

## TOPICAL REVIEW

# Visualizing Realistic Benchmarked IDS Dataset: CIRA-CIC-DoHBrw-2020

MOHAMMAD HAFIZ MOHD YUSOF<sup>1</sup>, AKRAM A. ALMOHAMMEDI<sup>2,3</sup>,  
VLADIMIR SHEPELEV<sup>1,2</sup>, AND OSMAN AHMED<sup>4</sup>

<sup>1</sup>Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Tapah, Perak 35400, Malaysia

<sup>2</sup>Automobile Transportation Department, South Ural State University, 454080 Chelyabinsk, Russia

<sup>3</sup>Electrical and Electronic Engineering Department, Karabük University, 78050 Karabük, Turkey

<sup>4</sup>Al Safwah Centre, SecurEyes Ltd., Riyadh 12223-7656, Saudi Arabia

Corresponding author: Mohammad Hafiz Mohd Yusof (hafizyusof@uitm.edu.my)


This work was supported in part by the Research Management Centre (RMC), Faculty of Computer and Mathematical Sciences (FSKM), Universiti Teknologi MARA (UiTM); and in part by the Research Project under Grant FRGS/1/2021/ICT07/UITM/02/3.

**ABSTRACT** Intrusion Detection System (IDS) dataset is crucial to detect lateral movement of cyber-attacks. IDS dataset will help to train the IDS classifier model to achieve earliest detection. A good near-realism public dataset is essential to assist the development of advanced IDS classifier models. However, the available public IDS dataset has long been under scrutiny for its practicality to reflect real low-footprint cyber threats, render real-time network scenario, reflect recent malware attack over newly developed DoH protocol, disregard layer 3 information and finally publish contradictory results of classification and analysis between various studies which makes it non-reproducible and without shareable results. This problem can be resolved by sophisticatedly visualizing a new realistic, real-time, low footprint and up-to-date benchmarked dataset. Visualization helps to detect data deformation before designing the optimized and highly accurate classifier model. Therefore, this study aims to review a new realistic benchmarked IDS dataset and apply sophisticated technique to visualize them. The review starts by carefully examining production network features. These are then compared with various well-established public IDS datasets. Many of them are static, unrealistic meta-features and disregard source and destination Internet Protocol (IP) information except CIRA-CIC-DoHBrw-2020 dataset. The study then applies Eigen Centrality (EC) technique from the graph theory to visualize this layer 3 (L3) information. Finally, using various visualization techniques such as Principal Component Analysis (PCA) and Gaussian Mixture Model (GMM), the study further analyzes and subsequently visualizes the data. Results show that the CIRA-CIC-DoHBrw-2020 simulated recent malware attack and has a very imbalanced dataset which reflects the realistic low-footprint cyber-attacks. The centrality graph clearly visualizes IPs that are compromised by recent DoH attack in real-time, and the study concludes decisively that smaller packet length of size 1000 to 2000 bytes is to fit an attack trait.

**INDEX TERMS** Intrusion detection system (IDS), IDS dataset review, imbalanced dataset, data visualization, machine learning in cybersecurity.

## I. INTRODUCTION

Intrusion Detection System (IDS) is always concealed by connected, ever-changing zero day cyber-attack. This stealthy attack is almost undetectable by conventional IDS technology and firewalls [1]. Hence it is critical to develop an

The associate editor coordinating the review of this manuscript and approving it for publication was Vicente Alarcon-Aquino .

advanced monitoring system and irreplaceable solution to detect unknown malware [2], [3].

From the literature as shown in Table 1, most of the major works focus on developing IDS classifiers and also it apparent that fewer works have been done particularly in the area of IDS dataset review and visualization. Hence, preliminary systematic reviews on the public IDS datasets were conducted at the early stage of this research. Various several notable public datasets have been surveyed which expose several advantages

and disadvantages. It is revealed that the existing IDS dataset has some issues; 1) the deficiencies to reflect modern network threats; 2) an augmented minority dataset has limited properties to replicate the nature of network attacks. Hence, real-time capture is needed; 3) none of these public datasets have been simulated against the recent Domain Name System over HTTPS (DoH) protocol. Most public datasets demonstrate the classical DDoS attacks and its variants; 4) none of these public datasets have included layer 3 information in its column's features; 5) finally, a contradictory analysis and classifying results have been shown in a few studies.

Those problems are further expounded from hereon. Various public datasets have been examined and some encompass various updated intrusion footprints. However, many of them were classical and well-known for their deficiencies to reflect modern network threats, as highlighted by [4], [5], [6], [7], and [8]. The authors in [4] also concluded that current IDS datasets suffer from realistic network traits.

Furthermore, many techniques were applied to oversample minority class or downsample the majority class [9], [6]. This is done to increase efficiency of the IDS model. However, imbalanced dataset usually has low threats of footprint. This will certainly be reflected in modern network threats, in which both attack and normal traffic are concealed on top of each other. Hence, rendering a very low threats footprint is needed and getting it augmented is a problem. On the other hand, instead of augmenting the dataset, a few studies suggested a real-time packet capture is needed, as in [7] and [4]. This can be done by capturing traffic from a real network set-up or through injected or simulated traffic. This is to ensure that the dataset is reproducible, shareable and has similar properties to the production network [4]. The absence of real-time and real network datasets in many IDS researches is still prevalent.

Subsequently, there is a new protocol called DNS over HTTPS (DoH) that was introduced in 2018 by Internet Engineering Task Force (IETF). It was published as Request For Comment (RFC) document number 8484 (RFC8484). From that development, there are more sophisticated exploits being introduced to compromise DNS over this DoH protocol. From Table 1, it is noticeable that DDoS and various DDoS-related attacks were analyzed by many researchers. However, none of them have simulated the attack against the recent DoH protocol.

PCAP's features are labels for ordinary network traffic. Obviously, PCAP's features, as shown in Table 2, have no attack and normal label. This requires unsupervised type of Machine Learning (ML) trainings. Many popularly cited public datasets have demonstrated different meta-data or raw-data features. A few features have some resemblance with the PCAP's features. However, many have totally different feature sets. Based on Table 2, none of them has the source IP and destination IP (layer 3 information) as one of the feature sets. Hence new coefficient values are needed so as to design an unsupervised IDS model.

It is worth highlighting that a few studies have shown contradictory results from its classification models. For instance, some studies have shown that RF has achieved highest accuracy as in [10] contrast to the report from [11] which reported that NB had achieved excellent performance. The irony is that both use similar CICIDS2017 standard dataset. Some contradictory analyses were also spotted. For instance, NSL-KD dataset is used to detect low-frequency attacks. It is well-known that NSL-KDD was not an inclusive depiction of a contemporary low footprint attack, as stated in [8].

This suggests that this domain might have non-reproducible and shareable results, as suggested by [4]. It also indicates that various classification models are event-specific and have to be handled case by case. However, the good news is that this gives plenty of room and opportunities for future improvements. This is particularly true in the area of data pre-processing and data visualizations. This issue serves as one of the reasons why this study is conducted.

In this study, since a real attack is made up from multiple frames and network packets, visualization through statistical analysis and machine learning approach is introduced. This will reduce the misclassification and contradictory analysis issues as highlighted in problem number 5. The discussion on visualization approach of this study is further expounded from hereon. Visualization in essence helps to dissect these complex network datasets into visual format. This will assist during the training process of IDS classifiers. It will eventually assist in the development of an advanced classifier that applies state-of-the-art machine learning techniques. Visualizing dataset also helps to detect data deformation before it is trained by the classifier model to achieve an optimized, highly accurate model. From the literature, a few pre-processing techniques were applied, such as PCA, *t*-SNE, *k*-Means, ADASYN, SMOTE, min-max method, Shrunken centroid and a few others. These techniques were applied for various reasons. For instance, to resolve issues in imbalanced dataset, for feature reduction and notwithstanding for visualization. Since lack of layer 3 information is apparent in the previous studies, as highlighted in problem number 4, this feature will be visualized in this study. Layer 3 or network layer is an essential feature in networking. Many underlying patterns can be revealed out of this feature. Due to that, the Eigen Centrality (EC) visualization concept from the graph theory will be applied. The outcome of this analysis will contribute to the discovery of centrality's degrees. This centrality pattern is drawn from the interaction of these IP addresses. This eventually will notify the source of lateral movement or the attack vector.

Several other approaches are utilized in this study to enhance the visualization analysis. These include PCA and GMM, a type of *k*-Means analysis. Pre-processing techniques for visualization like PCA and GMM are crucial to address data deformation problems that might exist prior testing the dataset against the classifier model [6]. Notwithstanding, various visualization techniques like bar plot, skewness and outlier distributions were also applied. As stated before, since

production network has no label, this approach will help to highlight a few coefficient values which helps as a target feature in the unsupervised IDS classifier model. In this study, meta-data features or raw-data features are treated equally. There is no differentiation between processes flow information and raw label.

On problem number 1; deficiencies to reflect modern network threats, problem number 2; lack of dataset that reflects real-time or real-network and problem number 3; lack of study on malicious attack over DoH, protocol involved in searching of **realistic** dataset. This dataset must have imbalanced properties to realistically portray the low footprint trait. The dataset must reflect real-time and real-network features and finally the dataset must furnish reached data over the attack on DoH protocol. For this study, after a thorough examination, we rely on the closest to the real environment dataset, which is CIRA-CIC-DoHBrw-2020. However, in the future, a real ground-truth dataset through real production network set-up should be initiated. Formerly CICIDS2017 dataset was claimed to be the closest related public dataset to production network [11]. Nevertheless, when comparable to the PCAP features, it satisfies most of the labels except source and destination IP. Missing this layer 3 information makes this dataset lacking up-to-date information of real-network traffic. Layer 3 information is a vital feature in network communication. This is the major concern highlighted in problem number 1 and number 2. In a nutshell, none of these public datasets include network layer information (OSI layer 3) except for CIRA-CIC-DoHBrw-2020.

CIRA-CIC-DoHBrw-2020 is a relatively new dataset that also simulates modern attack on DoH. Visualizing CIRA-CIC-DoHBrw-2020 will subsequently expose the essence of DoH attack. This will help to resolve problem number 3. As far as this research is concerned, there are very limited studies on IDS dataset survey, review or data visualization especially on CIRA-CIC-DoHBrw-2020 dataset. Hence, this study aims to review this new realistic benchmarked IDS dataset. Then sophisticatedly visualize CIRA-CIC-DoHBrw-2020 using EC, PCA and GMM. This is to provide intrinsic details which may assist on the development of the IDS model. Finally, the contributions of this work are listed below:

- 1- Design EC visualization technique on realistic real-time IDS dataset which involves layer 3 information. Layer 3 information is mandatory in any cyber-attack analysis and network intrusion studies.
- 2- Introduce a few coefficient features to assist the training process of the unsupervised IDS model. Since realistic real-time traffic has no attack and benign label, the chosen labels are essential.
- 3- Introduce GMM and PCA pre-processing and visualization techniques over low cyber-attack footprints.
- 4- Introduce time-series pre-processing and visualization techniques over real-time dataset to reduce the chances of contradictory analysis.

- 5- Highlight eminent problems in current public IDS dataset and how it contributes to the contradictory analysis issues. Subsequently, state the urgency of having a realistic real-time IDS dataset.

## II. RELATED WORKS

This section systematically reviews various related studies on IDS dataset. It contains three sections. Section A explains the gaps of the studies. It summarizes the related works and describe the gaps as an extended problem statement. Then, section B clarifies the benchmarked dataset of CIRA-CIC-DoHBrw-2020. Finally, section C explains the layer 2 frame which clarifies the differentiation between benchmarked dataset and ground-truth dataset column labels.

The authors in [6] visualized security dataset of UNSW-NB15 on malicious DoS attacks. They applied several pre-processing algorithms such as PCA, *t*-SNE, *k*-Means distance cluster, shrunken centroid, Elastic Net Algorithm and Manhattan distance. These were used to examine IDS dataset. They discovered two main issues; 1) an imbalanced dataset 2) an overlapped label. This information was crucial to address problems that might exist prior testing the dataset against the developed classification model. However, the study did not process datasets that were specific to network infrastructure such as IP address.

