Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 2450 – 2468
© TÜBİTAK
doi:10.3906/elk-2011-31

A new classification method for encrypted internet traffic using machine learning

Mesut UĞURLU[1,*], **İbrahim Alper DOĞRU**[2], **Recep Sinan ARSLAN**[3],
[1]Department of Information Security Engineering, Graduate School of Natural and Applied Sciences, Gazi University, Ankara, Turkey
[2]Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Turkey
[3]Department of Computer Engineering, Faculty of Engineering and Architecture, Kayseri University, Kayseri, Turkey

Received: 06.11.2020 • **Accepted/Published Online:** 19.04.2021 • **Final Version:** 23.09.2021

Abstract: The rate of internet usage in the world is over 62% and this rate is increasing day by day. With this increase, it becomes important to ensure the confidentiality of the information in the traffic flowing over the internet. Encryption algorithms and protocols are used for this purpose. This situation, which is beneficial for normal users, is also used by attackers to hide. Cyber attackers or hackers gain the ability to bypass security precautions such as IDS/IPS and antivirus systems with using encrypted traffic. Since payload analysis cannot be performed without deciphering the encrypted traffic, existing commercial security solutions fall short in this situation. In this study, it is aimed to classify the network traffic by analysing the outgoing and incoming data over the encrypted traffic using extreme gradient boosting (XGBoost), decision tree and random forest classification methods. Thus, without deciphering, it is possible to classify packets passing through encrypted traffic using some metadata like size and duration and to take precautions against attacks. ISCX VPN-NonVPN dataset was used to test the proposed model in this study. With the created framework, encrypted traffic was classified with a high success rate and 94.53% success was achieved by using the XGBoost classification method.

Key words: Internet traffic classification, traffic identification, machine learning, cyber security

1. Introduction

Although human life is getting easier with the increase in internet usage rate, cyber-attacks are also increasing tremendously. With the increasing importance of privacy, especially in internet use, users prefer encrypted channels. The rate of usage of Secure Hypertext Transfer Protocol (HTTPS) traffic on the internet is over 88% and it is gradually increasing [1]. In the absence of encryption, critical data such as personal, commercial or military information can be captured by cyber attackers listening to the line or network [2]. Despite its benefits, encryption reduces the traffic analysis capacity of cyber security experts and network experts on the network and ensures that cyber attackers are hidden [3]. By using encrypted traffic, cyber attackers can do cyber-attacks by hiding from security measures like antivirus, intrusion detection system, data loss prevention system and application firewall [4].

The most used protocols within the scope of traffic encryption are virtual private network (VPN) [5] and secure sockets layer (SSL) protocols. SSL protocol was developed by Netscape Company and is used to establish a secure communication channel between client and server [6]. Although both VPN and SSL protocol are used

*Correspondence: mesut.ugurlu@gazi.edu.tr

This work is licensed under a Creative Commons Attribution 4.0 International License.

to provide secure communication, there are differences between them. In SSL encryption, only the payload part of the package is encrypted, while in VPN, the encryption process is performed by taking the whole package into another package, which is called encapsulation. The encryption process with VPN performs encryption at the network layer, which is the third layer of the open systems interconnection (OSI) [7] model. SSL works between OSI layers 5 and 7. Whereas SSL is mostly used for encrypting http traffic, VPN is widely used to encrypt all traffic between two endpoints [8].

Encrypted packets cannot be deep packet analysis and the data passing through the traffic cannot be analysed [9]. In this way, attackers can do data leakage, encrypt the victim's data for ransom, communicate with the command and control center, download malware and other cyber attacks. As a precaution against these attacks, encrypted packets should be analysed by deciphering. In cases where the traffic bandwidth is high and the decryption-analysis-encryption time is long, the efficiency of this solution decreases. In addition, deciphering the encrypted traffic of the users also violates the user information privacy policy. Studies are carried out in the field of traffic classification by examining the dimensions of incoming packets on the traffic without decryption. Artificial intelligence algorithms are preferred and used in these studies. Unidentified network traffic can be prevented by defining protocol and application over encrypted traffic using flow-based methods. Especially since viruses and botnets communicate encrypted with the command and control center, these applications can be detected and kept under control with traffic classification.

Since the beginning of the 1990's, unencrypted packages have been classified using signatures and patterns. The content of unencrypted packets is required for deep packet analysis. These methods cannot be used in cases where the package content is encrypted. Flow-based classification methods are used for these situations. In flow and time based systems, features such as package length, acknowledge package number and package arrival and departure times are used instead of package content. Internet protocol (IP) address and port information are not used as a feature in encrypted traffic classifications. Since encrypted traffic can be established over different ports, which reduces the success rate in artificial intelligence-based approaches.

In this study, encrypted internet traffic is analysed and labelled with the help of its features extracted without decryption. In this way, it is aimed to strengthen security systems and to increase the security of information systems and users. Our major contributions are

- a general approach for encrypted network traffic classifier based on machine learning model as from network traffic flow
- using synthetic minority oversampling technique (SMOTE) techniques dealing with unbalanced data to have a good performance
- examining the effect on problem solving using different learning algorithms and comparing them

In the second part of this article, some current studies on this subject are included. The dataset used in the study is analysed in detail and explained in the third chapter. The classification methods used in the study and their details are presented in the fourth chapter. The success and performance metrics of the model created are shown in the fifth chapter. In the last section, a general evaluation of the research has been made and compared with other studies in this field. In addition, suggestions are given for future studies to classify internet traffic.

2. Related works

Classification of encrypted traffic is a critical issue today. From past to present, researchers have performed various studies on this subject, which are usually done on application or protocol basis. In this section, the most successful of these studies are selected and explained in detail. By this way, the review of the past literature and the contribution of the model to be proposed in this study will be shown comparatively.

Alshammari [10] et al. conducted a study for the classification of secure shell (SSH) and Skype traffic. In the scope of the study, binary classification application as No-SSH with SSH and No-Skype with Skype was applied. Five different machine learning algorithms were used in problem solving and it was observed that the C4.5 algorithm gave the best outcome. The C4.5 algorithm exhibited 83.7% detection rate and 1.5% false positive rate performance in the worst scenario, 97% detection rate and 0.8% false positive rate performance in the best scenario.

