International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 11, No.1 (Mar-2022)

https://dx.doi.org/10.12785/ijcds/110185

Malicious traffic Detection of DNS over HTTPS using Ensemble Machine Learning

Sunil Kumar Singh[1] **and Pradeep Kumar Roy**[2]

[1]*School of Computer Science and Engineering, VIT-AP University, Near Vijayawada, Andhra Pradesh, India*
[2]*Department of Computer Science and Engineering, Indian Institute of Information Technology Surat, Gujarat, India*

Received 30 Apr. 2021, Revised 10 Feb. 2022, Accepted 15 Mar. 2022, Published 31 Mar. 2022

Abstract: As the Internet is growing very fast, the Domain Name System remains under constant attacks and day by day its vulnerability is increasing. In the cyberattacks, it has been shown that the maximum attackers make target on Domain Name System. Several security add-ons came with DNS to secure it, but we have not come across any robust solution until now. DNS over HTTPS and DNS over TLS are introduced recently with encrypted DNS to reduce the visibility of DNS requests. DNS over HTTPS has been designed to mitigate the DNS security issues but it has own drawbacks like it bypasses the local firewalls. However, DNS over HTTPS is a popular protocol now, but it is also vulnerable. This paper presents a Machine Learning approach to detect. DNS over HTTPS traffic and to filter it into Benign-DNS over HTTPS traffic and Malicious-DNS over HTTPS traffic using ensemble machine learning algorithms. To find the best prediction results, we have applied various ML models such as; (i) Decision tree, ii) Logistic regression, (iii) K nearest neighboring, and (iv) Random forest. Several evaluation metrics have been considered to analyze the performance, like precision, recall, F1-score, and confusion matrix. The results analysis is carried out on a benchmark DNS over HTTPS dataset (CIRA-CIC-DoHBrw-2020) with 30 extracted features. To make this model robust, several parameters are used to check its performance. An ensemble learning-based RF classifier emerge as the best-suited model with 100% accuracy. The outcomes of the proposed ensemble learning model confirmed that it is the best choice to secure the DNS over HTTPS based DNS attacks because this model detected most malicious activities.

Keywords: Domain Name System, DNS, DNS-over-HTTPS. DoH, Machine Learning, DNS encryption, DNS Security, Ensemble Learning

1. Introduction and Overview

Domain Name System (DNS) is the directory or telephone book for the Internet. We access web pages and data through domain names, like Google.com or Amazon.com, etc. DNS converts the domain name into the equivalent IP address, which helps a browser to open any web resources [1], [2]. To find the appropriate IP addresses for the queried DNS, a resolver plays a major role. This resolver requests various servers to map the domain names with corresponding IP addresses. There are two ways of DNS service: authoritative DNS and recursive DNS. An authoritative DNS service offers to keep posted mechanism that designer's usage to manage their public DNS names. After that, it replies DNS queries by converting domain names into IP address and it help systems to communicate with each other [3]. In a recursive DNS look-up, one DNS server communicates with other servers to find the corresponding IP address and send it back to the user. It works in an iterative query where the resolver involves searching IP addresses in every DNS server [4]. The complete DNS process is illustrated in Figure 1.

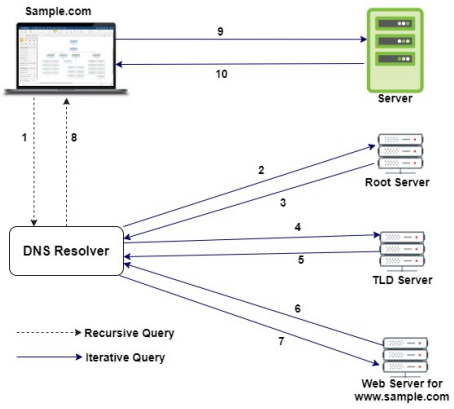

Figure 1. Domain Name Server

The following steps are followed in DNS mechanism:

- A user opens a website named sample.com with a web browser.

E-mail address: sunil.singh@vitap.ac.in, pkroynitp@gmail.com

- This request sample.com goes to a DNS resolver.
- The DNS resolver forwards the request for sample.com to a DNS root name server.
- The DNS resolver next forwards the query for www.sample.com again to one of the TLD name servers for .com domains.
- The DNS resolver next forwards the query for www.sample.com again to one of the TLD name servers for .com domains.
- Then TLD name server redirects to sample.com by giving the details to the DNS resolver.
- The .com name server searches in the sample.com and gets the associated IP address for it.
- This .com name server returns the mapped IP address for www.sample.com to the DNS resolver.
- Now DNS resolver has that IP address, and it provides to a web browser.
- The web browser sends a request for www.sample.com to the IP address that it got from the DNS resolver.
- The server replies the requested web page for www.sample.com to the web browser.

