect tunneling. Among the traffic analysis detecting methods are:

- Volume of DNS traffic per IP address
- Volume of DNS traffic per domain
- Number of hostnames per domain
- Geographic location of the DNS server
- Domain history
- Volume of NX Domain responses

3.3. Support Vector Machine

One of the most popular kernel-based learning algorithms, Support Vector Machines, was first developed by Vapnik and his team in the late 1970s and is currently utilised in many machine learning applications, particularly image classification. Solving a convex quadratic optimisation issue and generating a theoretically globally optimal solution is the main goal of SVMs. This overcomes the issue with the local extremum that is present in other machine-learning techniques.

An SVM is a linear binary classifier that identifies just one boundary between two classes in its most basic configuration. The linear SVM assumes that the input space contains linearly separable multidimensional data. Using the training data, SVMs, in particular, choose the most appropriate hyperplane to separate the dataset into a specific number of predefined classes. These samples are the hardest to categorise and directly affect where the decision boundary should be set. It is possible to define mathematically and geometrically the optimal hyperplane or the maximum margin. When the margin of separation is maximised, the greatest hyperplane is determined by selecting several hyperplanes with no samples in between them. The learning process is the iterative process of creating a classifier with an ideal decision boundary.

SVM's effectiveness greatly hinges on choosing a kernel function that produces dots in the higher dimensional feature space. Theoretically, this space may have an endless number of dimensions, making linear discrimination possible. There are multiple kernel models, including sigmoid, radial basis function, polynomial, and linear, to design various SVMs that satisfy Mercer's condition. Polynomial and Radial Basis Functions (RBFs) are frequently employed to analyse remotely sensed images. Kernels are often established by predefining their models, accompanied by tuning approaches, which may be very expensive computationally, to change the kernel's parameters.

The main factor to consider while choosing a kernel function is the performance of the trained handle on a sample from either the training sample or the validation collection. The fundamental drawback of kernel-based approaches may be the propensity for overfitting in kernel-based models. Finally, fresh approaches were put forth to address these problems, including multiple kernel learning and automatic kernel selection. It is significant that the task of choosing the best kernel belongs to the category of an optimisation problem. Numerous SVM parameters must be optimised, which uses a lot of resources. This requires the use of the Genetic optimisation Algorithm (GA) and Particle Swarm Optimisation (PSO) algorithms as an alternative method of determining SVM parameters. GA-SVM and SVM-PSO are evolutionary methods that maximise C and gamma using principles from biological systems. PSO has some attractive qualities and has often proven more effective than GA and other related evolutionary strategies.

Employing additional kernels in the site of linear bounds increases the decision boundary flexibility of SVMs, enhancing classification performance. Despite these benefits, there are still some issues, such as selecting the optimum kernel parameters, picking an appropriate kernel, and the SVM's somewhat sophisticated mathematics, which, from the viewpoint of a non-expert user, restricts the usefulness of cross-disciplinary applications of SVMs.

4. Result & Discussion

Support Vector Machines - a type of machine learning method was used in this research project. The results are assessed using the confusion matrix, a technique for comparing algorithm performance. Usually, it consists of a table describing the various attributes. Classification results are usually presented in a table. When the output is in the form of a “1,” which denotes the detection of a threat, or a “0,” which denotes regular network activity. The

initial values of the class feature that are already available in the evaluation database are evaluated with these values. The confusion matrix can be used to display four aspects:

- True Positive (TP) : As a result of the classifier detecting the attack in the right class feature, the attack has been correctly identified.
- True Negative (TN) : A negative value can be found for the class feature, i.e., normal traffic.
- False Positive (FP) : Normal traffic is wrongly classified by the classifier as an attacker.
- False Negative (FN) : An attacker's record is misidentified by the classifier as ordinary traffic.

These requirements enable the creation of five measures for classifier evaluation: accuracy, sensitivity, specificity, AUC, and Matthew's Correlation Coefficient (MCC). Here is the calculation for these two metrics: A record's likelihood of being correctly classified as either an attack or regular traffic might be used as an indicator of accuracy. The results of the calculation of total accuracy are as follows:

$$\text{Accuracy} = \frac{TN + TP}{TP + TN + FN + FN}$$

From each problematic computer data set, sensitivity analysis studies the impact of uncertainty on a specific model result.

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

Specificity, which indicates the possibility of testing threats without producing false-positive results.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