Di Mauro [11] et al. have made a classification study for web real-time communication (WebRTC) applications. The dataset used was created by the authors using Tshark, MySQL and Weka applications. Within the scope of the study, 4 different machine learning algorithms were used and binary classification was made. At the end of the study, it was stated that the Random Forest algorithm gave the most successful result. The random forest algorithm has an accuracy rate of 96.4% and an F1-measure rate of 96.4%.

In 2016, Draper-Gil [12] et al. have created a flow and time based dataset which consists of two different scenarios. The classification of traffic encrypted with VPN and traffic encrypted without VPN was performed in the study. There are fourteen class labels in dataset. C4.5 and k-nearest neighbours (kNN) algorithms are used within the scope of the study. The success rates for both the first scenario and the second scenario are given in detail. As a result of the study, it was determined that the C4.5 algorithm gives better outcomes than kNN. Using the C4.5 algorithm, the highest precision rate was stated as 90.6% in the first scenario and 80.9% in the second scenario.

Seddigh [13] et al. have made a study to classify encrypted packets on high speed networks. The dataset was collected from the campus network and although there are over two hundred features, feature selection was made due to the high traffic bandwidths. The dataset contains 6 classes. Six different machine learning algorithms were used and a framework called machine learning traffic analytics tool (MLTAT) was created. Hyperparameters selection was made by MLTAT and binary and multiclassifications were done. In the tests, the precision rate for all classes is above 88%, but it has been observed that the Web browsing class has significantly reduced the average success.

In 2018, Caicedo-Muñoz [14] et al. focused on a quality of service classification study of encrypted traffic on VPN and non-VPN traffic. ISCX VPN-NonVPN data set created by Draper-Gil [12] and five different machine learning algorithms were used in the study. In the second scenario, Bagging and Boosting give better results than other algorithms, but it has been observed that there is not much change in the first scenario. In the second part of Scenario 1, the highest accuracy rate for VPN traffic is 89.03% and for non-VPN traffic is 93.19%. In the second scenario, an accuracy rate of 84.81% was achieved.

Saqib [15] et al. classified Voice over Internet Protocol (VoIP) traffic with static analysis methods, using the features of the packet instead of the packet content. In the study, the detection of the voice traffic flowing over the network was made independently of the application, security protocols and encryption mechanisms. The most used applications such as Skype, Yahoo messenger and Gmail on the Internet were analysed. As a result of the analysis, the packet size distribution (PSD), packet variety and packet rate (PR) features of the packets were selected and used for classification purposes. It has been observed that voice traffic has low packet

size, low packet diversity and high PR value. Since the classification process in encrypted VoIP traffic is more difficult, additional features like the number of packages, total flow time, arithmetic mean, standard deviation, PR and max-min packet size have been used. With the created structure, it has 93.6% detection rate in offline traffic, 100% detection rate in VoIP traffic collected by authors and 95% detection rate in online traffic.

Zhang [16] et al. created a framework called stereo transform neural network (STNN) for data classification over transport layer security (TLS)/SSL encrypted traffic in 2019. STNN framework has been made up by using long short-term memory (LSTM) and convolutional neural network (CNN) algorithms together. The dataset was composed by the authors and collected over 17 applications. They succeeded an accuracy rate of over 99%. Chari [17] et al. made a classification process on encrypted traffic using J48 machine learning algorithm. ISCXTor2016 dataset was used as dataset and features based on packet lengths were selected. In the tests performed over 6 classes, 91% accuracy rate was acquired.

Pradhan [18] et al. presented a study to detect and classify encrypted SSH traffic over the network. Hybrid radial basis function network (RBFN) based on machine learning approach was employed in her studies. RBFN is a feed forward neural network using k-means clustering and gradient descent learning techniques. AMP, MAWI, DARPAA99 and NIMS4 datasets were also used in the study. In the tests, the results made with RBFN were compared with AdaBoost, decision tree, naive Bayes and random forest algorithms. It has been come out that RBFN provides high success with an accuracy rate of over 99% for all datasets.

In 2018, Yang [19] et al. used a deep learning-based approach to classify traffic encrypted with TLS/SSL using autoencoder neural network (ANN) and CNN algorithms. The parameters used for autoencoder and CNN are specified in detail. As the dataset for deep learning, 10 and 50 pages using TLS/SSL traffic among the most used 500 sites provided by Alexa [1] were selected and two data sets were created. Autoencoder and CNN algorithms have been compared with five different machine learning algorithms that give the most successful results. CNN yielded the most successful results by achieving 96.46% accuracy in 10 web page based dataset and 85%accuracy in 50 web page based dataset.

Al-Obaidy [20] et al. presented a machine learning-based approach to classify and detect Skype, Whatsapp, Facebook, Netflix and Youtube applications communicating over the encrypted channel. Support vector machine (SVM), multilayer perceptron (MLP), naive Bayes and C4.5 machine learning algorithms were preferred within the scope of the study. The dataset was created by the team and collected from end user computers via Wireshark. There are 14 features in the data set created. Firstly, the classification process was performed for four applications, Skype, Whatsapp, Facebook and Youtube, and it was determined that the C4.5 algorithm gave the best result with 88.29% accuracy. The Netflix application was added to these four applications and tests were carried out on five applications. Again, C4.5 algorithm was observed to be the most successful algorithm with an accuracy rate of 86.33

Khatouni [21] et al. performed a study to classify social media, voice and video applications using encrypted traffic. For this purpose, they created a dataset with a tool that automatically visits web pages and uses the application on Firefox and Chrome browsers. Thirteen different machine learning algorithms were used in the study. Although the random forest algorithm gives the best results, it has been determined that it consumes more resources than the decision tree algorithm. It has been stated that the decision tree algorithm works seven times faster than the random forest algorithm. As a result of the study, it was stated that the decision tree algorithm was the most powerful algorithm for the solution of the problem created with a true

[1] Alexa Internet Inc. (1996). The top 500 sites on the web [online]. Website: https://www.alexa.com/topsites [accessed 01 November 2020].

positive rate of 85% and a false positive rate of 5%.