DNS is the backbone of the Internet from the beginning and also more vulnerable due to its openness. On DNS, there are several cyberattacks; we have seen in the recent past. There are different types of DNS attacks people have encountered: Domain hijacking, DNS flood attack, DDOS or DRDOS, DNS cache poisoning, DNS tunneling, DNS hijacking, random subdomain, and NXDOMAIN attack [5]. The different types of DNS attacks can be seen in Figure 2. According to a survey of 900 technology professions across North America, Europe, and the Asia Pacific, the "2020 Global DNS Threat Report" found that 79% of organizations were affected by DNS attacks in 2019. As per the report, the application downtime was the major upshot of a DNS attack. In general, DNS performs its queries and responses in plaintext using UDP. This leads to attackers who can easily read or monitor data transmissions, as shown in Figure 3. A new privacy protection scheme has been for DNS with a public resolver [6]. This privacy scheme is also not sufficient to stop the privacy issues of DNS. To prevent DNS services from unauthorized users, ISPs, malicious parties, and advertisers, DNS over HTTPS (DoH) and DNS over TLS (DoT) are the two standard protocols developed. These protocols use encryption and decryption algorithms to interpret the actual request, as shown in Figure 4.

DoH queries and responses are encrypted in nature and communicate with the HTTP or HTTP/2 protocols instead

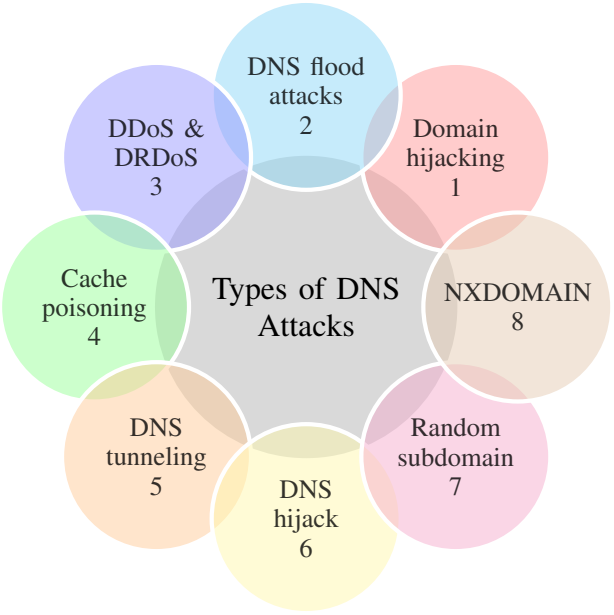

Figure 2. Types of DNS attacks.

of directly over UDP. Figure 5 illustrates the DoH operation. DoH ensures that a hacker should not snip or alter the actual DNS request or data. For US users, the Mozilla Firefox browser first integrated DoH and make it default in February 2020 [7]. DoT queries and responses are the alternatives of DoH. It also uses encryption to secure the DNS request. Its working is similar to HTTPS, which encrypts and authenticates communication between client and servers [8]. Both protocols are developed separately, and they have different (Request for Comments) RFCs documents. One major difference between DoH and DoT is port no., they use. DoT uses an 853 port, whereas DoH uses port 443. All the HTTPS traffic also uses these 443 ports. DoH and DoT both have their own advantages and disadvantages, like DoT is better for a network security point of view due to its ability to monitor network and block DNS queries. DoH is best suited for privacy protection issues due to its ability to hide DNS queries in a high network traffic flow. But as per the experts and ZDNet Security, DoH is not as effective as their developers claimed. In fact, it has so many issues, which raised various DNS-related problems. This paper is trying to test an automatic ML based prevention system if the DoH traafic is vulnerable or compromised. According to Haddon et al. [9], there are several possible ways of data ex-filtration using DoH. This DoH is also misused for malicious activities [10] and reported the first incidence of malware that deliberately uses DoH to hide its communication with Command-and-Control servers. The DoH works on an encryption-based service, even though it is vulnerable to several security and privacy issues [11]. The main problems with the DoH are:

- It does not detect in local Firewalls, IDS, etc.