In [5], the authors offered a review on IDS technology especially on classification models. Part of the work was to compare on benchmarked Network IDS dataset, for instance, NSL-KDD, ADFA-LD/WD, AWID, UNSW-NB15, CIC-IDS 2017, CIC-DDoS2019 and BoT-IOT. ADFA-LD/WD dataset from Australian Defense Force Academy of host-based system calls traces from Linux and Windows operating system. AWID from Aegean contains labelled Wi-Fi dataset. CICIDS 2017 dataset showed attacks on various DoS, DDoS. CIC-DDoS2019 contained 88 features with normal and assorted types of DDoS attacks and finally BoT-IOT also demonstrated various attacks on DoS and DDoS. The data were summarized in a simple tabular form. ADFA, AWID, UNSW-NB15 and CICIDS contained deficiencies and CIC-DDoS2019 and BOT-IoT dataset encompassed latest intrusion traits [5].

The authors in [12] proposed an intrusion detection model that integrates deep learning technique. NSL-KDD and CIS-IDS2017 datasets were used to train and test the model. Both have been adopted by many studies during the evaluation process. Adaptive Synthetic Sampling (ADASYN) was applied to resolve the issue on imbalanced dataset. Some other pre-processing steps that were applied include *k*-Mean and *t*-SNE. The classifier was modelled by using Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) and Random Forest (RF) for binary classification. The main objective of this study is an IDS classification model.

Similarly, authors in [13] introduced a classification model that applies improved CNN, which is known as Split Module CNN (SPCCNN) and ADASYN which is used to augment

dataset distribution. The well-known NSL-KDD dataset was used, which is extremely unbalanced. Therefore, ADASYN algorithm augmented minority features. Imbalanced dataset weakened the training process of the model. However, imbalanced dataset certainly reflected modern network threats. Both attack and normal traffic were concealed on top of each other and rendered a very low threats footprint.

In [4], the authors highlighted current threat taxonomy in IDS researches and concluded that current IDS datasets suffered from realistic network threats practicality. They suggested that datasets could either be captured from a real network set-up or through injected or simulated traffic. This dataset must be reproducible, similar to production network and shareable. The analysis in [4] has helped to improve the creation of close-to-real-world datasets and subsequently improved the classifier efficiency. CAIDA DDoS, Waikato, ISCXIDS2012, CTU-13, STA2018, Botnet dataset, TUIDS, Booters, DDoSTB, Unified Network Dataset, ADFA-IDS are amongst the new observed datasets.

The authors in [7] offered a study on the development of real-time web intrusion using CNN and LSTM. They realized no study had been done to analyze large scale network traffic in real time. Dataset of fixed real-time HTTP traffic was normalized using Spatial Feature Learning (SFL) technique. Finally encoded UTF-8 characters were extracted. This experiment was repeated using two public datasets of CSIC-2010 and CICIDS2017. It is stated that NSL-KDD dataset is not suitable to train real-time detection as it deals with metadata. Metadata processes statistics information from a raw input and generates a new dataset. According to [7], most of the published datasets have repetitive features and inefficient attack traits to reflect the recent web-attacks trend.

The authors in [11] extensively reviewed the efficacy of the anomaly IDS model that uses various algorithms and techniques. Then, the performance of the selected machine learning approach was tested over CICIDS2017 dataset. This dataset is claimed to be the closest dataset related to production network. The authors stated that current studies on anomaly IDS are not meant for benchmarking the modelling techniques, the methodology and the algorithm, particularly on deep learning. The study in [11] also highlighted that  $k$ -NN, Naïve Bayes (NB) had achieved excellent performance.

The authors in [10] analyzed significant and relevant features to improve anomaly detection and also to reduce execution time. Information Gain (IG) was the chosen feature selection technique which applies in CICIDS-2017. This involves ranking and segmenting the features following its smallest possible values. The reduced dataset was then tested over RF, Bayes Net (BN), NB and J48 classifier algorithms. Paper in [10] also showed that RF has achieved highest accuracy contrary to the report from [11] which demonstrated excellent performance from NB classifier.

This shows this domain has non-reproducible and shareable results, as suggested by [4]. It also indicated that various classification models were event-specific and have to

be handled case by case. Hence, there is still plenty of room for future improvements in this domain, particularly on IDS dataset and visualizations.

In [9], the authors applied data generation model named Synthetic Minority Oversampling Technique (SMOTE) to increase efficiency of the IDS model. Data from minority class were oversampled to increase the average data size. This method basically used  $k$ -NN algorithm to augment new data. The final machine learning model with a few fixed hyper-parameters was then tested on CSE-CIC-IDS2018 dataset. There was obviously an imbalanced data size in each class.

The authors in [14] worked on intrusion detection machine learning model over imbalanced dataset. They proposed a Difficult Set Sampling Technique (DSSTE) algorithm to separate imbalanced dataset into difficult set and easy set. The algorithm used “edited” Nearest Neighbor which subsequently applied  $k$ -NN to compress the majority samples. This compressed majority was then combined to the easy set to produce a whole new dataset. To verify the performance of the classifier, CSE-CIC-IDS2018 and NSL-KDD were used to train the model. The authors used  $t$ -SNE to visualize these datasets.

The authors in [15] proposed a detection model called SAVAER-DNN which applied auto-encoder with regularization technique to detect low-frequent attacks. The model was evaluated against benchmarked dataset from NSL-KDD variants and UNSW-NB15. The work in [15] then applied Uniform Manifold Approximation and Projection (UMAP) techniques to visualize spatial distribution of original and synthetic samples. A few pre-processing techniques on data scaling and one-hot data encoding were performed.

The authors in [16] proposed an intrusion detection that applied a technique known as Intrusion Detection Based on Feature Graph (IDBFG). It started with generating filtered normal connections using grid partitions and subsequently recorded those patterns with a graph structure. The behavioral pattern arising from the graph indicates intrusion traits. The model was evaluated against KDD-Cup 99 dataset, the old version of NSL-KDD. The result was compared against Support Vector Machine (SVM) and Decision Tree (DT). However, NSL-KDD is not an inclusive depiction of a contemporary low footprint attack environment [8].

The authors in [17] proposed network IDS model based on bio-inspired metaheuristic algorithm. The first objective was to get optimized features for the input dataset. This model applied various bio-inspired algorithms such as Multiverse Optimizer (MVO), Moth-Flame Optimization (MFO), Grey Wolf Optimizer (GWO), Bat Algorithm (BAT) and Firefly Algorithm (FFA). The next objective was to classify the generic attacks through SVM, J48 and DT. The model was trained with UNSW-NB15 dataset.

In [18], the authors developed a hybrid network IDS model to address low false-negative and high false (cited as per the text) rates.

The process included three phases, which were 1) Data normalization using min-max method, 2) Feature and 3) Attacks'



TABLE 1. Summary of related studies.

Author	Pre-processing Technique (visualizing dataset)													Primary Attack Type				Benchmarked Dataset	Main Contribution					Year
	PCA	t-SNE	K-Means variants	Tabular, Pie, Bar chart, Histogram	ADASYN (Data Augment)	Spatial Feature Learning (SFL)	SMOTE	UMAP	Grid Cells, Feature Graph	Firefly Algorithm	min-max method	DoH	DoS/DDoS	Web (Botnet, XSS)	Brute Force (SSH, FTP, Heartbleed)	Data Visualization	Feature Selection/		IDS Review	IDS Model				
Zoghie et al. [6]	✓	✓	✓										✓			UNSW-NB15	✓				2021			
Ozkan-Okay et al. [5]				✓									✓	✓	✓	NSL-KDD, CIC-DDoS2019, CIC-IDS 2017, BOT-IoT			✓		2021			
Liu Chao et al. [12]		✓	✓		✓								✓			NSL-KDD, UNSW-NB15				✓	2021			
Hu Zhiqian et al. [7]					✓								✓			NSL-KDD				✓	2020			
Hindy Hanan et al. [1]				✓									✓	✓	✓	CAIDA DDoS, Waikato, ISCXIDS2012, CTU-13, STA2018, Botnet dataset, TUIDS, Booters, DDoSTB, Unified Network Dataset, ADFA-IDS			✓		2020			
Kim Aechan et al. [8]						✓							✓	✓	✓	CICIDS2017				✓	2020			
Maseer Z K et al. [2]				✓									✓	✓	✓	CICIDS2017			✓		2021			
Kurniabudi et al. [3]				✓									✓	✓	✓	CICIDS2017		✓			2020			
Karatas Gozde et al. [9]							✓						✓			CSE-CIC-IDS2018		✓			2020			
Liu L et al. [10]		✓											✓			CSE-CIC-IDS2018 and NSL-KDD		✓			2021			
Yang Yangqing et al. [11]								✓					✓			NSL-KDD variants and UNSW-NB15				✓	2020			
Yu Xiang et al. [12]									✓				✓			KDD-CUP99				✓	2019			
Mehmod M et al. [18]										✓			✓			NSL-KDD				✓	2021			
Almomani O [7]												✓	✓			UNSW-NB15				✓	2021			
Hao X et al. [14]										✓			✓			KDD-CUP99				✓	2020			

TABLE 2. PCAP labels.

PCAP Features vs. Public Dataset	PCAP Labels								
	Time	Source IP	Destination IP	Protocol i.e 443	Frame Length (bytes on wire)	Source Port	Destination Port	Window Size	TCP Segment Length
NSL-KDD	✓			✓				✓	
UNSW-NB15	✓			✓		✓		✓	
CIC-IDS2017	✓			✓	✓		✓	✓	✓
CSE-CIC-IDS2018	✓			✓	✓		✓	✓	✓
CIRA-CIC-DoHBrw-2020	✓	✓	✓			✓	✓	✓	

detection and categorization process by using Fine Gaussian SVM (FGSVM) and Adaptive Neuro-Fuzzy System (ANFIS) technique.

NSL-KDD was used to perform the training and testing process.

The authors in [19] offered a cloud network intrusion model based on Bi-LSTM and attention mechanism. This was claimed as an effective measure to address the problem of learning attack pattern. Particularly attacks in massive and high dimensional data. This massive data with high dimensionality can be found in the complex and variable nature of production network traffic. In [19], public dataset KDDCup 99 was used to analyze the efficacy of the IDS classifier. Data first were normalized by using min and max method. However, according to [8], KDDCup99 suffers from redundant records in its training set.

Table 1 summarizes the important characteristics from the past related works. The discussion is available in the following section, which establishes the study gaps.

A. SUMMARY OF RELATED STUDIES (EXTENDED PROBLEM STATEMENT)

From Table 1, most of the major works were done on developing IDS classifiers and obviously fewer works have been seen particularly in the area of IDS dataset review and visualization. Those classifier models manipulate various IDS datasets, which is discussed in the next paragraph. A few pre-processing techniques were applied on previous works such as PCA, t-SNE, k-Means, ADASYN, SMOTE, min-max method and a few others. These techniques were applied for

various reasons, for instance, to resolve issues on imbalanced dataset, for features reduction and also for visualization. Since network intrusion involves Internet Protocols (IPs), hence a graph model is an essential technique. None of the previous works attempted to visualize the dataset by using graph model.

Various standards and public IDS datasets have been examined, which are UNSW-NB15, NSL-KDD, CIC-DDoS2019, CIC-IDS 2017, BOT-IoT, CAIDA DDoS, Waikato, ISCX-IDS2012, CTU-13, STA2018, Botnet dataset, TUIDS, Booters, DDoSTB, Unified Network Dataset, ADFA-IDS, CICIDS2017, CSE-CIC-IDS2018 and the KDD-CUP99. These datasets were mainly to simulate attack traffic over computer networks. Amongst them, NSL-KDD, UNSW-NB15, CIC-IDS2017, CSE-CIC-IDS2018 were popularly cited many times. These datasets were specifically to demonstrate DDoS attacks and their variants. Some of them encompassed latest intrusion traits and some of them were classic and well-known for deficiencies to reflect modern network threats.

For instance, NSL-KDD was not an inclusive depiction of a contemporary low footprint attack environment and KDD-Cup99 suffered from redundant records and many others [8]. Some web attacks datasets have repetitive features and are inefficient to reflect recent web-attacks trends [8]. On the other hand, [4] concluded that current IDS datasets suffer from realistic network threats practicality.

Some datasets show imbalanced traits between normal and attack's label and between attack and another attack's label. Imbalanced dataset weakens the training process of the model [13]. However, imbalanced dataset really reflects modern network threats. Both attack and normal traffic were concealed on top of each other and render a very low threats footprint. This is the characteristic of a stealth attack.