3. Dataset

In order to test the proposed model, encrypted traffic data generated from real traffic data was needed. For this, the data set prepared by Draper-Gil [12] was preferred. This dataset is generated from real traffic data and includes class label. Two users named Alice and Bob were created and packets were captured using simple mail transfer protocol secure (SMTPS), Post Office Protocol 3 (POP3), Browser, Skype and similar applications. The traffic class and the applications used in the content are given in Table 1. There are 2 scenarios in the data set and it is shown in Figure 1. In Scenario A, the purpose is to define the traffic that is non-VPN with VPN, and after this definition is done, the traffic is classified into subcategories as browsing, email, chat, streaming, file transfer, VoIP and peer-to-peer (P2P). For this process, Scenario A is done in 2 different steps. In Scenario B, the whole process is made in a single classification and consists of 14 classes in total.

Table 1. List of class and applications [12].

Traffic class	Applications
Web browsing	Chrome and Firefox
Email	SMTPS, POP3S and IMAPS
Chat	ICQ, Skype, Facebook and Hangout
Streaming	YouTube and Vimeo
File transfer	Skype, FTPS, SFTP
VoIP	Facebook, Skype and Hangout voice calls
P2P	uTorrent and Bittorent

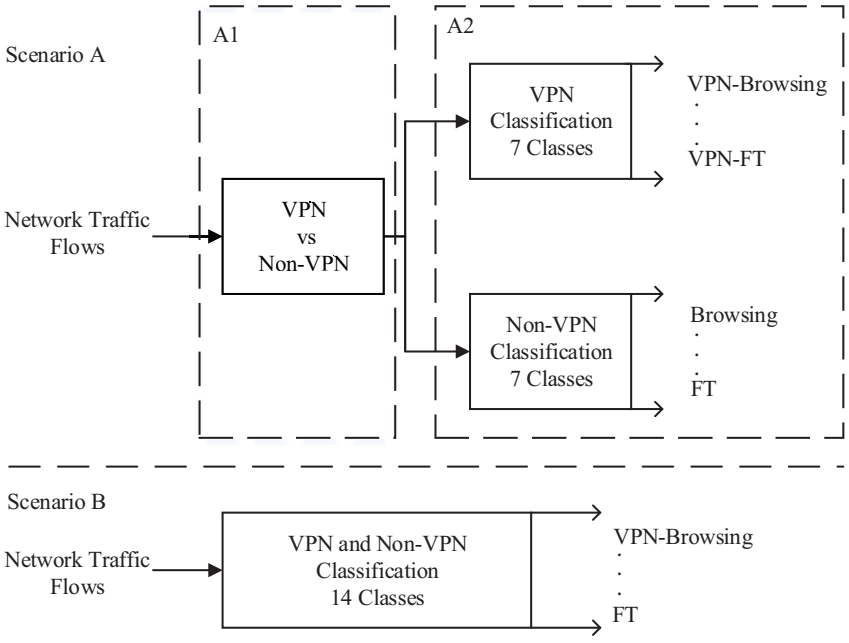

Figure 1. Dataset scenarios [12].

A dataset was prepared by Draper-Gil [12] et al. using a time-based approach together with flow-based detection. For the time-based approach, each scenario has datasets of 15, 30, 60 and 120 s. Through these datasets, time-based classification of traffic can be done.

There are 24 features in the dataset and one of them is the label class. The other 3 properties are duration, the number of bits occurring in 1 s, and the number of packets occurring in 1 s. The remaining 20 features consist of the maximum, minimum, average and standard deviation values of 5 different features. These 5 features are defined as forward arrival time, backward arrival time, arrival time between flows, actively used time and idle time [12]. Source ip address, destination ip address, source port and destination port information are not used as a feature. Features and description are shown in Table 2.

Table 2. List of features and description [12].

Feature Name	Description
duration	Total time of flow.
total_fiat	The total time among two network packets in going forward.
min_fiat	The minimum time among two network packets in going forward.
max_fiat	The maximum time among two network packets in going forward.
mean_fiat	The mean time among two network packets in going forward.
total_biat	The total time among two rearward going network packets.
min_biat	The minimum time among two rearward going network packets.
max_biat	The maximum time among two rearward going network packets.
mean_biat	The mean time among two rearward going network packets.
flow_pkts_per_second	The total number of packets generated per second.
flow_bytes_per_second	The total number of bytes generated per second.
min_flowiat	The minimum time among two network packets sent in both way.
max_flowiat	The maximum time among two network packets sent in both way.
mean_flowiat	The mean time among two network packets sent in both way.
std_flowiat	The standard deviation time among two network packets sent in both way.
min_active	The minimum amount of time a flow has been in use prior to it goes into idle state.
mean_active	The mean amount of time a flow has been in use prior to it goes into idle state.
max_active	The maximum amount of time a flow has been in use prior to it goes into idle state.
std_active	The standard deviation amount of time a flow has been in use prior to it goes into idle state.
min_idle	The minimum amount of time a flow has been in idle prior to it goes into active state.
mean_idle	The mean amount of time a flow has been in idle prior to it goes into active state.
max_idle	The maximum amount of time a flow has been in idle prior to it goes into active state.
std_idle	The standard deviation amount of time a flow has been in idle prior to it goes into active state.

4. Methodology and methods

The classification of encrypted traffic is made by using the size of the outgoing and incoming packets and the round-trip time features on the network. Since each application has a different structure, values like the size, number and time of outgoing and incoming packets can be different and these differences are used for classification purposes. Using the features specified, the classification of packets encrypted with VPN and encrypted without VPN using XGBoost, decision tree and random forest algorithms was performed in this study.

4.1. Proposed model architecture

In the proposed architecture for the classification of encrypted traffic, firstly the data in the dataset must be preprocessed. For this purpose, nonnumeric class labels such as VPN-browsing, VPN-FT in the dataset were converted to numerical values. For example, after this process, a value of 0 for VPN-browsing and 6 for VPN-FT has been assigned. Each class is labelled with a different number. In addition to this operation, the values in the data set were reduced to a value between 0 and 1 by normalization. After data preprocessing is completed, the data is ready for use.

Before the classification process, the features in the dataset should be analysed in order to increase the proposed system performance. For this purpose, the data in the dataset are weighted according to their importance. The sum of the weights of all the features is equal to 1. The weighted graph of the features in the dataset is shown in Figure 2. which contains the weight values of the features in the vertical part and the features names in the horizontal part. Eight features with a weight value below 0.03 and affecting the classification least were eliminated, and a total of 15 features were chosen.The selected list of features is shown in Table 3.