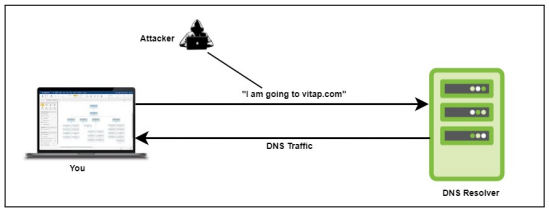

Figure 3. A normal unencrypted DNS query

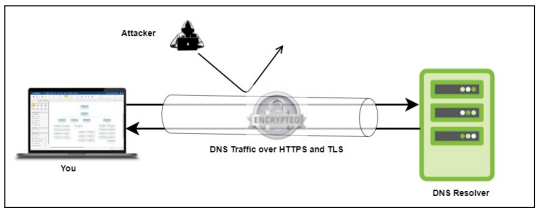

Figure 4. . DNS over TLS or DoT

- The assessment of DNS traffic is not easy under DoH.
- The visibility of DNS name are increases.
- Examination of security threats are very tough.
- As this is a new protocol handling troubleshooting is not easy, required skilled people.
- DNS blocking is not possible in DoH.

In this paper, we investigate DoH security using machine learning models. To achieve our motivation, we have used a recent dataset created based on DoH as well as normal DNS queries. In the datasets, both types of requests: benign and malicious are included. We are analyzing the encrypted traffic of DoH. Based on the analysis, we evaluate and reveal possible information (if any) for network security. The main objective is to find out the malicious and benign traffic using machine learning (ML) models with newly organized training and validation datasets. Here, our primary goal is to checks the DoH traffic that how much safe it is and can ML models be capable to filter the malicious request. The ML models are used to predict the malicious and benign DNS requests in DoH. The proposed system capable of detecting the malicious DNS request with high accuracy. Hence, this system may use for fighting against security breaches due to DoH. The key contributions are as follows:

- We proposed an ensemble learning framework to predict traffic is malicious or benign.
- The proposed model uses the benchmark dataset for training and testing.
- The proposed model predicting the malicious traffic on DNS over the DoH with high accuracy.

The rest of the paper is organized as follows: Section 2 reviews recent related papers. Section 3 explains the datasets and feature selection methods with all captured features. The malware or malicious traffic detection model is presented in Section 4. Section 5 discusses the performance and compares various ML models. The paper is concluded with future scope in Section 6.

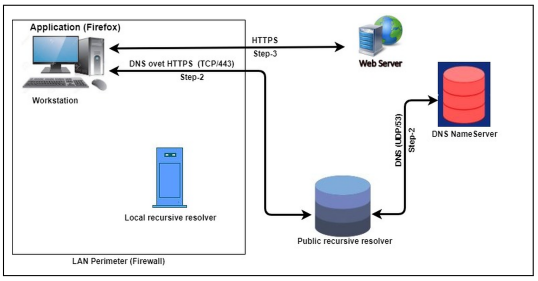

Figure 5. DNS over HTTPS (DoH)

2. Related Works

DoH and DoT are both very recent technology for Internet standards. However, some efficient works have been proposed and published that focused on various security aspects of it. Borgolte et al. [10] offer a common discussion about DoH with multiple areas like performance, security, and privacy. But the main issue with their work is that they have not analyzed the DoH at the network level. Böttger et al. [12] surveyed DoH with the help of accessing standard compliance and major features of the open general DoH server. They also compare different transports to secure DoH. In their work, they have emphasized the improvements of DoH over its predecessor and DoT. Wijenbergh et al. have presented a performance comparison of DoH and unencrypted DNS [13]. In their study, they have analyzed extra overhead found in DoH query.

Nijeboer has provided a privacy analysis of DoH traffic in his article [14]. In his research, Firefox is used for capturing the traffic in different time intervals. The main features used to filter the DoH traffic is: temporal features and packet sizes. This research also suggests a new padding scheme to the query to improve the privacy in DoH traffic. To filter the genuine request from the DoT traffic, Houser et al. [15] provided a fingerprint method to examine it. This method separates a real user and an attacker. Even though it works on encrypted DoT, information leakage possibilities are always there. Siby et al. [16] have considered a new feature set to attain the attacks related to DoH. Their analysis concluded that if attackers are resourceful, then padding methods are not fair enough to prevent it. Bumanglag et al. [8] have reviewed the problems of the DNS service and with malware exploitation in the context of these problems. They also examined the improvements of DNS security and how to filters malware from DNS traffic. The authors have given more importance to the DNS over HTTPS, which is favorable for an organization's security. Vekshin et al. [17] analyses encrypted traffic mainly related to DoH with the help of ML algorithms. They have used five ML classifiers and achieved a 99.9% success rate to

differentiate DOH clients accurately. Konopa et al. [18] presented an automated DoH traffic detection using ML techniques. This can be easily used in firewalls to detect any anomalies.