Next, Table 2 depicts the compatibility report between these public features set and the real production network feature set denoted as PCAP features. Here CICIDS2017 dataset, which is considered as the closest related dataset to production network, as stated in [11] when comparable to the PCAP features, satisfies most of the labels except source and destination IP. This real-world column features of PCAP labels are usually extracted from the Wireshark, an analysis tool API for network that sniffs frames information which enables for deep packet inspection. The frame details are as depicted in Fig. 2. The explanation of the frame or layer 2 information is available in section C.

From Table 2, duration, protocol\_type, src\_byte and dst\_byte from the NSL-KDD dataset are tuples that accordingly have resemblance to time, protocol and window size of the PCAP labels. Dur, proto, dwin, is\_sm\_ips\_ports, ct\_src\_dport\_ltm for UNSW-NB15 similar to time, protocol, window size and source port. DstPort, proto, timestamp, Pkt-SizeAvg and a few others flow bytes information for CIC-IDS2017 can be accounted for representing destination Port, protocol, time and window size of the PCAP features. This

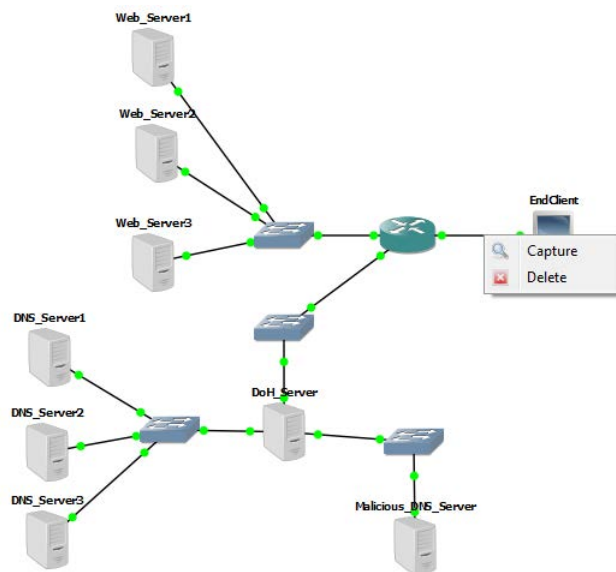


FIGURE 1. Network topology that is used to capture the DoH traffic.

Destination Layer 2 Address (Physical)	Source Layer 2 Address (Physical)	Layer 2 Protocol	Source IP Address (Layer 3)	Destination IP Address (Layer 3)
IP Protocol Number	Source Port Number	Destination Port Number	Data	Checksum

FIGURE 2. Layer 2 information (data link layer). Reassembled bits from electrical signal into a frame field.

similar representation applies to CSE-CIC-IDS2018 benchmarked dataset.

It seems there are no clear-cut similarities or differences between popularly cited benchmarked datasets and PCAP features. Hence the comparison requires some expert judgments and field experiences. These benchmarked datasets are claimed to closely resemble the real-world network dataset, similar to ground-truth dataset. However, none of these include network layer information (L3) of source and destination IP except for CIRA-CIC-DoHBrw-2020. This is highlighted in Table 2.

CIRA-CIC-DoHBrw-2020 is a DNS over HTTPS dataset, which is relatively a new protocol introduced in 2018 by IETF and published as RFC8484 [6]. A simulated version of a network traffic over this protocol was generated in 2020. It is under the initiative of study in [19] and this dataset is mainly used for IDS studies. It aims to reinforce security and privacy issue of DNS request over HTTPS channel. Many trusted web browsers such as Firefox, Safari, Chrome and Edge have adopted DoH. DoH will combat DNS data manipulation, Man-in-the-Middle (M2M) attacks and eavesdropping. Further discussion on CIRA-CIC-DoHBrw-2020 is available in section B.

In conclusion, a few problems were identified from the previous related studies. **First problem** is that the majority of the works were emphasized on classifiers development. In contrast, less effort has been put into data preprocessing, specifically in the area of data visualization [4].

**Second problem** is the authors in [4] concluded that current IDS datasets suffer from realistic network traits. Most public datasets have deficiencies to reflect complex modern network threats [5], [20]. Furthermore, those datasets have redundant records, repetitive features and overlapped label [6], [7]. The unrealistic properties limit their ability to represent contemporary low footprint and stealthy cyber-attack [8]. Thus, it is important to have a realistic dataset.

**Third problem** is the practice of generating augmented dataset will not represent the underlying nature of cyber-attack. Usually, the practice is to increase or replicate the volume of minority class, like in [9]. Imbalanced traits have low threats footprint. This is actually the characteristic of a stealthy attack. Stealthy attack has low traffic frequency [15]. Hence a dataset that renders a very low threats footprint is highly recommended. This dataset reflects real-world cyber-attacks.

**Fourth problem** is most of the public datasets demonstrate the classical DDoS attacks and its variants. Many researches have analyzed these types of attacks. However, none of them have simulated against the recent DoH protocol that was published by IETF in 2018. There are more sophisticated exploits that have been introduced to compromise DNS servers in recent years. Due to that, IETF introduces DNS over HTTPS. However, this protocol is vulnerable to Malicious-DoH, a type of exploit that can be generated using off-the-shelf tools like dns2tcp, DNSCat2 and Iodine. Hence effort to study this attack type is crucial.

**Fifth problem** is none of these public datasets include network layer information or the OSI layer 3 information. Layer 3 information is the most significant protocol in interconnected networks [21]. It is responsible for packets routing [4] which is crucial for packets' forwarding. Since, the attack's transaction takes place over network, hence the presence of this protocol is mandatory. This layer-related makes most of the popular cited public datasets categorized differently from the standard PCAP features. Many meta-data and raw-data features of the popular dataset have different features from the real production features. PCAP features of the real production network are discussed in section C of this section. Worth to note here, CICIDS2017 dataset, for example, is claimed to be the closest related public dataset to production network [2]. However, when comparable to the PCAP features, it satisfies most of the related labels except the layer 3 information (source and destination IP).

Finally, the **sixth problem** is a few studies have shown contradictory analysis and classifying results. For instance, some studies showed RF achieved highest accuracy, as in [10], contrary to the report from [11] which reported that NB had achieved excellent performance. The irony is that both used similar CICIDS2017 standard dataset. There are studies

utilizing NSL-KDD to train on detection model of modern low-frequent attack. As mentioned in [8], NSL-KDD was not a depiction of low footprint attack. This suggests that this domain has realistic, non-reproducible and shareable results, as suggested by [4]. It also indicated that various classification models are event-specific and have to be handled case by case.

## B. CIRA-CIC-DoHBrw-2020 DATASET

Domain Name System (DNS) has several security loopholes and has been a great concern for cybersecurity researchers. More sophisticated exploits have been introduced to compromise DNS servers over the years. To countermeasure some issues related to DNS vulnerabilities, DNS over HTTPS was introduced by IETF in 2018. This is done by encrypting DNS queries and sending them over a covert tunnel. This DoH transaction has been replicated in CIRA-CIC-DoHBrw-2020. CIRA-CIC-DoHBrw-2020 is a synthetic dataset which aims to evaluate DoH traffic in a network environment.

This network topology implements two-layered approaches which are used to generate normal and attack DoH traffic along with non-DoH traffic. DoH traffic is generated by accessing top 10,000 Alexa websites. It is subdivided into non-DoH, benign-DoH and malicious-DoH. A non-DoH is a traffic generated through HTTPS protocol. Then a benign-DoH is a non-malicious DoH traffic that is also generated through HTTPS and it is accessed by clients that use Mozilla Firefox and Google Chrome web browsers. These two browsers support DoH protocol. Finally, the malicious-DoH is generated by using tools like dns2tcp, DNSCat2 and Iodine.

Fig. 1 shows the network diagram that is used to capture the DoH traffic. Firstly, for the first layer, traffic with normal web browsing activity that involves benign DoH is generated through the web browsers. This will generate non-DoH HTTPS and benign DoH traffic. This traffic was then captured by a few web servers. Secondly, for the second layer, malicious DoH was generated by a mixture of tools to be captured by malicious DNS server and DoH server. These generated traffics were then captured for pre-processing phase.

The web browsers utilized various public DoH resolvers. To utilize this resolver and various capturing tools, Firefox web browser was connected to GeckoDriver and Chrome web browser to ChromeDriver. These generated traffics were captured by tcpdump. A Python script that uses Scapy was developed to generate a DoH traffic flow generator and analyzer. A tool named DoH Data Collector was then mounted to simulate different sets of DoH tunneling incidents.

For DoH server infrastructure, it is implemented by using Adguard, Cloudflare, Google and Quad9 platform. For the non-DoH and benign DoH, the packets generated amounted to 48952 Kbytes packets. On the other hand, the malicious packets that were generated amounted to 219458 Kbytes packets of traffic. The transmission rate is set randomly between 100bps to 1100bps. The dataset document provides lists of IP addresses used to generate non-DoH, normal DoH

**TABLE 3.** IP addresses in relation with DoH traffics.

IP addresses	Destination IPs (all TLS packets are DoH packets)	Source IPs (Connect to Google Chrome client)	Source IPs (Connect to Mozilla Firefox)	Source IPs (used to create DoH tunnels)
1.1.1.1	✓			
8.8.4.4	✓			
8.8.8.8	✓			
9.9.9.9	✓			
9.9.9.10	✓			
9.9.9.11	✓			
176.103.130.131	✓			
176.103.130.130	✓			
149.112.112.10	✓			
149.112.112.112	✓			
104.16.248.249	✓			
104.16.249.249	✓			
192.168.20.191		✓		
192.168.20.111			✓	
192.168.20.112			✓	
192.168.20.113			✓	
192.168.20.144				✓
192.168.20.204				✓
192.168.20.205				✓
192.168.20.206				✓
192.168.20.207				✓
192.168.20.208				✓
192.168.20.209				✓
192.168.20.210				✓
192.168.20.211				✓
192.168.20.212				✓

and malicious DoH traffic. This data are generated by running the simulation simultaneously over the entire servers. The generated traffic captured all destination IPs that are used for browsing public DoH servers. It means that all the TLS traffic to these servers is DoH packets. It also captured source IP of the clients that had used various web browsers to access those websites. As designed, only Google Chrome and Mozilla Firefox were used to depict client’s web browsers. Finally, the source IPs that utilized DoH tunnels were also captured and recorded. Table 3 visualizes the lists of IP addresses and its relation to DoH traffics.

The generated packets were extracted as flow-based or meta-data features. In this study, meta-data features or raw-data features are treated equally. The study doesn’t discriminate between processes flow information label and raw label.

**C. LAYER 2 INFORMATION**

Generic PCAP features as depicted in Table 2 show column time; the time for which the frames were captured. Time here, however, measures delta time up to microseconds from sequence of a completed handshake network transactions. Then there are columns source and destination IP address. These are valuable network layer information (L3). It shows the communication between packet originator and the intended recipient.

Next column is protocol which is a set of rules that are used in network communication. The column frame length is the size of communication wire in bytes of a particular transaction and finally is the info column which is not included in Table 2. This column is to provide more descriptions about a particular packet in text form. Usually, it is difficult to process this column in a classification machine learning (ML) training program, hence it is safe to drop this column. Obviously, in PCAP’s features there are no attack and normal labels which require unsupervised type of ML trainings. Here, extra features like source port, destination port (L2 information) and a few more from the frame field information can be added into the column. It is added as additional filters and sometimes through careful examination and deep packet inspection. There are obviously more vital OSI layer components that need to be added.

These vital OSI layer components reside in the data layer link layer, which encapsulates most of the information from the upper layers and provides function to transfer Protocol Data Unit (PDU) between nodes. It serves a request from network layer and directs it to the physical layer. During this transmission, data can be successfully received and acknowledged. However, sometimes that transfer can become unreliable. Hence, in those cases, upper layer protocols like data link layer will perform error checking, acknowledgments and retransmission. It includes application layer protocol information, transportation layer protocol number (either TCP or UDP) information, source and destination IP or simply layer 3 information, source and destination Media Access Control (MAC) address information, source and destination port number and finally checksum.

In IEEE 802 Local Area Networks (LAN) standard, this data link layer is defined in great detail. Logical Link Control (LLC) and Media Access Control (MAC) are amongst the sub-layers sitting in the data link layer. MAC layer determines who is allowed to enter the medium of communication. Carrier Sense Multiple Access (CSMA) Collision Detection or Avoidance (CD/A) are the protocols to control this access. This MAC sub-layer is also important for frame synchronization and bit stuffing. On the other hand, LLC sub-layer is important for error control and flow control.