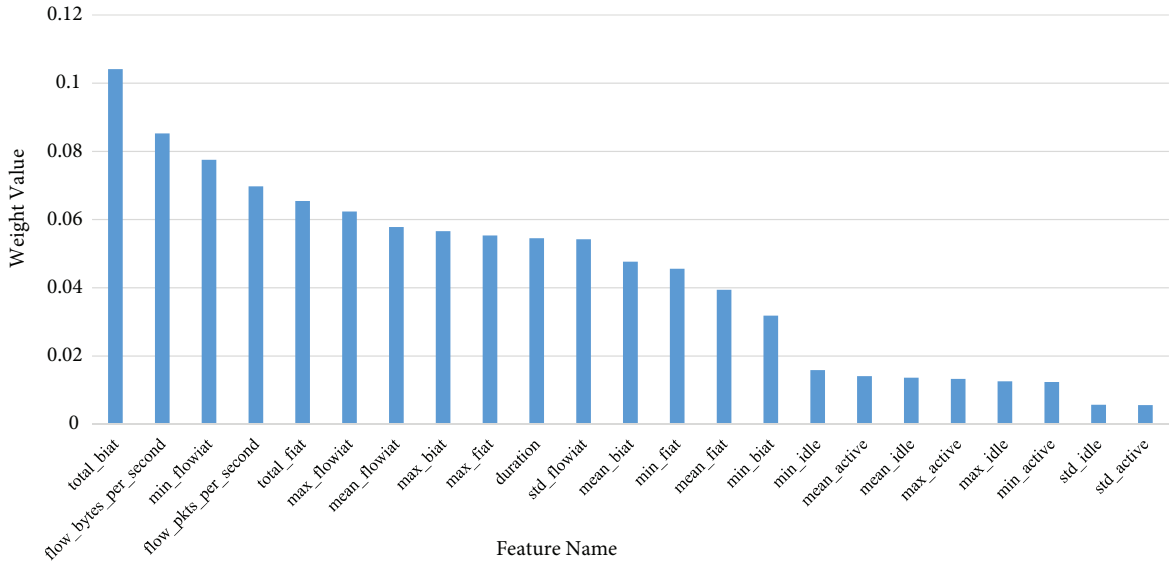

Figure 2. Weighted values of dataset features.

It is necessary to carry out a learning process for the current problem of the model to be written. Training data are needed to train the model and test data to test the success of the model. For this reason, 30% of the existing data set was divided as test data and the training was done on the remaining 70% data. The dividing

Table 3. List of selected features.

Feature Name	Weight Value
total_biat	0.104074357
flow_bytes_per_second	0.085232544
min_flowiat	0.077517041
flow_pkts_per_second	0.069723517
total_fiat	0.06542718
max_flowiat	0.062340503
mean_flowiat	0.057769575
max_biat	0.056603319
max_fiat	0.055294986
duration	0.054522434
std_flowiat	0.05422234
mean_biat	0.047622901
min_fiat	0.045579098
mean_fiat	0.039412682
min_biat	0.031789681

process was made randomly by using the sci-kit learn library. Only training data were used in the training phase, and test data were not used in the training phase. There is an imbalance in the class distributions of the data on the selected dataset. In order to eliminate the imbalance in the data distribution, data balancing was performed using the SMOTE [22] algorithm. Before the SMOTE process, the browsing class with the highest number of data had 2500 data and the streaming class with the lowest number of data had 475 data. After the SMOTE process, the data distribution counts of the classes in the dataset are equalized and all class had same counts of data.

Prior to the use of algorithms, hyperparameter optimization was carried out in order to provide the best performance of the installed system. Along this process, the parameter values used by the algorithms are given in a certain range and format and the parameters providing the best performance are chosen. The selected parameters and their values are given in Table 4. GridSearchCV library was used during parameter selection which is made separately for each algorithm.

After parameter optimization, the model was trained with training data and its performance was tested with test data. The stages of the proposed model are shown in Figure 3. Raw data primarily goes through digitization and normalization processes. After these processes, feature selection is made and the dataset is divided into test and training sets. Dataset imbalance is eliminated by sampling on the training set. Finally, the classification process is performed by selecting hyperparameters for machine learning algorithms.

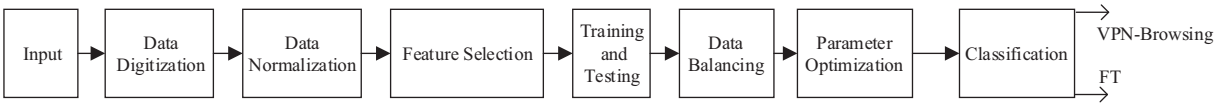

Figure 3. The stages of the proposed model.

Table 4. List of chosen hyperparameters.

Algorithms	Hyperparameters	Values	Description
Decision tree	class_weight	balanced	Weight distribution between classes.
	max_depth	20	The maximum depth of the tree.
	max_features	auto	The number of features to size up when search the best split.
	min_samples_leaf	1	The number of instances that must be in a leaf node.
	min_samples_split	2	The minimum number of instances required to split a node.
	splitter	best	The method used to select the split at the node.
Random forest	n_estimators	300	Number of trees to be used when creating a forest.
	criterion	entropy	The function for measuring the quality of a split.
	class_weight	balanced_subsample	Weight distribution between classes.
	max_features	sqrt	The number of features to size up when search the best split.
XGBoost	eta	0.2	Step size reduction used in update to halt overfitting.
	max_depth	50	The maximum depth of the tree.
	gamma	2	Minimum loss required for tree splitting.
	min_child_weight	4	The minimum amount of weight required for a child.
	subsample	0.83	Subsample rate of training sets
	max_bin	256	Biggest number of separate bins to bucket ongoing features.
	objective	multisoftmax	Multiclass classification method.
	tree_method	hist	The tree creation algorithm.
	num_class	7	Number of classes.
	learning_rate	0.1	The amount that the weights are updated during training.

4.2. Algorithms

Machine learning algorithms are used to solve the current problem. Three different algorithms were preferred within the scope of the study.

4.2.1. Decision tree

It is a widely used machine learning algorithm developed by Quinlan [23]. Decision tree algorithm composes many nodes in problem solution and establishes relationship between these nodes. The first node is called a root node. It creates a decision tree through these nodes. Decision tree algorithm has the ability to operate on categorical and numerical data.