MontazeriShatoori et al. [19] have provided security concerns of DNS service and created a covert channel using tunneling data through DNS packets. They identify tunneling events that use DNS communications over HTTPS. They have designed a two-layered method to distinguish and portray DoH traffic using time-series classifiers. Singh and Roy [20] presented an ML-based scheme to predict a DoH traffic is malicious or benign. They used five popular ML Model such as: (i) Naive Bayes (NB), ii) Logistic Regression (LR), iii) Random Forest (RF), (iv) K-Nearest Neighbor (KNN), and (v) Gradient Boosting (GB) to distinguish the malware at DNS level in the DoH traffic. Huang has provided an inclusive study of DoH to downgrade the attacks [21]. He has used six different browsers and four different types of attacks. This article concluded that internet uses needs to revisit the standards and implementations of DoH.

Hjelm designed a real intelligence threat analysis framework [22] to detect DoH traffic during his research. To bypass basic security controls by DoH, he has tested that traffic several times. He has used a Mozilla Firefox, which is DoH enable browser for examining various news and entertainment websites. To create a logs file, Zeek IDS tools are applied based on the network traffic. Finally, he examines the DoH requests are coming from real clients or fake clients with a real intelligence threat analysis framework. Varshney et al. have presented a new approach for content filtering and phasing detections under DoH enable browsers [23]. They have exposed the DoH traffic by directly sniffing URLs from the RAM of end points/client machines instead of operating system DNS client. Although on DNS security concerns, several researches published but DoH is still in its early stages. However, it is today's need to come up with more secure DNS-based services to gain the trust of Internet users. It is also our duty to check all new protocols like DoH and DoT with all the measures and assured that it is safe for our Internet. This paper checks DoH security features with the help of machine learning. The next section discusses the complete methodology to detect malicious activities with ML algorithms.

3. DATASETS AND FEATURE SELECTION

A suitable dataset is an important requirement for an ML model. The superiority of the model is directly related to the conglomeration of data contained in the dataset. The dataset used in this research was taken from the source . To the best of our knowledge, this is the only source where publicly this dataset is available. Two separate files are provided by the source, namely: Benign.csv and Malicious.csv. During the data capturing process, the Malicious DNS server and benign DoH server are replicated in web browsers. Next, DoH tunnels generated the malicious DoH. The complete data capturing process can be seen in Figure 6.

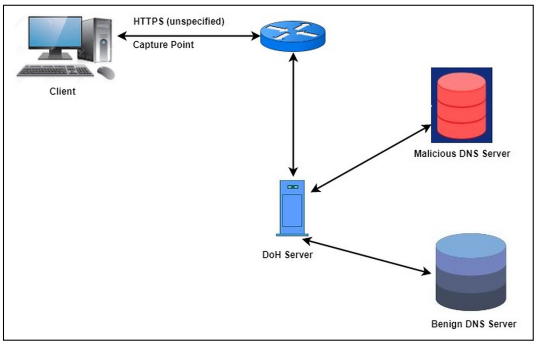

Figure 6. Dataset capturing process

In Malicious.csv, 249,836 samples are present, whereas, in Benign.csv, the number of samples is 19,807. To prepare the dataset, we have merged these two files. From the combined dataset, the null attributes are removed during the preprocessing steps. After removing the null attributes, the dataset has 269,299 useful samples that belong to Malicious and Benign classes—the dataset consisting of a large number of features. To select the relevant features, DoHMeter tool is used. The tool was developed using the Python library. The tool helps analyze and extract the useful feature from the PCAP file and produce a CSV file. The dataset consisting a large number of features set; among those, the relevant features are selected for this work. The list of the selected features with their description is shown in Table I.

4. PROPOSED METHODOLOGY

The aim of this research is to detect malicious activity over the DoH traffic. The working of the proposed ensemble learning model is shown in Figure 7.

As shown in Figure 7, the dataset is split into two parts, i.e., training and testing. The number of samples in training is 75% of the total sample, whereas we reserve 25% of samples for testing the model performance. To create an ensemble framework of the learning models, we have used three classifiers, namely; (i) Decision Tree [24], (ii) Logistic Regression [25], and (iii) K- Nearest Neighbour [26]. Apart from this, the Random Forest [27] ensemble learning classifier was also used.

The training samples are pass to each classifier to train the model. On the trained model, we have passed the test samples. Each classifier individually predicted the output class of the test samples; however, the predicted results are not used directly. Instead, a voting-based mechanism is used to decide the final output class of the test sample. This way, the ensemble of multiple classifiers is created. The experimental outcomes of these models are discussed in Section 5.