Obviously, PCAP’s features have no attack and normal label as represented in many synthetic datasets. This label will help classifier model to learn to ultimately reduce the loss function. Hence, to train classifier model against PCAP file or from ground-truth dataset requires unsupervised learning. A feature or a combination of coefficient values are needed



as the output label. This feature requirement is perfectly matched to the data link layer information, as depicted in Fig. 2. Many of these features can be extracted and elected as the output label.

From various public datasets or synthetic datasets, CICIDS2017 is claimed to be the closest related public dataset to production network [2]. However, when comparable to the PCAP features, it satisfies most of the input labels except source and destination IP. As stated, source and destination IP, or L3 information is crucial information in network communication. In a nutshell, none of these public datasets include network layer information (OSI layer 3) except for CIRA-CIC-DoHBrw-2020, which is a relatively new dataset that simulates attack on a new protocol DNS over HTTPS (DoH). This dataset was introduced in 2018 by IETF and published as RFC8484.

### III. RESEARCH METHODOLOGY

This section introduces the research methodology to complete this study. Altogether, there are three essential methods. Firstly, the dataset will be processed through EC, a centrality density method that is applied in Graph and Network theory. Secondly, through the PCA and finally through the GMM analysis, a prominent feature for unsupervised learning is unleashed. GMM is a variant of  $k$ -NN and mostly used in machine learning algorithms [22].

#### A. GRAPH MODEL

Network is a collection of interacting elements, for instance the World Wide Web (WWW), social networks, brain networks, and, in our use case, a connected DNS over HTTPS networks. To understand these DoH network as to perform clustering and classification tasks, graph theory is used to model their relationship. A graph,  $G$  is represented by vertices,  $v$  (or nodes or in this case the IP address) and its edges,  $e$  (or links or communication between IPs) and is denoted as in (1).

$$G = (v, e) \quad (1)$$

$v$  and  $e$  indicate the number of vertices and edges in the represented graph. Graphs can be either; 1) undirected or bidirectional between two nodes or 2) directed, which implies only one path from one node to another. In this case, the graph is undirected, as shown in Fig. 13(a) in Section IV; Results. It illustrates a simple three nodes network in different subnets that are able to reach each other in bidirectional or in full duplex communication. Creating the graph with list of source IP and destination IP is defined in (2).

$$\begin{aligned} v &= \{\text{Allipaddresses}\} \\ e &= \{\text{sourceip, destinationip}\} \end{aligned} \quad (2)$$

This graph is then transformed into adjacency matrix which shows the relationship between nodes and how many edges are set between them. Since this is an undirected graph where all the edges go bidirectional, the adjacency matrix is symmetrical. To define adjacency matrix, it starts with a set of

vertex  $v = v_1, \dots, v_n$  where the matrix is a square of  $n * n$  of matrix  $A$  of element  $i, w$ . This  $A_{i,w}$  must be the element of an edge from  $v_i$  to  $v_w$ . It will be denoted as 0 if there is no edge. Eventually, all the diagonal elements of this matrix will be zero since a vertex is connected to itself (a loop).

The next step is to calculate the degree of the graph,  $d$ . This is calculated by looking at the number of edges that are connected to a particular vertex. It is denoted by (3).

$$d = 2e/v(v - 1) \quad (3)$$

where  $v$  is the number of vertices or nodes (IP addresses) and  $e$  is the number of edges (links between source and destination IP). Then, the next step is to calculate degree of centrality. Degree of centrality for a node  $v$ , is the fraction of nodes it is connected to. They are normalized,  $s$ , by dividing to the maximum number of possible degrees in a graph  $n-1$ , as shown in (4).

$$s = 1/(n - 1) \quad (4)$$

where  $n$  is the number of nodes,  $v$  in the graph  $G$ . Hence degree of centrality is calculated by (5).

$$\text{Centrality, } C = d * s \quad (5)$$

Next Eigenvector centrality (EC) is computed. EC computes the centrality of a node according to the **centrality** of its neighbors. It is also to measure the influence of a node in a network. For the given graph  $G = (v, e)$ , where  $|v|$  are the vertices and  $e$  are the edges, let adjacency matrix be as  $A = (a_{v,w})$  where  $v$  and  $w$  are two different vertices. When  $a_{v,w} = 1$ ,  $v$  and  $w$  are connected to each other, and when  $a_{v,w} = 0$ , these are disconnected to each other. Given relative centrality of node or vertex  $v$  as  $x_v$  it is denoted as in (6).

$$x_v = 1/\lambda * \sum_{w \in M(v)} x_w \quad (6)$$

where  $M(v)$  is all the neighbors of node  $v$  and  $\lambda$  is a constant and  $x_w$  is the sum of relative centrality between node  $v$  and  $w$ , which is denoted as  $x_w = 1/\lambda \sum_{w \in V} a_{v,w} x_w$ . This can be simplified into vector notation of Eigenvector as denoted in (7)

$$Ax = \lambda x \quad (7)$$

The next step is to calculate the Shortest Path or Betweenness Centrality (BC). Shortest Path or BC of a node  $v$  is computed by summing up all the fractions of all shortest path pairs that pass-through  $v$ . It is expressed in (8).

$$\text{Betweenness, } C\_B(v) = \sum_{s, w \in V} (\sigma(s, w|v) / \sigma(s, w)) \quad (8)$$

where  $v$ , is the set of vertices (nodes),  $\sigma(s, w)$  is the number of the shortest path between  $s$  and  $w$ ,  $\sigma(s, w|v)$  is the number of the shortest path between  $s$  and  $w$  given some other nodes, the set of vertices  $v$ . When  $s = w$ ,  $\sigma(s, w) = 1$ , and if  $v \in s, w$  then  $\sigma(s, w|v) = 0$ . The latter condition is understood since there is no unique shortest path in a given new node.

## B. PRINCIPAL COMPONENTS

This method is used to find **major patterns** in this dataset. This typical pattern is called Principle Component (PC). It is used when data points contain a lot of measurement and not all of those are meaningful, or defined as a lot of covariance in the measurements. Variance helps to understand how far the random variable is spread out from its mean. First step is to calculate each column average  $\bar{x} = 1/n \sum_{i=1}^n x_i$ . Then check how each frame deviates from that average,  $deviation_i = x_i - \bar{x}$  and subsequently compute the covariance between two locations, as given in (9).

$$\sigma(x, y) = 1/(n-1) \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \quad (9)$$

Sum of all deviations from all frames will form a covariance matrix which contains terms for all possible pairs of features. PCA can be computed from these covariance matrices. Eigenvectors with the largest Eigenvalues are the PCs. The equation as stated in (7) is applied here.  $Av = \lambda v$ , where  $A$  is transformed into covariance matrix,  $v$  is the Eigenvector and  $\lambda$  is the Eigenvalue. To process PCA, column SourceIP, DestinationIP and Timestamp, however, have to be dropped as calculation can only process real numbers. Then ResponseTimeTimeSkewFromMedian and ResponseTimeTimeMedian have to be imputed since they include missing values. These are done with the help of a Python library package called NetworkX.

PC shrinks all encoded vectors into a single line. PC will capture all the major axis of variation but doesn't lose much of the information. In this research, negative covariance is discovered, which will be reported in the Results and Discussion section.

## C. GAUSSIAN MIXTURE MODEL

This model is used to fit a vector of unknown prior parameter,  $\theta$  or the means  $\mu_i$  and covariance matrices  $\Sigma_i$ , as shown in (10). In this research, it is used in **clustering** model based on the underlying pattern from the dataset.

$$p(\theta) = \sum_{i=1}^K \phi_i N(\mu_i, \Sigma_i) \quad (10)$$

where  $i$ th vector component is characterized by normal distributions with weights  $\phi_i$ , means  $\mu_i$  and covariance matrices  $\Sigma_i$ . To integrate this prior into a Bayesian approximation, the prior is multiplied with the known distribution of  $p(x)$  given the unknown parameter  $\theta$ . This  $p(x|\theta)$  is also known as posterior distribution and can be expressed as in (11).

$$p(\theta|x) = \sum_{i=1}^K \tilde{\phi}_i N(\tilde{\mu}_i, \tilde{\Sigma}_i) \quad (11)$$

With another new parameters of  $\tilde{\phi}_i$ ,  $\tilde{\mu}_i$  and  $\tilde{\Sigma}_i$ , another algorithm is needed to update them. This is usually done by Expectation Maximization (EM) algorithm, an iterative method to find maximum likelihood between parameters. For example, given a set of  $X$  observed data, a set of missing values  $Z$  and a vector of unknown parameters, the likelihood function is  $L(\theta; X, Z) = p(X, Z|\theta)$ . The Maximum likelihood is determined by maximizing the marginal likelihood.

It can be done iteratively to find Expectation step ( $E$  step) and Maximization step ( $M$  step).  $E$  step  $Q(\theta|\theta^{(t)})$  is computed by (11).

$$Q(\theta|\theta^{(t)}) = E_{z|x, \theta^{(t)}} [\log L(\theta; X, Z)] \quad (12)$$

where  $E$  is the expected value,  $z|x, \theta^t$  is the distribution of  $Z$  given  $X$  and the current estimation of parameters  $\theta^{(t)}$ ,  $\log L(\theta; X, Z)$  is a log likelihood function of parameter  $\theta$  with respect of all that. To maximize the step, the  $M$  step is denoted by (13)

$$\theta^{(t+1)} = \arg \max Q(\theta|\theta^{(t+1)}) \quad (13)$$

Which denotes to find the maximum parameters that finally satisfy this equation.

## IV. RESULTS

DNS over HTTPS is relatively a new protocol that was introduced in 2018. It aims to reinforce security and privacy issue of DNS requests over HTTPS channel. Many trusted web browsers such as Firefox, Safari, Chrome and Edge have adopted DoH. DoH combats DNS data manipulation, Man-in-the-Middle (M2M) attacks and eavesdropping.

Despite that, it also suffers other security breaches such as spoofing. Spoofing will lead to data exfiltration and C&C attacks through malware proliferation. DoH dataset of CIRA-CIC-DoHBrw-2020 establishes security flaws in DNS like DNS tunneling and DNSbased malware. This flaw can bypass firewalls. Hence detecting DoH threats is crucial. Dataset features here is defined as flow information or a processed meta-data. Table 4 below shows the output of data.info() from CIRA-CIC-DoHBrw-2020 dataset.

From Table 4, there are 35 columns (from 0 to 34) altogether. An entry index from 0 to 167516. It has one entry datatype (dtypes) of boolean, 26 entries of float64 datatype, five entries of int64 datatypes and, three objects datatype. Memory usage to process this 167k counts of dataset is about 44Mbytes.

The source and destination IP by far haven't been found in any benchmarked dataset except in the CIRA-CIC-DoHBrw-2020. Fig. 3 shows its description. Most of the features' mean value lay at the floor level except for PacketLength information. These are attributed for value ranges from minimum to 50%. It is also clearly seen a back wall that contains vertical values range from 70% of sizes to maximum. Those features are coming from FlowBytes and the PacketLengthVariance. Most of these back wall features are coming from the raw features an, in contrast, most of the features below the floor level are the processed features or meta data.