4.2.2. Random forest

The random forest algorithm was evolved by Breiman [24] in 2001. This algorithm builds a random forest using decision trees. Each decision tree begets a class, and the final result is reached by evaluating the information from all decision trees. The random forest algorithm derives its strength from the use of many decision trees together.

4.2.3. XGBoost

The XGBoost algorithm is based on gradient boosting algorithm, but performance improvement has been achieved through optimization. Together with the parallel and distributed calculation management, it can completes the learning process much faster [25–27]. It can also perform cache optimization for efficient use of hardware and distributed computing to process large models quickly [25, 28]. XGBoost can work ten times faster than existing well-known remediation on a single machine [29]. Jiong Wang [30] et al. stated in his study for android malware detection that the XGBoost algorithm works much faster than the SVM algorithm when the dataset increases. Zhuo Chen [28] et al. observed that the XGBoost algorithm works faster than SVM and general boosting decision tree (GBDT) algorithms in his study, which he performed to detect distributed denial of service attack (DDOS) attacks in software defined networking (SDN) based cloud systems.

4.3. Performance measure

Many metrics are used to evaluate the performance of the models suggested in the literature. The most prevalent of these metrics are accuracy, precision, recall and f-measure. These metrics are figured out on the values of true positive (TP), false positive (FP), true negative (TN) and false negative (FN). If the data that is originally abnormal is classified as abnormal by the model, it is called by true positive. False positive data that is actually normal is defined as abnormal by the model. The data that is really normal is considered to be true negative if it is categorized as normal by the model. False negative data that is indeed abnormal is labelled as normal by the model.

The most popularly used success criterion in the literature is accuracy (Acc) which is computed by dividing the number of correctly classified data by the number of all data. The formula for calculating the accuracy criterion is shown in Equation 1.

$$Acc = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}+\mathrm{TN}} \tag{1}$$

Although accuracy is extensively used, it is not sufficient to measure model performance alone. If the ratio of the number of the data known as positive is too low compared to the number of data known as negative, accuracy will not be a very meaningful criterion [31]. For this aim, sensitivity, precision and f-measure are used in addition to the accuracy value in the literature. The recall is calculated by dividing the data correctly classified as abnormal by the number of all abnormal data [32]. The formula for calculating the recall criterion is shown in Equation 2. Precision is reckoned by the ratio of the number of data classified as abnormal to the number of all data classified as abnormal by the model. The formula for calculating the precision criterion is shown in Equation 3. F-measure is figured out with the harmonic mean of the precision and recall criteria. Equation 4 demonstrates the formula for calculating the F-criterion.

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{2}$$

$$Precision = \frac{\text{TP}}{\text{TP}+\text{FP}} \tag{3}$$

$$F1 - Measure = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall}+\text{Precision}} \tag{4}$$

5. Experiment and results

There are two different scenarios in the dataset used in the study. However, the data set consists of data recorded as 15, 30, 60, and 120 s [12]. In this way, time-based analysis can be performed in addition to flow-based analysis. Three different algorithms were tested separately for each scenario and each time-based dataset and the results were compared. Weighted average values are given for precision, recall and f1-measure metrics in multiple classifications.

5.1. Scenario A1

There are two different subscenarios in Scenario A. In the first of these subscenarios, the classification process is made whether the incoming traffic packets are encrypted with or without VPN. At this stage, binary classification is done. For this scenario, there are four time-based datasets consisting of 15, 30, 60, and 120 s in the dataset. It has been observed that the XGBoost algorithm gives the most thriving result for all time frames of this scenario. In Figure 4, accuracy, precision, recall and f1-measure values for three different algorithms are shown. The highest success rate was achieved in a 15-s dataset. In this dataset, the best accuracy rate is 93.02%, precision is 93.04%, recall is 93.02% and f1-measure is 93.01%.

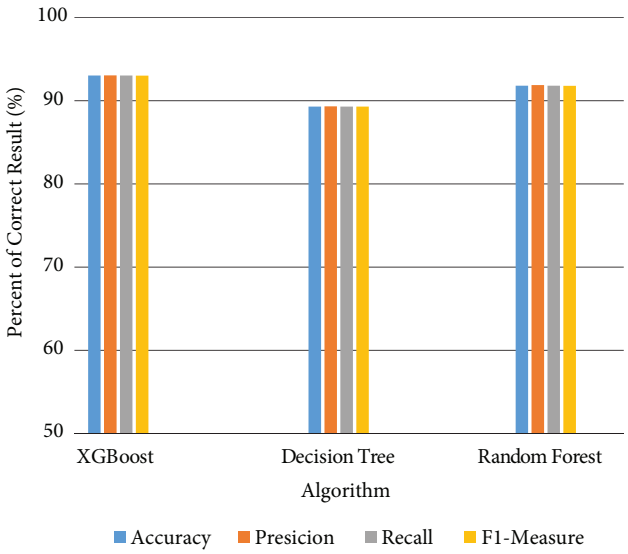

Figure 4. Scenario A1 test results.

When the time-based analysis was done, it was determined that the success rate of the model decreased from 15 s to 120 s. Time-based change graph is shown in Figure 5.