TABLE I. Selected features and their description

Feature Name	Description
Source Port	The source port number
Destination Port	The destination port number
Duration	The time gap between message generation and delivery
Flow Sent Rate	The rate of data transmission
Flow Bytes Received	The number of flow bytes received
Flow Received Rate	The rate of flow at which it was received
Packet Length Variance	The value of variance in packet length
Packet Length Standard Deviation	The value of standard deviation in packet length
Packet Length Mean	Mean value of packet length
Packet Length Median	Median value of packet length
Packet Length Mode	Mode Value of packet length
Packet Length Skew From Median	Skewed from the median packet length
Packet Length Skew From Mode	Skewed from the mode packet length
Packet Length Coefficient of Variation	Coefficient of Variation of packet length
Packet Time Variance	The value of the variance of Packet Time
Packet Time Standard Deviation	The value of the standard deviation of packet time
Packet Time Mean	The mean value of packet time
Packet Time Median	The median value of packet time
Packet Time Mode	The mode value of packet time
Packet Time Skew from Median	Skewed from the median of packet time
Packet Time Skew from Mode	Skewed from the mode of packet time
Packet Time Coefficient of Variation	The Coefficient value of the variation of packet time
Response/Request Time Variance	The variance value for request or response time difference
Response/ Request Time Standard Deviation	The standard deviation value for request or response time difference
Response/ Request Time Mean	The mean value for request or response time difference
Response/ Request Time Median	The median value for request or response time difference
Response/ Request Time Mode	The mode value for request or response time difference
Response/ Request Time Skew from Median	Skewed from the median for request or response time difference
Response/ Request Time Skew from Mode	Skewed from the mode for request or response time difference
Response/ Request Time Coefficient of Variation	The Coefficient of Variation for request or response time difference

5. EXPERIMENTAL RESULTS

The complete model is developed on a system having i5 processor and 16GB of RAM. For model development, Python libraries such as Numpy, sklearn, and others are used. The performance of the developed model is evaluated using metrics called: precision, recall, F1-score [28]. Mathematically, the precision, recall, and F1-score are defined as follows:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

Here: T_p represents the true positive, F_p represents the false positive, and F_n represents false negative. To begin the experiment, the K-nearest neighbor (KNN) classifier was first trained and then tested with 25% of the samples. The outcomes of the KNN classifier are shown in Table II.

TABLE II. Results obtained using KNN classifier

Class	Precision	Recall	F1-Score
Benign	0.98	0.93	0.95
Malicious	0.99	0.99	0.99

TABLE III. Results obtained using DT classifier

Class	Precision	Recall	F1-Score
Benign	0.97	0.97	0.97
Malicious	0.95	0.98	0.96

Next, the Decision Tree (DT) classifier use to predict the malicious attack. The outcomes obtained using the NB classifier are shown in Table III.

Finally, another machine learning classifier, namely Logistic Regression (LR) is applied. The outcomes of the LR classifier are shown in Table IV.

The results obtained using tested classifiers such as

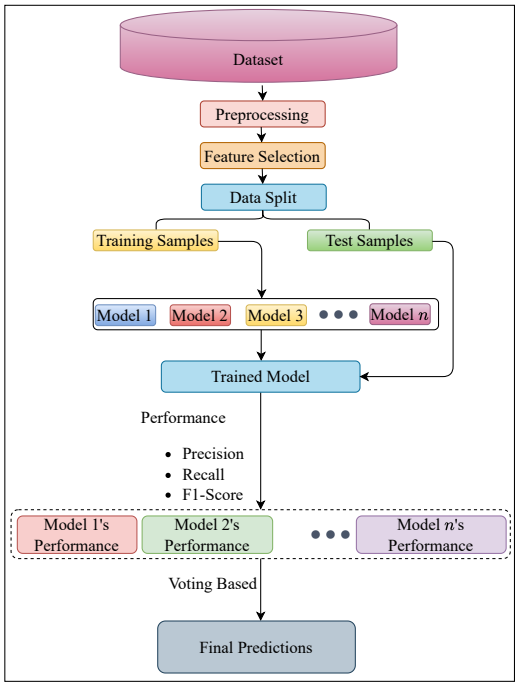

Figure 7. The proposed ensemble framework to detect Malicious DoH attack

TABLE IV. Results obtained using LR classifier

Class	Precision	Recall	F1-Score
Benign	0.86	0.69	0.77
Malicious	0.98	0.99	0.98

KNN, DT, and LR indicated the prediction for the malicious attack is acceptable; however, from the benign class, many instances are misclassified. The best result was obtained using the DT followed by KNN and LR. The DT classifier obtained a recall value for benign and malicious classes are 0.97 and 0.98, whereas the KNN and LR classifiers yielded the recall value of 0.93, 0.99, for benign and 0.99, 0.99 for the malicious class. The recall value obtained by LR classifier for benign class is 0.69, which is the lowest one.