From Fig. 4 of skewness and outliers distribution graphs, there are many insightful informations revealed. For instance, Fig. 4(a) shows most of the traffic was attributed to the DoH attack's label. Almost 99.9% of the traffic or 167,486 frames are labelled malicious and only 0.01% or 31 frames are labelled normal. It is indeed an imbalanced dataset of CIRA-CIC-DoHBrw-2020 whereby most of its traffic are the attack's traffic and only 31 of them are considered benign. In a

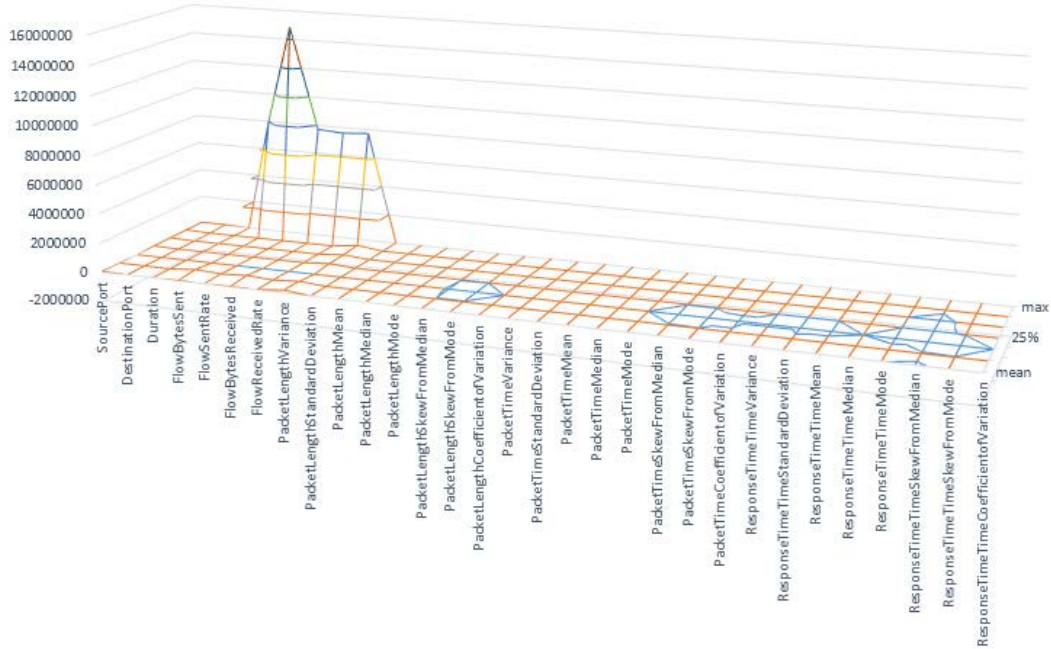


FIGURE 3. CIC-DoHBrw-2020 dataset descriptions.

low footprint attack, this is considered a near-realistic public dataset. In retrospective, most of the features were rightly skewed as opposed to the DoH label. Duration, for instance, as shown in Fig. 4(b) has a bimodal shape that shows the distribution of time from 0 seconds to under 20 seconds and from 30 seconds to under 40 seconds. These are the typical times attributed to attack traffic. On the other hand, a longer duration or the outliers (rightly skewed) are from 80 seconds to below 140 seconds and these are considered normal traffic duration.

Similarly, features like FlowBytesSent, FlowSentRate, FlowBytesReceived, FlowReceivedRate, PacketLengthVariance, PacketLengthStandardDeviation, PacketTimeMean, ResponseTimeTimeMean, PacketLengthMean and ResponseTimeTimeMedia as represented by Fig. 4(c) and (d) have a similar rightly skewed distribution. PacketLengthMean, as shown in Fig. 4(d) for instance, has the packet length around 0 to 400 bytes. This is mainly the packet size of an attack traffic. The outliers' packet length size of 500 byte to 2500 bytes, on the other hand, indicates the normal traffic. FlowBytesSent and FlowBytesReceived have majority bytes of size below  $1 \times 10^6$ . This is also the majority bytes of attack traffic. Anything above this size to the maximum of  $7$  to  $8 \times 10^6$  bytes is the minority normal traffic. Based on these observations, most of the normal traffic has the least outliers.

Fig. 5 shows the evolution of DoH label over time. These were collected every second in 2020. For efficiency, the data collected and displayed here are between March 2020 until April 2020. It is clearly seen here that the entire duration was

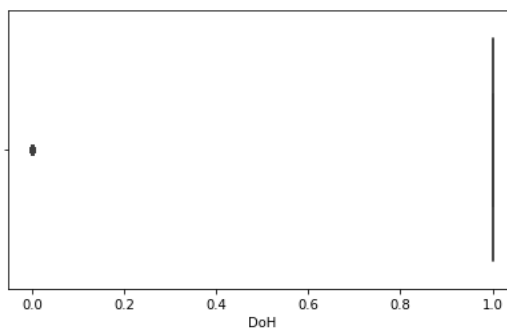
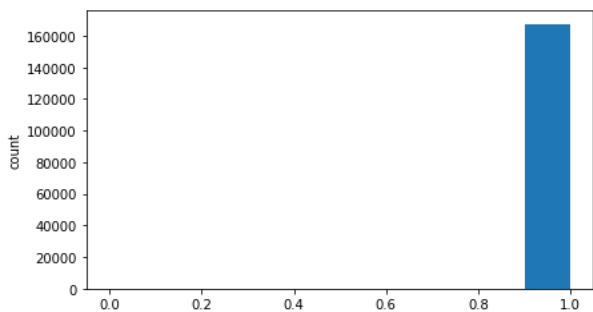
filled by attacks' attempt over DoH traffic. There are some indications of a **normal** condition which is on 31 March, 2020 at time 06 hour and the similar pattern was spotted on 01 April 2020 at time 06 hour, which also indicates a normal traffic. Time here is formatted as %H: %M: %S. These are the only two timestamps where the traffic get normal and were characterized before as having the minority outliers' traffic distribution.

To visualize the DoH traffic better, Fig. 6 shows the evolution of three flows, namely FlowSentRate, FlowBytesReceived and FlowReceivedRate. From the graph, it is seen, prior the normal condition both on 31 March, 2020 at 06 hours and 01 April, 2020 at 06 hours, these three flows show a spike in network traffic. These three labels could reflect the DoH normal traffic traits. These spikes might indicate the outliers' distribution of the flows. Low footprints of an attack traffic are clearly demonstrated in this graph.

Similarly, Fig. 7 shows the evolution of three packets types namely PacketLengthMedian, PacketLengthVariance and PacketLengthMean. On the 06th hours of both dates (31 March, 2020 and 01 April, 2020) those packet types show increases in their sizes. They reach up to  $1000 \times 10^6$  size in bytes (1000Mbytes). This is also another indication to show how a normal traffic behaves. Again, it is demonstrated here that a normal DoH traffic will have outliers' distribution, which is usually off the mean and reaches its maximum sizes.

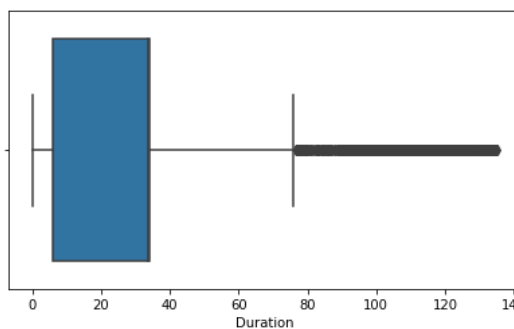
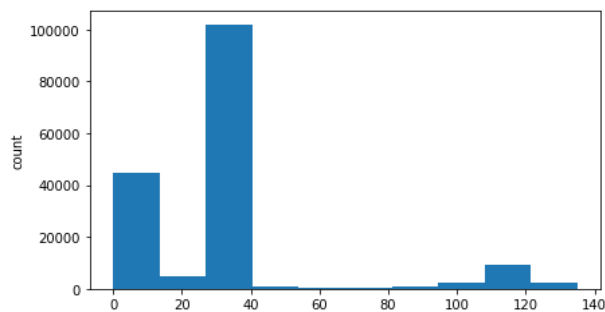
Fig. 8 shows the evolution of PacketTimes' label over time. PacketTime, however, shows a different characteristic. This PacketTime (Mean, Median, Mode) evolution doesn't indicate any significant difference of sizes as compared to

DoH  
Skew : -73.49



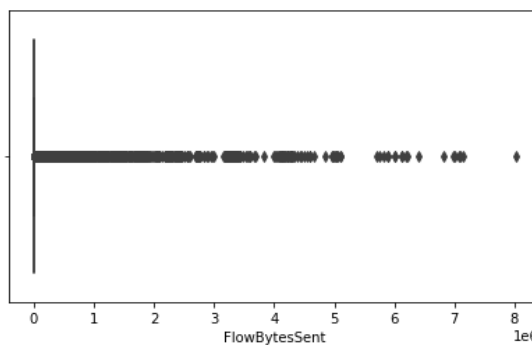
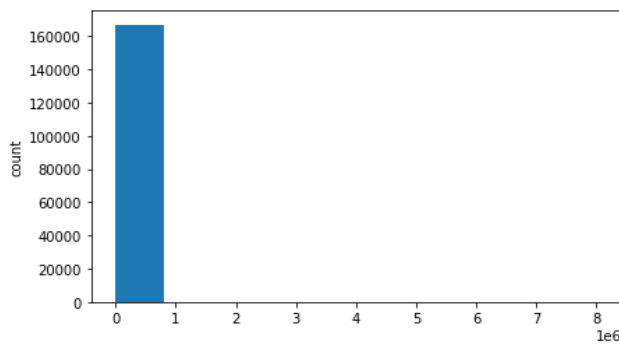
a) Attack label DoH contributed 99% from total traffic

Duration  
Skew : 1.75



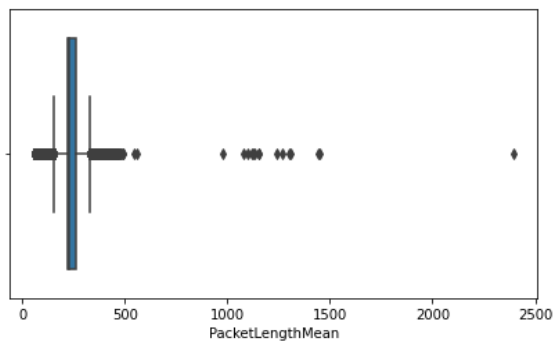
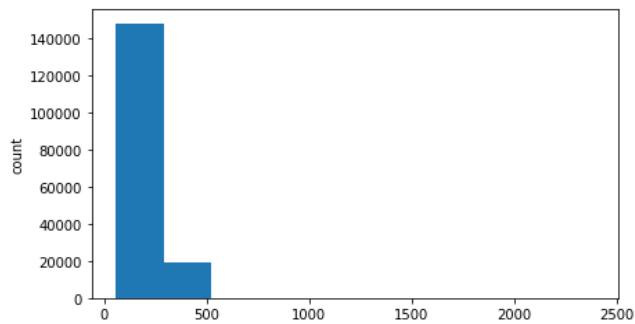
b) Bimodal distribution of Duration label

FlowBytesSent  
Skew : 24.82



c) Flow distribution

PacketLengthMean  
Skew : 0.43



d) Packet distribution

FIGURE 4. CIRA-CIC-DoHBrw-2020 skewness and outliers distribution.



TABLE 4. Dataset Info.

Column	Dtype (datatype)	Count	
SourceIP DestinationIP TimeStamp	Object	167517	
SourcePort DestinationPort FlowBytesSent FlowBytesReceived	Int64		
PacketLengthMode Duration FlowSentRate FlowReceivedRate PacketLengthVariance PacketLengthStandardDeviation PacketLengthMean PacketLengthMedian PacketLengthSkewFromMedian PacketLengthSkewFromMode PacketLengthCoefficientofVariation PacketTimeVariance PacketTimeStandardDeviation PacketTimeMean PacketTimeMedian PacketTimeMode PacketTimeSkewFromMedian PacketTimeSkewFromMode PacketTimeCoefficientofVariation ResponseTimeTimeVariance ResponseTimeTimeStandardDeviation ResponseTimeTimeMean ResponseTimeTimeMode ResponseTimeTimeSkewFromMode ResponseTimeTimeCoefficientofVariation	Float64		
ResponseTimeTimeMedian ResponseTimeTimeSkewFromMedian			167318
DoH (Note: Attack label)	Bool		167517

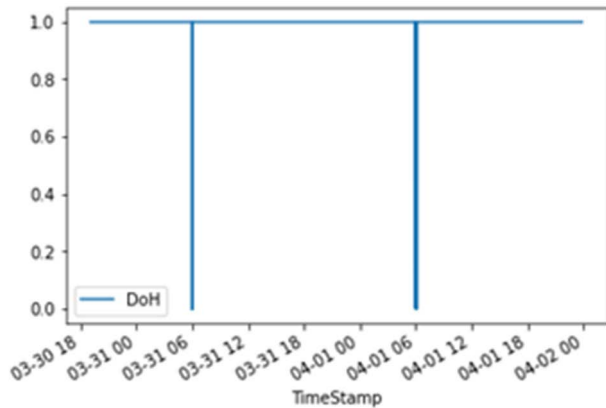


FIGURE 5. DoH Timestamp.

the Flow and PacketLength features on both dates. This is easy to comprehend since transmitting a packet of size 1 kbytes will have similar sending time effect as transmitting a packet of size 1Mbytes given a locally connected network. However, sending time changes dramatically when a packet is transmitted over a WAN network or IPSEC tunnel for instance.