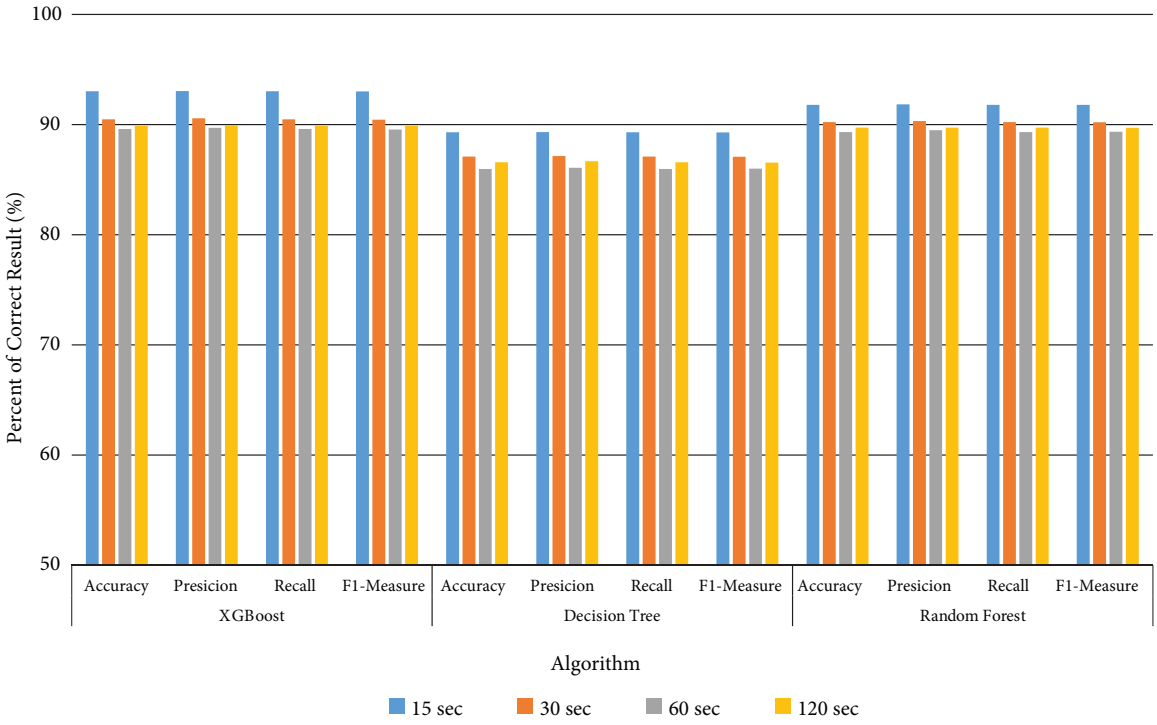

Figure 5. Scenario A1 time-based test results.

5.2. Scenario A2

In Scenario A2, traffic encrypted VPN and without VPN is classified into 7 different categories with multiple classification. Success metrics for VPN and success metrics for traffic encrypted without VPN are given separately.

The XGBoost algorithm has been found to give the most successful result for non-VPN encrypted traffic in Scenario A2. The accuracy rate of the XGBoost algorithm is 94.53%. The weighted average value is taken, the precision value is 94.5%, the recall value is 94.53% and the f1-measure value is 94.48%. Analysed at the class based, the category with the highest success rate is the VoIP class. The category with the lowest success rate was found to be the streaming class. The test results of traffic encrypted for 15 s without VPN are represent in Figure 6. In the case of time-based analysis, it was seen that the success rate of the correct model decreased from 15 s dataset to 120 s dataset.

In Scenario A2 VPN, it has been observed that the XGBoost algorithm gives the best conclusion for encrypted traffic. With XGBoost algorithm, 90.75% precision, 90.76% recall and 90.73% f1-measure value were obtained. VoIP is again the class that can be detected with the highest success rate. However, in this scenario, the category with the lowest success rate is the chat class. Figure 7 demonstrates the success rates of traffic encrypted with Scenario A2 VPN 15-s.

Comparing the VPN results with non-VPN, it can be clearly seen that the model has a higher success rate for non-VPN traffic in scenario A2. The classification performance of the model decreases as traffic encrypted with VPN is subjected to encapsulation.

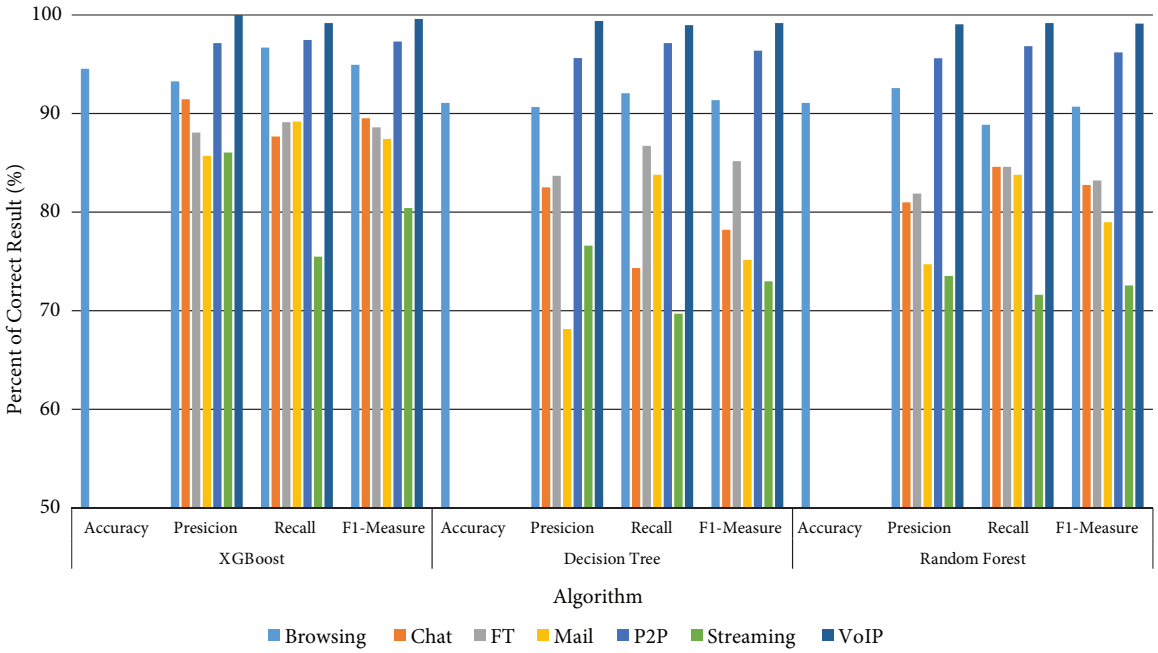

Figure 6. Scenario A2 non-VPN 15-s test results.