The outcomes of these classifiers indicate that malicious attacks are correctly predicted. However, many non-malicious attacks are also predicted as malicious. However, for a good classification model, a minimum misclassification rate is needed. To overcome this issue, we have created an ensemble framework using KNN, DT, and LR classifiers. The ensemble learning-based model uses the voting mechanism to give the final prediction. Further, we have also used the Random Forest classifiers. In RF, many decision trees (DT) are constructed using the subset of features; the features are selected by applying the replacement technique. It means many DT may use the same features. Each DT predicts the final class of the test sample; finally, the RF classifier uses the voting concept and provides the decision. The outcomes of the formed ensemble and RF classifier are

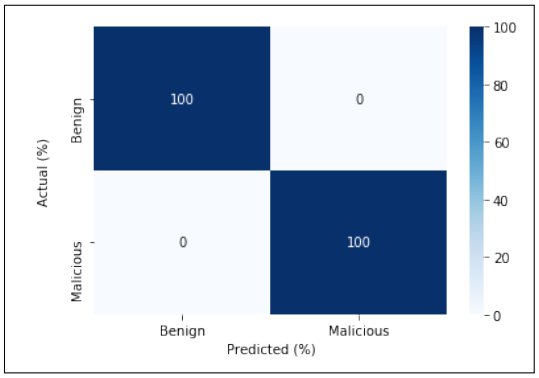

Figure 8. Confusion matrix obtained using RF classifier.

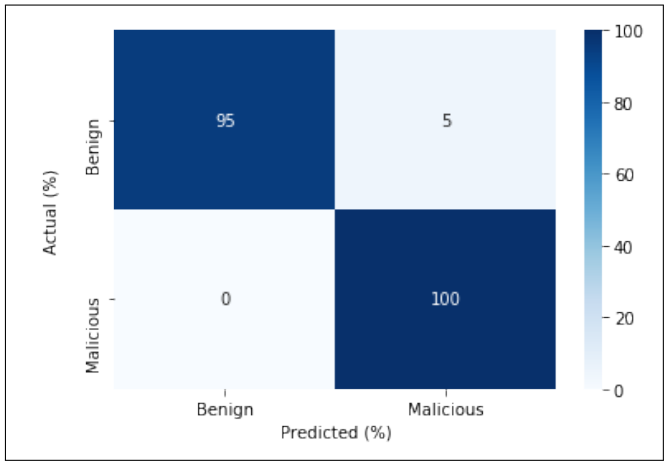

Figure 9. Confusion matrix obtained using created ensemble classifier

shown in Table V.

As shown in Table V, the RF classifier results are better than the created ensemble learning framework. The RF classifier misclassified a few samples of both the classes (three from benign and one from malicious). In contrast, the created ensemble framework misclassified 0.05 samples only. From benign class but, there is no misclassification in malicious class. This indicates that the RF classifier is the best classifier for the said problem. The confusion matrix obtained using the RF ensemble learning classifier and created ensemble framework and using the classifier is shown in Figures 8 and 9. The AUC-ROC curve obtained using the created ensemble learning is shown in Figure 10, and the AUC-ROC of the RF classifier is shown in Figure 11. The AUC-ROC curve of both ensemble techniques is the same.

6. Conclusion

The major use of DoH is to secure the DNS traffic and reduce the client's visibility using encryption methods over traditional DNS. As DoH protocol is very new, its security features are also known to all, which invites many

TABLE V. Results obtained using created ensemble and RF classifier

Classifier	Class	Precision	Recall	F1-Score
Created Ensemble	Class	0.98	0.95	0.97
	Benign	1.00	1.00	1.00
Random Forest	Class	1.00	1.00	1.00
	Benign	1.00	1.00	1.00