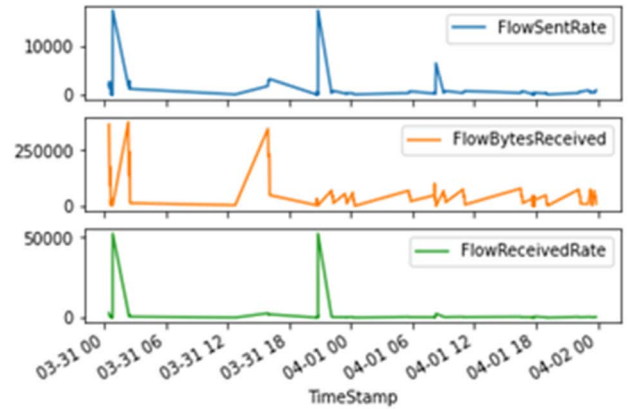


FIGURE 6. Flow timestamp.

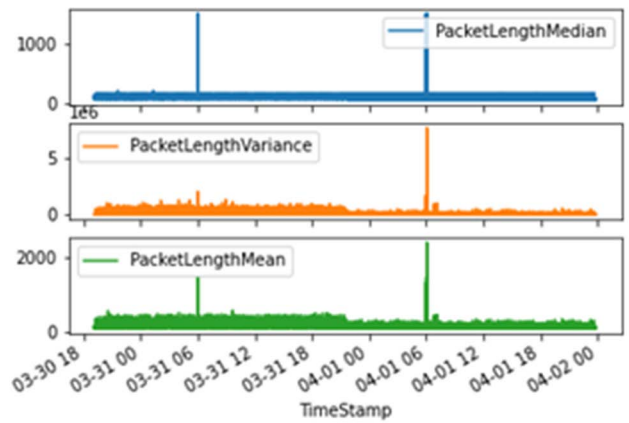


FIGURE 7. PacketLength timestamp.

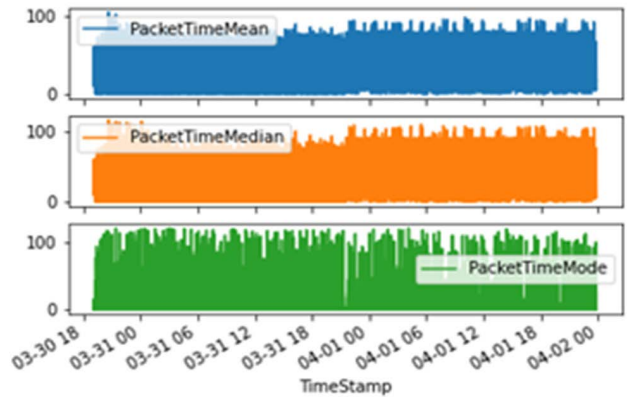


FIGURE 8. PacketTime timestamp.

A. DOH GRAPH MODEL

Graph and network model are used to understand this DoH network as well as understand their IP relationships. This will better assist on visualizing the dataset subsequently to perform clustering and classification tasks. Fig. 9 shows the SourceIP relationships with the DoH label. Almost all of these IPs have been compromised by DoH traffic. However, there are a few IPs which have not been listed as the source

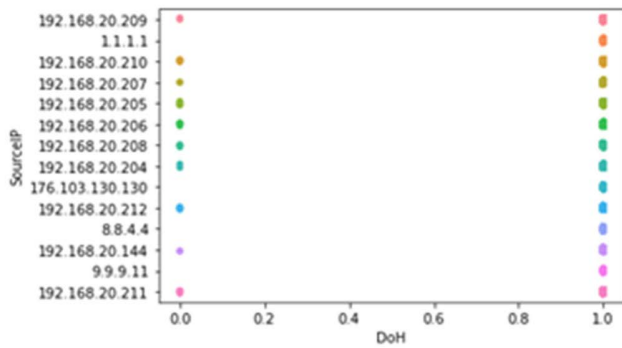


FIGURE 9. SourceIP distribution vs DoH (label).

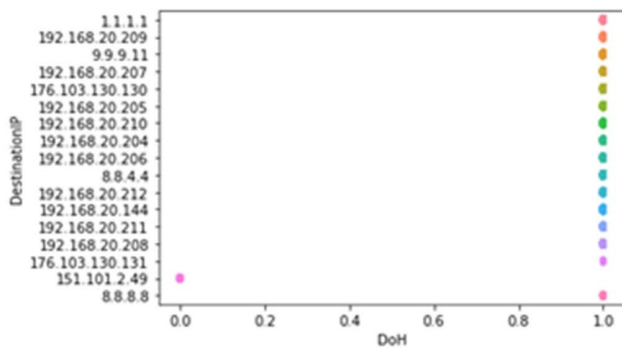


FIGURE 10. DestinationIP distribution vs DoH (label).

of benign DoH traffic, which are 1.1.1.1, 176.103.130.130, 8.8.4.4 and 9.9.9.11.

Lest we forget, from the document, those IPs were marked as destination IPs that are equipped with TLS packets over DoH traffic. In contrast, the IPs range from **192.168.20.144 to 212**, these are marked as the source IPs that had generated the DoH tunnels (traffic originator). That is reasonably why most of these IPs were plotted as DoH attack’s label.

Meantime, Fig. 10 shows the DestinationIP relationships with the DoH label. This graph has similar pattern from the previous graph in Fig. 9. Again, most of the IPs were destined to DoH’s attack terminal except from IP **151.101.2.49**.

1) BAR PLOT INFORMATION

Apparently, this IP wasn’t registered in any part of the dataset’s official document either from the source IPs connected to Google Chrome or Mozilla or from the destination IPs. This is very interesting because, on the later analysis using the centrality graph, on many occasions throughout these analyses, this IP was sorted as one of the most important nodes.

Fig. 11 shows the count of Source IP addresses against the DoH label to further support the given graph in Fig. 9 Previously, Fig. 9 shows most of these IPs were the source of DoH’s attack traffic. From Fig. 11, it is known that the most attack traffic was generated from IP **192.168.20.144**. The very least attacks generator is from the host IP

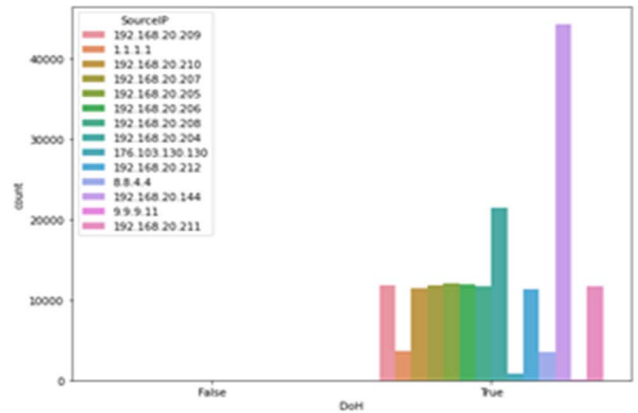


FIGURE 11. SourceIP count vs DoH (label).

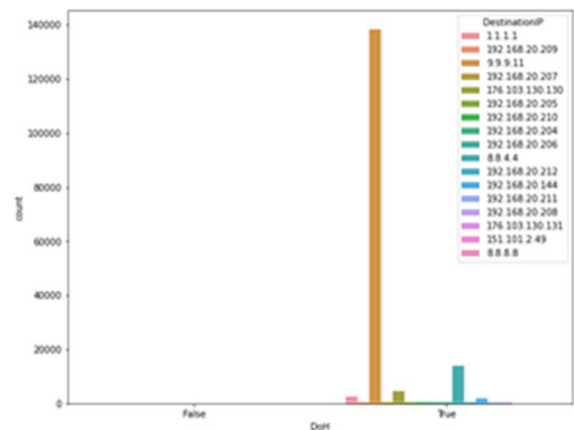


FIGURE 12. DestinationIP count vs DoH (label).

**176.103.130.130** and followed by IPs 1.1.1.1 and 8.8.4.4. In contrast, these are the IPs that have not been listed as the source of benign DoH traffic. Surprisingly, in similar condition, IP **9.9.9.11** is amongst the top attacks generator, which has similar counts as 192.168.20.x IPs range.

Fig. 12 shows the count of DestinationIP against the DoH label. The destination’s host compromised heavily by the DoH’s attack is **9.9.9.11**. This is the host also marked as attacks generator. This host has the characteristic of a Command and Control (C&C) server which can transmit and serve exploits traffic concurrently. A compromised host with C&C exploit is also known as Zombie.

2) NETWORK GRAPH MODEL (CENTRALITY INFORMATION)

Fig. 13 shows the generated graph model of CIRA-CIC-DoHBrw-2020 dataset. Graph *G*, which was introduced in Section III (a) is shown in Fig 13 (a). It shows all the nodes, *v* and its edges, *e*. To understand the centrality information of the graph *G*, degree of the node was being measured. Fig. 14 shows the degree information.

Nodes with IPs 1.1.1.1, 9.9.9.11, 176.103.130.130, 8.8.4.4, 151.101.2.49 and 8.8.8.8 have the highest degree. These

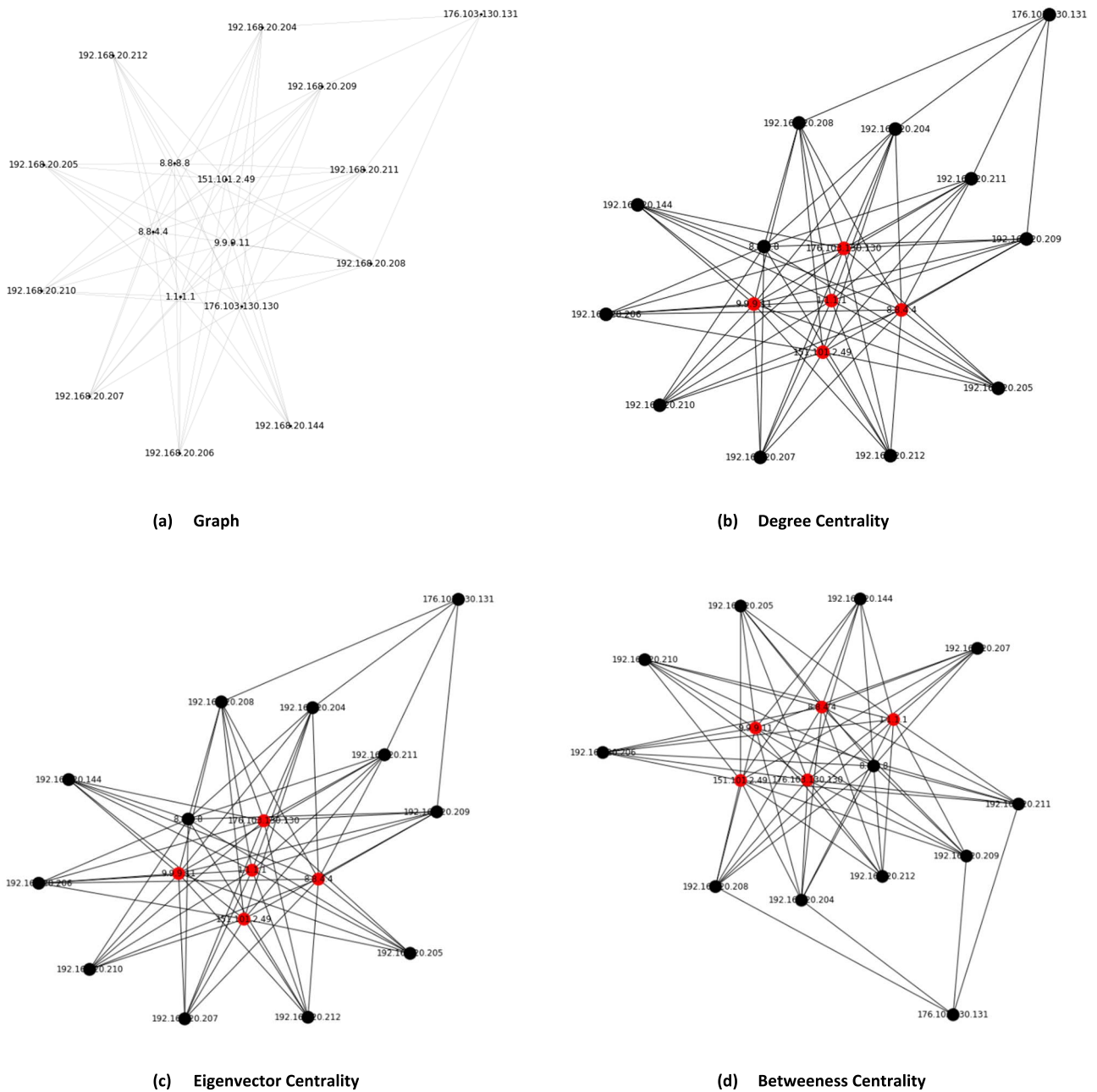


FIGURE 13. DoH graph model.

are the nodes that have been identified previously as the DoH attacks' generator and the most visited attack's DoH destination host. The newest recorded node is IP 8.8.8.8. This is known as Google Public DNS IP. From the dataset manual, it is stated that Public DNS is used as public DoH resolver.