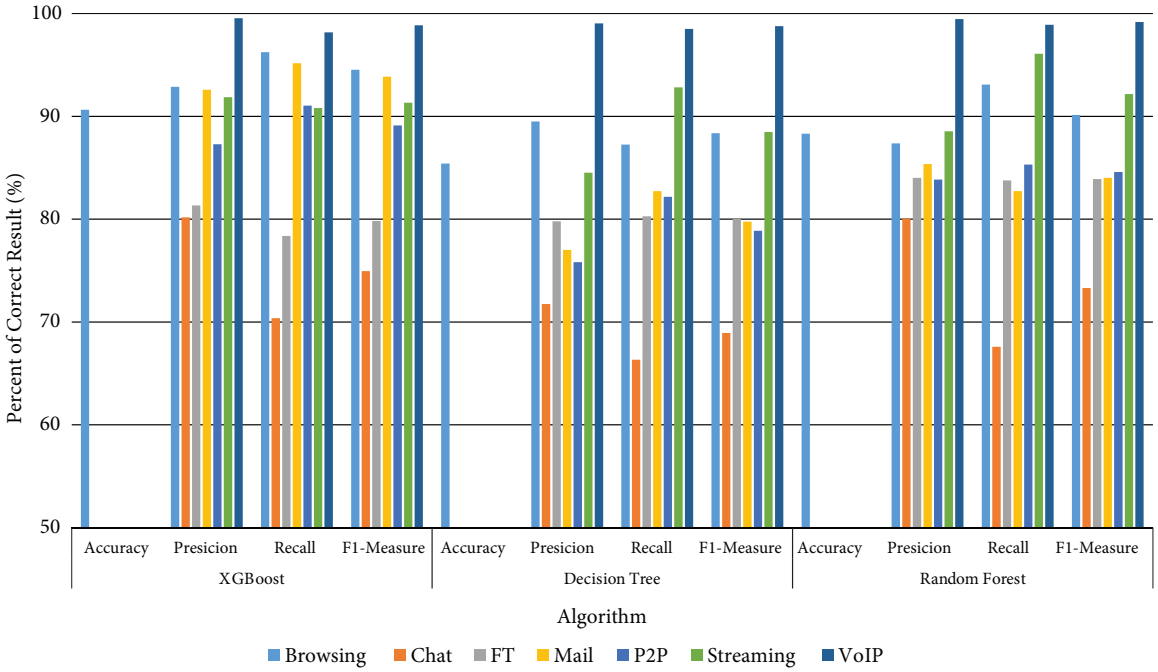

Figure 7. Scenario A2 VPN 30-s test results.

5.3. Scenario B

In Scenario B, multiple classifications were made among 14 classes without distinguishing between VPN and non-VPN. The XGBoost algorithm gave the most successful results similar to other scenarios. The XGBoost

algorithm has an accuracy rate of 85.79%, a weighted average value of 86.07%, a precision value of 85.76% and an F1-criterion value of 85.75%. In Scenario B, the class detected with the lowest success rate is VPN-P2P and the class detected with the highest success rate is the VPN-VoIP class. The success rates of the traffic encrypted with Scenario B VPN 15-s are shown in Figure 8. Also it was observed that the success rate of the model decreased from 15 s to 120 s.

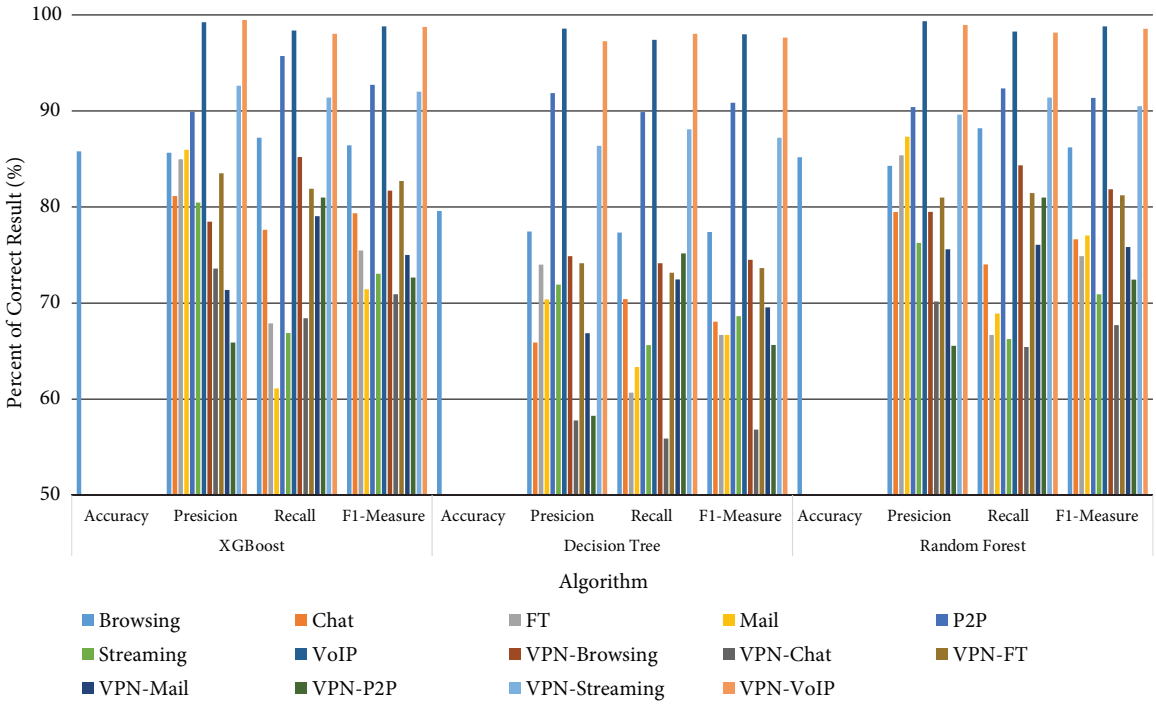

Figure 8. Scenario B VPN 15-s test results.

When the results are examined in general, it is seen that VoIP traffic is detected with the highest success rate among all scenarios. But the lowest detection rate class is different in every scenario. While the proposed model achieves the highest success rates on encrypted traffic without VPN, the success rate drops from 94.53% to 85% in cases where there is VPN traffic. Except for Scenario A2 VPN traffic, datasets captured with 15 s time give the best outcomes. For Scenario A2 VPN dataset, the data set captured with 30 s time yielded the best results. Accuracy rates and weighted f1-measure values for all scenarios are shown in Table 5.

The computer where the study was carried out has an Intel Core i7-8650U 1.9GHz processor, 16 GB Ram and a GeForce MX130 graphics card. Comparing the 3 algorithms based on time, the fastest and most effective algorithm is the decision tree algorithm. It completes training and testing processes in an average of 0.19 s. Decision tree algorithm is followed by XGBoost algorithm and it performs operations in an average of 2.37 s. The algorithm with the highest training and testing time with an average of 8.96 s is the random forest algorithm. Time analysis of the 3 algorithms for all scenarios is shown in Table 6.