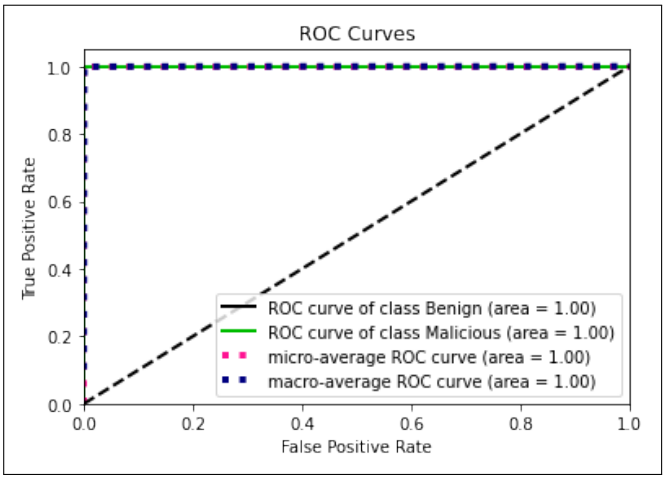

Figure 10. AUC-ROC curve obtained using formed ensemble classifier

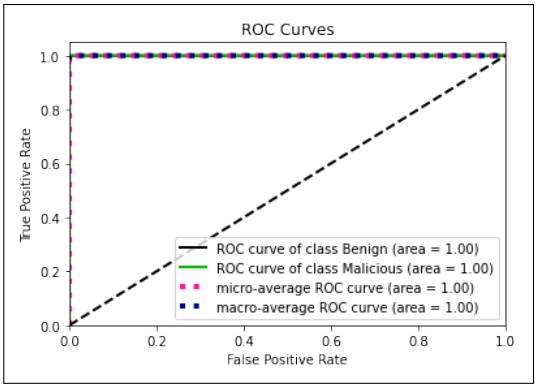

Figure 11. AUC-ROC curve obtained using RF ensemble classifier

security threats. This is one of the reasons we have selected these protocols to analyze. KNN, DT, LR, and RF- ML classifiers are applied to the selected dataset. Two different traffic benign and malicious DoH requests can be noticed during performance evaluation. The results analysis shows that the RF outperforms with the highest 100% accuracy and F1-measure in both the traffic. KNN and DT also perform well for malicious DoH. However, LR classifier performance is comparatively low as compared with other classifiers. The developed ensemble framework with KNN, DT, and LR classifiers also receive good accuracy for both classes. However, it is lesser than that of the RF classifier. Hence, it can be suggested that an ensemble learning-based RF classifier is the best alternatives for this problem.

The current study utilizes the available features directly. This research uses a single dataset with pre-determined features. In the future, the proposed work can be extended by testing the model performance with other datasets as well as including the additional features to check the model's robustness.

Acknowledgment

This paper is a revised and expanded version of an article entitled 'Detecting Malicious DNS over HTTPS Traffic Using Machine Learning' presented at the International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT-2020), Bahrain, University of Bahrain, 20–21 December 2020 [29].

References

[1] A. Gulbrandsen, P. Vixie, and L. Esibov, "A dns rr for specifying the location of services (dns srv)," RFC 2782, February, Tech. Rep., 2000.

[2] A. Shaikh, R. Tewari, and M. Agrawal, "On the effectiveness of dns-based server selection," in *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No. 01CH37213)*, vol. 3. IEEE, 2001, pp. 1801–1810.

[3] D. Dagon, M. Antonakakis, K. Day, X. Luo, C. P. Lee, and W. Lee, "Recursive dns architectures and vulnerability implications." in *NDSS*, 2009.

[4] S. Ariyapperuma and C. J. Mitchell, "Security vulnerabilities in dns and dnssec," in *The Second International Conference on Availability, Reliability and Security (ARES'07)*. IEEE, 2007, pp. 335–342.

[5] T. H. Kim and D. Reeves, "A survey of domain name system vulnerabilities and attacks," *Journal of Surveillance, Security and Safety*, vol. 1, no. 1, pp. 34–60, 2020.

[6] J. Van Heugten, "Privacy analysis of dns resolver solutions," *Master's thesis, Master of System and Network Engineering, University of Amsterdam, The Netherlands*, 2018.

[7] C. Cimpanu, "First-ever malware strain spotted abusing new doh (dns over https) protocol," 2019.

[8] D. A. Haddon and H. Alkhateeb, "Investigating data exfiltration in dns over https queries," in *2019 IEEE 12th International Conference on Global Security, Safety and Sustainability (ICGS3)*. IEEE, 2019, pp. 212–212.

[9] C. Cimpanu, "Here's how to enable doh in each browser, isps be damned," 2020.

[10] K. Borgolte, T. Chattopadhyay, N. Feamster, M. Kshirsagar, J. Holland, A. Hounsel, and P. Schmitt, "How dns over https is reshaping

privacy, performance, and policy in the internet ecosystem," *Performance, and Policy in the Internet Ecosystem (July 27, 2019)*, 2019.

[11] K. Bumanglag and H. Kettani, "On the impact of dns over https paradigm on cyber systems," in *2020 3rd International Conference on Information and Computer Technologies (ICICT)*. IEEE, 2020, pp. 494–499.

[12] T. Böttger, F. Cuadrado, G. Antichi, E. L. Fernandes, G. Tyson, I. Castro, and S. Uhlig, "An empirical study of the cost of dns-over-https," in *Proceedings of the Internet Measurement Conference*, 2019, pp. 15–21.

[13] J. Wijenbergh, V. Moonsamy, R. van Rijsdijk-Deij, and D. D. Kuijsters, "Performance comparison of dns over https to unencrypted dns," 2019.