This degree information will help to generate the centrality graph of Degree Centrality, Eigen Centrality and Betweenness Centrality, as shown in Fig. 13 (b), (c) and (d) accordingly. From the three cen-

tralities' graphs, the following nodes [ '1.1.1.1', '9.9.9.11', '176.103.130.130', '8.8.4.4', '151.101.2.49' ] define the most importance features, i.e. the most traffic travels in and out of these nodes. Again, these are all the IPs which have been described as an attacks generator and destination nodes.

**B. DOH PRINCIPAL COMPONENT (PC) MODEL**

Fig. 15 shows three different PC values which are generated from its original features. These PC explains the original

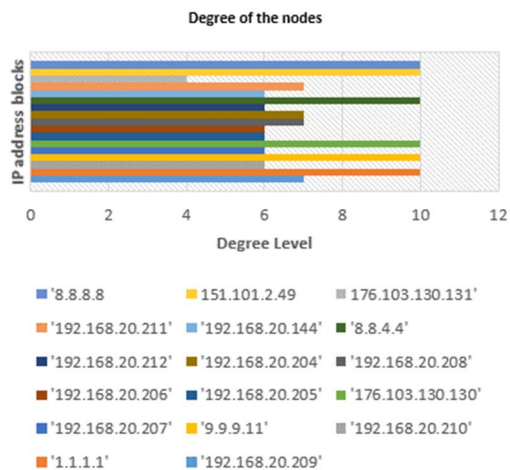


FIGURE 14. Degree of the nodes.

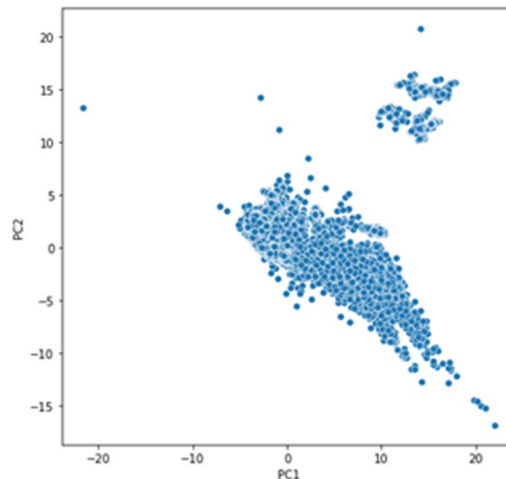


FIGURE 16. DoH dataset visualization using two PCs values.

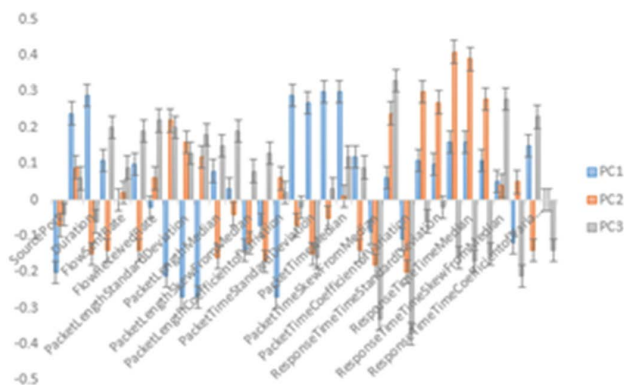


FIGURE 15. DoH principal components (1,2,3).

variance and look as a linear combination from original features. Some of the variance score higher than 0.2 and some scores below  $-0.2$ . These signify different covariance matrix values. This value is used to understand some underlying patterns of the data. Since the equation can only process real numbers, Source IP and Destination IP are dropped in this function.

PC1 is normally associated with high scores of all features even though, in this case, a few PCs indicate negative values. PC2 has also generated a few important coefficients. PC1 and PC2 will be good candidates to visualize this dataset in 2-dimensional (2D) graph. PC3 on the other hand performs poorly. Many of the coefficients lay just above zero and many more lay below zero (negative coefficients), which shows the least significant features.

Fig. 16 shows the 2D PC's graph for DoH dataset. It has two observable clusters. Do these two groups explain any of these underlying patterns of normal or attack's traffic? Fig. 17 unearths a few characteristics of this graph by associating the scatter plot with some hues information, such as SourcePort, DestinationPort, PacketLength and DoH. These features' labels were selected based on the previous flow and graph analysis.

Furthermore, Fig. 17c) depicts the PCs value with hue information on DoH label. Legend 0 indicates benign or normal and 1 indicates attack or malicious. It is noticeable that both clusters are mostly populated by attack traffic. Normal traffic fills a tiny spot from the bottom part of the big cluster. From Fig. 17d) on the other hand depicts the PC's value with hue information of the PacketLengthMean label. This label was chosen based on the timestamp characteristic in Fig. 8; PacketLengthMean timestamp. It has similar trait to the normal traffic of the DoH dataset.

In Fig. 17d), PacketLengthMean of size **1000 to 2000** bytes have filled up the exact same spot of the normal traffic characterized in Fig. 17c). Hence, larger packet length size seems to fit a normal traffic. This is a vital information as it assists to design an unsupervised IDS classifier model.

Then, Fig. 17a) and b) show the PC's graph for DoH dataset with hue information from SourcePort and DestinationPort. Apparently those two figures do not demonstrate similar traits, as shown in Fig 17d). However, it has revealed a few important attributes. For instance, the small cluster is entirely populated by the DestinationPort which has been labelled as malicious destination. Hence, the destination mostly has been compromised by malware.

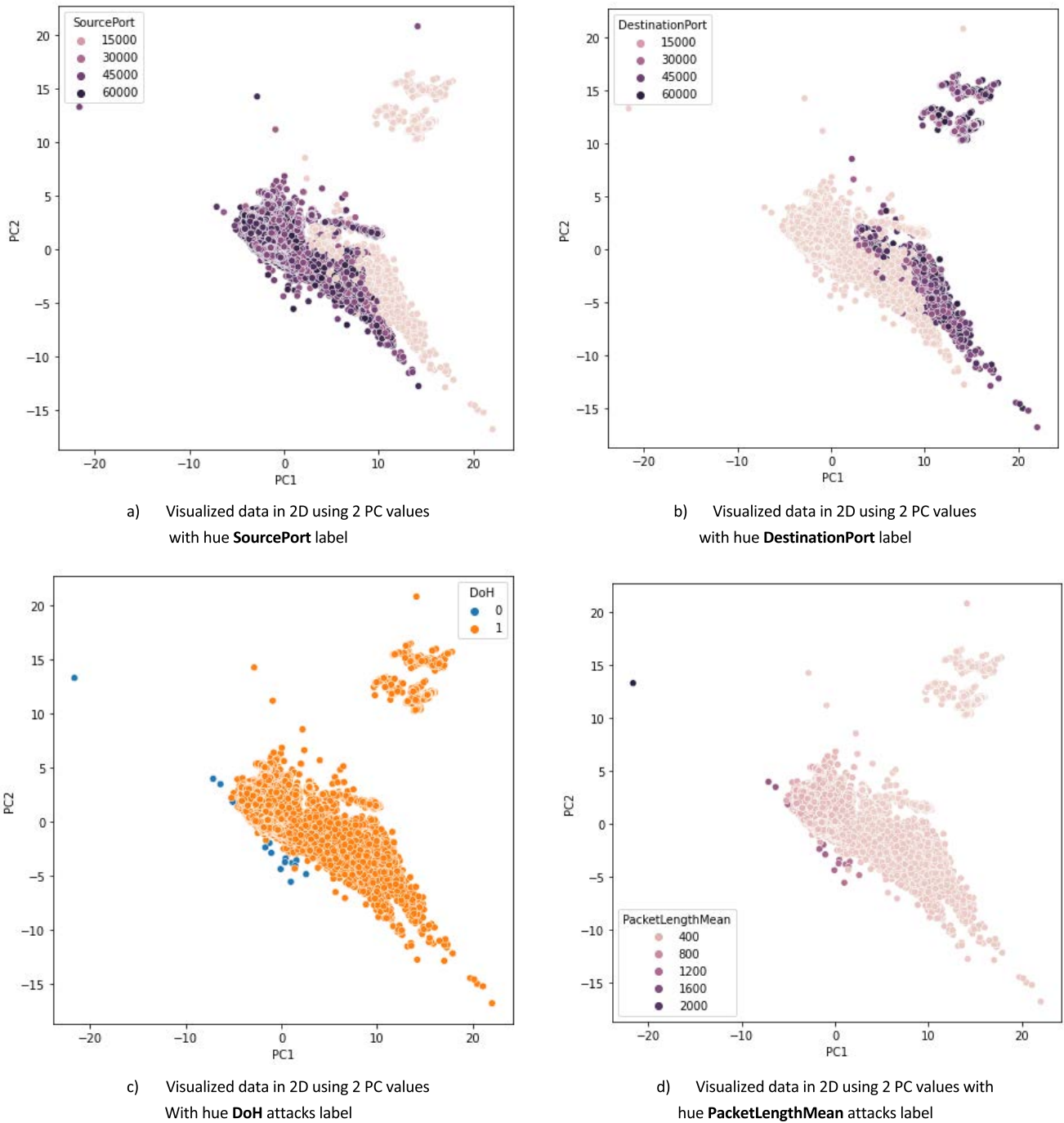
Meanwhile in the west region of the biggest cluster, it is entirely populated by the SourcePort, as shown in Fig. 17a), which is also the source for benign hosts. In this region, the majority of the hosts have been infected by malware.

**C. DOH GMM MODEL**

This model, on the other hand, has unearthed three clusters as depicted by graphs in Fig. 18. These 2D graphs show dataset features with GMM values against DoH attack and benign label. Cluster 0 has 94877 plots' count. Cluster 2 is the second highest with 39915 count and finally cluster 1 with 32725 counts. Total counts from these three clusters will sum up to 167,517 which is the total number of entries.

Since this is not a spatial dataset, the plot looks very straightforward with one-dimensional (1D) outlook. Cluster



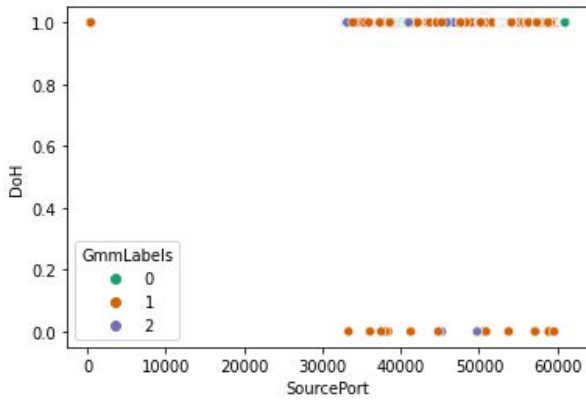


**FIGURE 17.** DoH dataset visualization using two PCs values with SourcePort, DestinationPort, PacketLengthMean and DoH hue information.