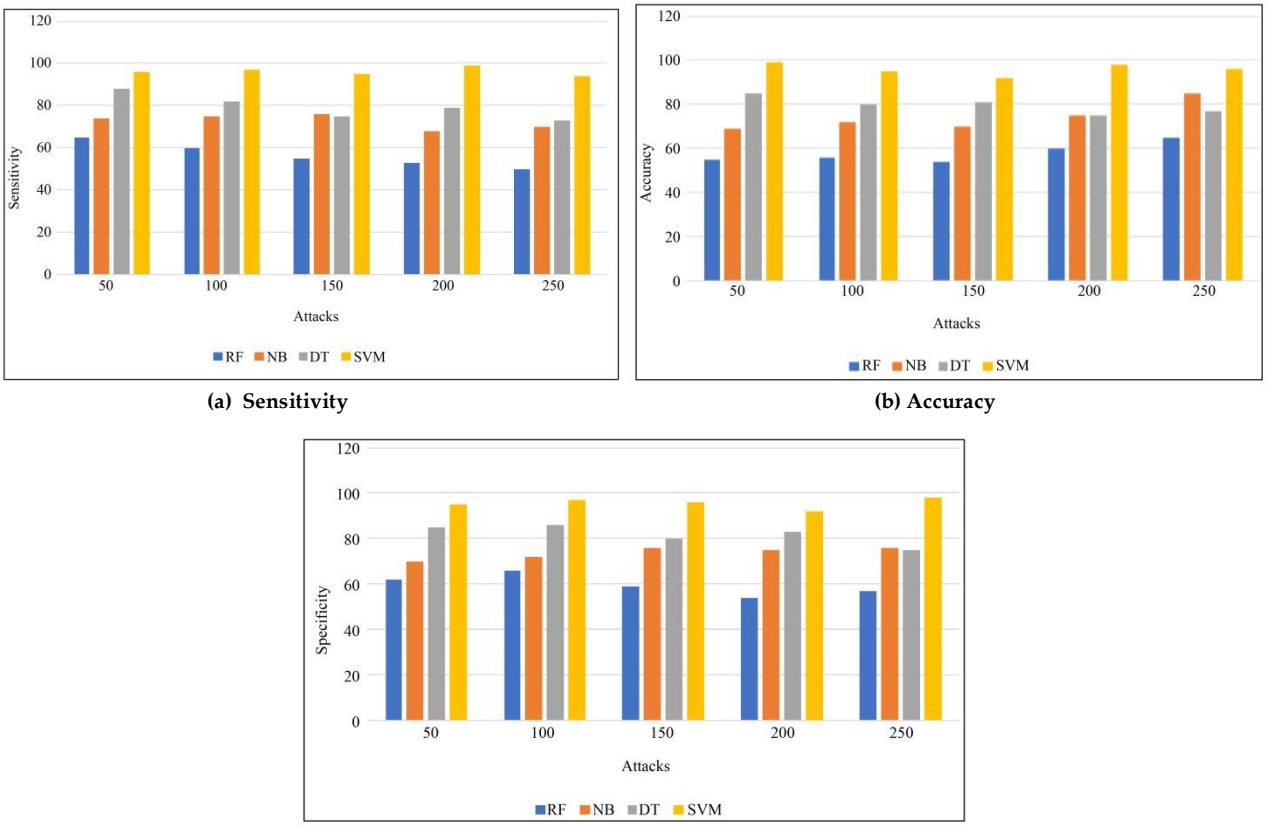
Table 1. AUC classification of accuracy

Sl. NO.	AUC Range	Classification
1	0.95<AUC<1.05	Very good
2	0.85<AUC<0.95	Good
3	0.75<AUC<0.85	Poor
4	0.65<AUC<0.75	Very poor

The accuracy is measured by the AUC, or area under the ROC curve. For calculating the AUC of ROC curves, use the following equation:

$$AUC = \int_0^1 ROC(t)dt$$

Where t=1-Specificity, and ROC(t) is sensitivity.



(a) Sensitivity **(b) Accuracy**

(c) Specificity

Fig. 2 Comparative survey of the proposed with existing (a) Sensitivity, (b) Accuracy, and (c) Specificity.

In order to show that the suggested Support Vector Machine methodology is more effective than existing methods, it was compared to those methods. Sensitivity, accuracy, and specificity all affect performance. The suggested strategy received a sensitivity score of 96%, an accuracy score of 99%, and a specificity score of 95% for the suggested method. The suggested approach performs better in terms of accuracy, sensitivity, and specificity when compared to current classifiers. According to the suggested technique, the test is 99% accurate and predicts attacks properly. Compared to the existing models, the suggested method produces more accurate results.

5. Conclusion

In this research paper, a proposed DNS-DoS detection is used while using a Support Vector Machine to detect major DNS-related attacks. A novel hybrid domain features-based Support Vector Machine employing a DNS-DoS detection system detects DNS attacks. Three parameters are considered to validate the developed model's effectiveness: sensitivity, accuracy, and specificity. Here, the suggested strategy produces results with greater accuracy than the current models. The proposed approach outperforms Random Forest (RF), Support Vector Machine (SVM), Decision Tree (DT), and Naive Bayes (NB) in terms of overall accuracy improvements of 3.9%, 1.6%, and 0.41%, respectively. Future research will concentrate on automatically creating the necessary signatures and detecting network attacks of targeted cyber threats and DNS tunneling attack tools.

References

[1] S. Cheung, and K.N. Levitt, "A Formal-Specification Based Approach for Protecting the Domain Name System," *Proceeding International Conference on Dependable Systems and Networks, DSN* 2000, pp. 641–651, 2000. [CrossRef] [Google Scholar] [Publisher Link]

[2] Samaneh Rastegari, M. Iqbal Saripan, and Mohd Fadlee A. Rasid, "Detection of Denial-of-Service Attacks against Domain Name System Using Neural Networks," *International Journal of Computer Science Issues*, vol. 6, 2009. [Google Scholar] [Publisher Link]

[3] Filip Hock, and Peter Kortiš, "Design Implementation and Monitoring of the Firewall System for a DNS Server Protection," *2016 International Conference on Emerging Elearning Technologies and Applications*, *ICETA*, pp. 91–96, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[4] Stanislav Špaček et al., "Current Issues of Malicious Domains Blocking," *In: 2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM),* pp. 551–556, 2019. [Google Scholar] [Publisher Link]

[5] Tirthankar Ghosh, Eman El-Sheikh, and Wasseem Jammal, "A Multi-Stage Detection Technique for DNS-Tunneled Botnets," *Proceedings of 34th International Conference on Computers and Their Applications*, pp. 137–143, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[6] Yasir Faraj Mohammed, "*Network-Based Detection and Prevention System against DNS-Based Attacks*," University of Arkansas, 2021. [Google Scholar] [Publisher Link]

[7] Rikima Mitsuhashi et al., "Malicious DNS Tunnel Tool Recognition Using Persistent DoH Traffic Analysis," *IEEE Transactions on Network and Service Management*, 2022. [CrossRef] [Google Scholar] [Publisher Link]