[14] F. Nijeboer, "Detection of https encrypted dns traffic," B.S. thesis, University of Twente, 2020.

[15] R. Houser, Z. Li, C. Cotton, and H. Wang, "An investigation on information leakage of dns over tls," in *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, 2019, pp. 123–137.

[16] S. Siby, M. Juarez, C. Diaz, N. Vallina-Rodriguez, and C. Troncoso, "Encrypted dns–¿ privacy? a traffic analysis perspective," *arXiv preprint arXiv:1906.09682*, 2019.

[17] D. Vekshin, K. Hynek, and T. Cejka, "Doh insight: Detecting dns over https by machine learning," in *Proceedings of the 15th International Conference on Availability, Reliability and Security*, 2020, pp. 1–8.

[18] M. Konopa, J. Fesl, J. Jelínek, M. Feslová, J. Cehák, J. Janeček, and F. Drdák, "Using machine learning for dns over https detection," in *ECCWS 2020 20th European Conference on Cyber Warfare and Security*. Academic Conferences and publishing limited, 2020, p. 205.

[19] M. MontazeriShatoori, L. Davidson, G. Kaur, and A. H. Lashkari, "Detection of doh tunnels using time-series classification of encrypted traffic," in *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. IEEE, 2020, pp. 63–70.

[20] S. K. Singh and P. K. Roy, "Detecting malicious dns over https traffic using machine learning," in *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT)*. IEEE, 2020, pp. 1–6.

[21] Q. Huang, D. Chang, and Z. Li, "A comprehensive study of dns-over-https downgrade attack," in *10th {USENIX} Workshop on Free and Open Communications on the Internet ({FOCI} 20)*, 2020.

[22] D. Hjelm, "A new needle and haystack: Detecting dns over https usage," *SANS Institute, Information Security Reading Room, August*, 2019.

[23] G. Varshney, P. Iyer, P. Atrey, and M. Misra, "Evading doh via live memory forensics for phishing detection and content filtering," in *2021 International Conference on COMmunication Systems & NETworkS (COMSNETS)*. IEEE, 2021, pp. 1–4.

[24] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE transactions on systems, man, and cybernetics*, vol. 21, no. 3, pp. 660–674, 1991.

[25] C. M. Bishop, "Pattern recognition," *Machine learning*, vol. 128, no. 9, 2006.

[26] Y. Liao and V. R. Vemuri, "Use of k-nearest neighbor classifier for intrusion detection," *Computers & security*, vol. 21, no. 5, pp. 439–448, 2002.

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

[28] P. K. Roy, A. K. Tripathy, T. K. Das, and X.-Z. Gao, "A framework for hate speech detection using deep convolutional neural network," *IEEE Access*, vol. 8, pp. 204 951–204 962, 2020.

[29] S. K. Singh and P. K. Roy, "Detecting malicious dns over https traffic using machine learning," in *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT)*, 2020, pp. 1–6.

Sunil kumar Singh is currently working as an Assistant Professor in the School of Computer Science and Engineering at VIT-AP University, Vijayawada, India. He has done his Ph.D. in Computer Science and Engineering form National Institute of Technology Patna, India in 2018. He received the M. Tech and B. Tech degrees in Computer science and Engineering and Information Technology, both from Kalyani Government Engineering College, Kalyani, India in 2010 and 2007, respectively. He has over 30 publications in various National/International Journals & Conferences (viz. IEEE, ACM, Springer and Elsevier). He is also the reviewer of several reputed journals indexed in SCI, SCIE and Scopus. He is also in the Program Committee of various National/International Conferences. He has delivered expert talks and guest lectures at various prestigious institutes. His research area includes Wireless Sensor Networks, Internet of Things, MANETs etc.

Pradeep Kumar Roy received the B. Tech degree in Computer Science and Engineering from BPUT University Odisha. He received his M. Tech and Ph.D. degree in Computer Science and Engineering from the National Institute of Technology Patna in 2015 and 2018, respectively. He received a Certificate of Excellence for securing a top rank in the M. Tech course. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Indian Institute of Information Technology (IIIT) Surat, Gujarat, India. He also worked in Vellore Institute of Technology, Vellore, Tamil Nadu, India. His area of specialization straddles across question answering, text mining and information retrieval, social network, and wireless sensor networks. He is part of the technical program committee and chaired many technical sessions of International Conferences. He has published articles in different journals, including IEEE Transaction on Artificial Intelligence, Neural Processing Letters, IJIM, Neural Computing and Applications, Future Generation Computer Systems, and others. He has also published the conference proceedings in various international conferences.