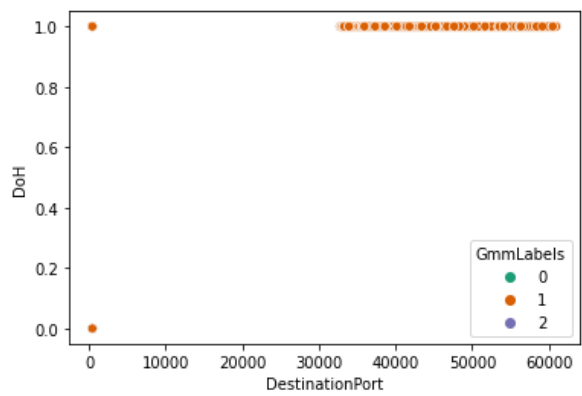
0 and cluster 1 have noticeable plots, whilst cluster 2 in some regions has the least plots. However, it is still recognizable. Fig. 18d) shows PacketLengthMean clustered in 1, 2 and 3. Cluster 1 and a tiny spot of cluster 0 have PacketLengthMean less than  $\pm 500$  bytes. They populate at the attack traffic or DoH label 1. A few spots in the range of 1000 to 2500 bytes are grouped in cluster 2. They populate at the benign traffic or DoH label 0. This is coherent to the finding in Fig. 17d) of the PCs 2D graph. In that graph PacketLengthMean of size

**1000 to 2000** filled up the exact same spot of the normal or benign traffic.

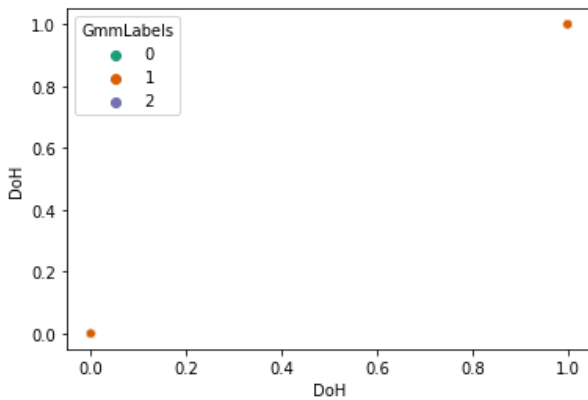
Fig. 19 shows the boxplot graph for SourcePort which is also clustered into three classes. Most of these clusters have SourcePort mean ranges from port 40000 to port 50000. In Fig. 18a) cluster 1 populates both attack and benign traffic. This is also coherent to the finding shown in Fig. 17a) where some of the SourcePort are safe and source from a benign traffic. On the other hand, majority of the cluster 1 ports



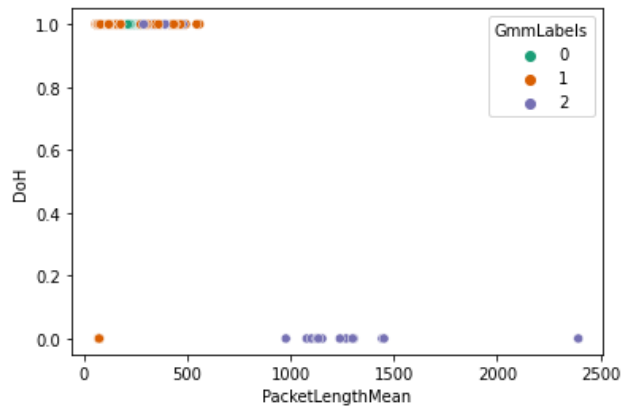
a) Visualized data in 2D using 2 GMM values with hue **SourcePort** label



b) Visualized data in 2D using 2 GMM values with hue **DestinationPort** label

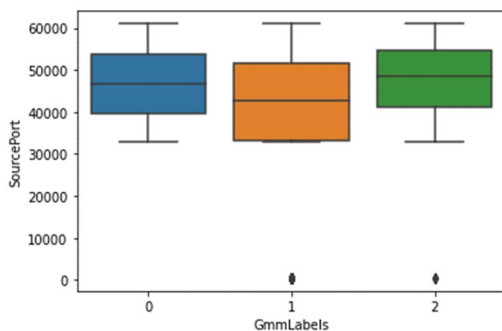


c) Visualized data in 2D using 2 GMM values With hue **DoH attacks** label



d) Visualized data in 2D using 2 GMM values with hue **PacketLengthMean** attacks label

**FIGURE 18. GMM model for DoH.**



**FIGURE 19. GMM labels for SourcePort.**

populate attack DoH traffic, as can be seen in Fig. 17b). Only one port, based on GMM model, from this range is considered benign.

**V. CONCLUSION**

The advancement in security mechanism revolves around protection and detection system. Intrusion Detection System security is still an important technology in the network and identity perimeter. It is used to detect classical and zero

day attacks in corporate network. It also provides just in time reporting during investigation and response process. However, the available public IDS dataset is impractical to reflect real cyber threats, to render real-time network scenario, to reflect recent malware attack, disregard layer 3 information and publish contradictory results. This problem can be resolved by sophisticatedly visualizing a new realistic, real-time, low footprint and up-to-date benchmarked dataset. Visualization helps to detect data deformation before designing the optimized and highly accurate classifier model. This study aims to review a new realistic benchmarked IDS dataset and apply sophisticated technique to visualize them. The study then applies Eigen Centrality (EC) technique from the graph theory to visualize this layer 3 (L3) information. Finally, it uses various visualization techniques such as Principal Component Analysis (PCA) and Gaussian Mixture Model (GMM). Results show the centrality graph clearly visualizes IPs that are compromised by recent attacks in real-time and the study concludes decisively that smaller packet length of size 1000 to 2000 bytes is to fit an attack trait.

## ACKNOWLEDGMENT

The authors declare that there are no conflicts of interest to report regarding the present study.

## REFERENCES

- [1] M. H. M. Yusof, A. M. Zin, and N. S. M. Satar, "Behavioral intrusion prediction model on Bayesian network over healthcare infrastructure," *Comput., Mater. Continua*, vol. 72, no. 2, pp. 2445–2466, 2022.
- [2] J. Yang, X. Chen, S. Chen, X. Jiang, and X. Tan, "Conditional variational auto-encoder and extreme value theory aided two-stage learning approach for intelligent fine-grained known/unknown intrusion detection," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3538–3553, 2021.
- [3] R. Heartfield, G. Loukas, A. Bezemskij, and E. Panaousis, "Self-configurable cyber-physical intrusion detection for smart Homes using reinforcement learning," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 1720–1735, 2021.
- [4] H. Hindy, D. Brosset, E. Bayne, A. Seem, C. Tachtatzis, R. Atkinson, and X. Bellekens, "A taxonomy of network threats and the effect of current datasets on intrusion detection systems," *IEEE Access*, vol. 8, pp. 104650–104675, 2020.
- [5] M. Ozkan-Okay, R. Samet, Ö. Aslan, and D. A. Gupta, "A comprehensive systematic literature review on intrusion detection systems," 2021, *arXiv:2101.05067*.
- [6] Z. Zoghi and G. Serpen, "UNSW-NB15 computer security dataset: Analysis through visualization," 2021, *arXiv:2101.05067*.
- [7] A. Kim, M. Park, and D. H. Lee, "AI-IDS: Application of deep learning to real-time web intrusion detection," *IEEE Access*, vol. 8, pp. 70245–70261, 2020.
- [8] K. Jiang, W. Wang, A. Wang, and H. Wu, "Network intrusion detection combined hybrid sampling with deep hierarchical network," *IEEE Access*, vol. 8, pp. 32464–32476, 2020.
- [9] G. Karatas, O. Demir, and O. K. Sahingoz, "Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset," *IEEE Access*, vol. 8, pp. 32150–32162, 2020.
- [10] D. Kurniabudi, D. Stiawan, M. Y. B. Idris, A. M. Bamhdi, and R. Budiarto, "CICIDS-2017 dataset feature analysis with information gain for anomaly detection," *IEEE Access*, vol. 8, pp. 132911–132921, 2020.
- [11] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F. M. Foozy, "Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset," *IEEE Access*, vol. 9, pp. 22351–22370, 2021.
- [12] C. Liu, Z. Gu, and J. Wang, "A hybrid intrusion detection system based on scalable K-means+ random forest and deep learning," *IEEE Access*, vol. 9, pp. 75729–75740, 2021.
- [13] Z. Hu, L. Wang, L. Qi, Y. Li, and W. Yang, "A novel wireless network intrusion detection method based on adaptive synthetic sampling and an improved convolutional neural network," *IEEE Access*, vol. 8, pp. 195741–195751, 2020.
- [14] L. Liu, P. Wang, J. Lin, and L. Liu, "Intrusion detection of imbalanced network traffic based on machine learning and deep learning," *IEEE Access*, vol. 9, pp. 7550–7563, 2021.
- [15] Y. Yang, K. Zheng, B. Wu, Y. Yang, and X. Wang, "Network intrusion detection based on supervised adversarial variational auto-encoder with regularization," *IEEE Access*, vol. 8, pp. 42169–42184, 2020.
- [16] X. Yu, Z. Tian, J. Qiu, S. Su, and X. Yan, "An intrusion detection algorithm based on feature graph," *Comput., Mater. Continua*, vol. 61, no. 1, pp. 255–274, 2019.
- [17] O. Almomani, "A hybrid model using bio-inspired metaheuristic algorithms for network intrusion detection system," *Comput., Mater. Continua*, vol. 68, no. 1, pp. 409–429, 2021.
- [18] M. Mehmood, T. Javed, J. Nebhen, S. Abbas, R. Abid, G. R. Bojja, and M. Rizwan, "A hybrid approach for network intrusion detection," *Comput., Materials Continua*, vol. 70, no. 1, pp. 91–107, 2022.
- [19] M. MontazeriShatoori, L. Davidson, G. Kaur, and A. H. Lashkari, "Detection of DoH tunnels using time-series classification of encrypted traffic," in *Proc. IEEE Intl Conf Dependable, Autonomic Secure Comput., Intl Conf Pervasive Intell. Comput., Intl Conf Cloud Big Data Comput., Intl Conf Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCCom/CyberSciTech)*, Aug. 2020, pp. 63–70.
- [20] X. Hao, J. Zhou, X. Shen, and Y. Yang, "A novel intrusion detection algorithm based on long short term memory network," *J. Quantum Comput.*, vol. 2, no. 2, pp. 97–104, 2020.

- [21] M. Ebbers, *Introduction to the New Mainframe: IBM Z/VSSE Basics*. Armonk, NY, USA: IBM, 2016.
- [22] H. Patel, D. S. Rajput, O. P. Stan, and L. C. Miclea, "A new fuzzy adaptive algorithm to classify imbalanced data," *Comput., Mater. Continua*, vol. 70, no. 1, pp. 73–89, 2022.



**MOHAMMAD HAFIZ MOHD YUSOF** received the B.Tech. degree (Hons.) in information technology from the University of Technology PETRONAS (UTP) Malaysia, in 2004, the M.Sc. degree in computer networking from the University of Technology MARA (UiTM), Malaysia, in 2014, and the doctoral degree in computer science from the National University of Malaysia (UKM) specializing in network-level malware classification model in 2020. He is currently a Researcher with the Research Interest Group (RIG), Machine Learning and Interactive Visualization (MaLIV), UiTM. He is a CISCO Associate with ID:CSCO11024858 and completed his CCNP (BSCI), CCNP (BCRAN) and CCNP (CIT) and successfully received Juniper JNCIA-Junos Certification. He had spent almost 12 years at IT industry specifically in IT infrastructure. His research interests include machine learning in cybersecurity, deep learning in IDS, and network security.



**AKRAM A. ALMOHAMMEDI** received the B.Sc. degree in electronics engineering from Infrastructure University Kuala Lumpur, Malaysia, in 2012, the M.Sc. degree in electrical and electronics engineering majoring in computer and communication system from Universiti Kebangsaan Malaysia, Malaysia, and the Ph.D. degree in wireless communications and networks engineering from University Putra Malaysia (UPM), in 2019. He is currently an Assistant Professor with the University of Karabuk (UNIKA), Turkey, and a Senior Researcher (remote-based) with South Ural State University (SUSU), Russia. His research interests include the Internet of Vehicles, the Internet of Things, multi-channel MAC, steganography, and wireless sensor networks. He was a recipient of the 2018 Best Paper Award from IEEE Malaysia ComSoc/VTS.



**VLADIMIR SHEPELEV** received the Ph.D. degree from Chelyabinsk State Agroengineering University, Russia, in 2000. He is an Associate Professor with South Ural State University (SUSU). He has been a Technical Scientist with LLC NTK-Logistics Transport and Logistic Company, since 2007. He has published several articles, books, and patents. His research interests include the Internet of Things (IoT), real-time traffic management, and the Internet of Vehicles (IoV). He is currently working in a project entitled "Development of an Intelligent Digital Platform for the Management of Transportation Systems of Cities based on Artificial Intelligence."



**OSMAN AHMED** was born in Riyadh, Saudi Arabia. He received the Diploma degree in networking from Imam Mohammad Ibn Saud University, Riyadh, in 2015 and the bachelor's degree in network security from INTI International University, Nilai, Malaysia, in 2020. He is currently an Information Security Consultant and a Researcher with the Banking Industry focusing on Vulnerability Assessment and Penetration Testing (VAPT). He ensures proactive monitoring on vulnerabilities in multi-tiered network environment. He also helps in various bug bounty programs that enable organizations secure applications and achieve continuous security testing at larger scale. He has published a Research Paper *Architecture Based on Tor Network for Securing the Communication of Northbound Interface of SDN* in 2020.

...