6. Discussion and conclusion

In this study, a machine learning based approach was used for the classification of encrypted traffic using XGBoost, decision tree and random forest algorithms. ISCX VPN-NonVPN data set, which is open source,

Table 5. Accuracy and f1-measure values for all scenarios.

Algorithm	XGBoost		Decision tree		Random forest	
Scenario	Accuracy	F1-measure	Accuracy	F1-measure	Accuracy	F1-measure
A1 15 s	93.02	93.01	89.29	89.28	91.79	91.78
A1 30	90.47	90.44	87.09	87.08	90.30	90.27
A1 60 s	89.59	89.54	85.97	85.99	89.32	89.34
A1 120 s	89.88	89.87	87.52	87.51	90.03	90.02
A2 non-VPN 15 s	94.53	94.48	91.08	91.03	93.65	93.57
A2 non-VPN 30 s	92.99	92.86	88.57	88.49	92.69	92.58
A2 non-VPN 60 s	92.99	92.86	88.70	88.65	92.38	92.27
A2 non-VPN 120s	92.59	92.43	89.29	89.35	92.35	92.13
A2 VPN 15 s	89.35	89.14	85.40	85.37	88.30	88.14
A2 VPN 30 s	90.76	90.73	85.47	85.36	90.64	90.60
A2 VPN 60 s	86.98	86.55	83.40	83.43	87.20	86.87
A2 VPN 120 s	87.04	86.57	83.16	82.79	86.98	86.61
B 15 s	85.79	85.75	79.57	79.59	85.17	85.12
B 30 s	85.46	85.28	79.09	78.99	84.72	84.65
B 60 s	83.30	83.43	77.40	77.53	82.64	82.59
B 120 s	80.75	80.92	76.59	76.48	80.98	80.75

Table 6. Time cost values for all scenarios.

Scenario	XGBoost (s)	Decision tree (s)	Random forest (s)
A1 15 s	3.11	11.09	0.26
A1 30 s	1.54	8.35	0.17
A1 60 s	1.72	9.61	0.2
A1 120 s	1.41	6.24	0.1
A2 non-VPN 15 s	1.66	6.23	0.14
A2 non-VPN 30 s	1.29	4.73	0.11
A2 non-VPN 60 s	1.66	6.63	0.15
A2 non-VPN 120 s	1.07	3.98	0.08
A2 VPN 15 s	1.41	3.69	0.09
A2 VPN 30 s	1.8	5.75	0.13
A2 VPN 60 s	1.67	4.9	0.11
A2 VPN 120 s	1.62	3.86	0.09
B 15 s	5.15	22.92	0.52
B 30 s	4.24	16.43	0.36
B 60 s	5.03	18.15	0.42
B 120 s	3.67	10.91	0.25
Average time	2.378125	8.966875	0.19875

is used. Before using machine learning, data was preprocessed and feature selection was made. In addition, hyperparameter selection was done using the GridSearchCV library.

The XGBoost algorithm has a higher accuracy rate for each scenario and has yielded more successful results for this problem than the decision tree and random forest algorithms. In Scenario A1, the highest precision rate was obtained in the 15 s dataset and was 93.04%. This result was 3% more successful compared to the study conducted by H. Lashkari [12]. In non-VPN scenario, which is the first part of Scenario A2, the best classification was achieved in 15 s data set and with 94.53% accuracy. Although the results we obtained in the A2 non-VPN, A2 VPN and B scenarios provided a higher success rate than the study done by Draper-Gil [12], it was slightly better than the study conducted by Caicedo-Muñoz [14]. In Scenario A2 VPN, the success rate of the model has decreased according to the non-VPN scenarios and an accuracy rate of 90.76% has been achieved in the 30 s data set. Comparisons with similar studies are shown in Table 7. It has been observed that the proposed model performs the classification of encrypted traffic with high success rates for each scenario. Compared to similar studies, more successful results were obtained for each scenario.

Table 7. Comparison with similar studies.

Studies	Scenario	Accuracy	Precision	Recall
Draper-Gil [12]	Scenario A1	*	90.6	*
	Scenario A2 non-VPN	*	89	*
	Scenario A2 VPN	*	84	*
	Scenario B	*	78.3	*
Caicedo-Muñoz [14]	Scenario A2 non-VPN	93.19	*	*
	Scenario A2 VPN	89.03	*	*
	Scenario B	84.81	*	*
Present study	Scenario A1	93.02	93.04	93.02
	Scenario A2 non-VPN	94.53	94.49	94.52
	Scenario A2 VPN	90.76	90.75	90.76
	Scenario B	85.79	0.8607	0.8579

*No data.

When the class-based evaluation is made in Scenarios A2 and B, where multiple classifications are made, the class detected with the highest success rate is the VoIP class. VoIP encrypted traffic can be detected with accuracy rates close to 100%. The proposed model detects VoIP traffic with a much higher success rate compared to the study done by Saqib [15]. The class with the lowest success rate in the non-VPN dataset is the streaming class and the accuracy rate is 86.03% in Scenario A2. Class of chat has the minimum success rate in Scenario A2 VPN dataset and its accuracy rate is 84.64%. The VPN-P2P class has the least success rate of Scenario B with 65.88% accuracy, and it highly reduces the model performance for Scenario B.

When traffic classification is performed with a high success rate, cyber security experts and network experts can implement security policies that will automatically block the traffic when unwanted traffic occurs. By integrating the proposed model with perimeter network security products [33] such as firewalls, intrusion prevention system and data leakage prevention systems, unwanted traffic in the VPN or SSL tunnel can be automatically dropped and generate an alarm to inform experts. In addition, classifying the traffic in the tunnel and making it meaningful is also very important for cyber incident analysis studies.

Artificial intelligence-based classification studies for encrypted traffic that started in the 1990s still continue today. Classification of encrypted traffic is much more difficult than classification of unencrypted traffic because deep packet analysis cannot be performed. With the increase in the rate of internet usage, new applications and new protocols are emerging. Applications encrypt the traffic going over the internet with encryption methods and protocols in order to ensure confidential and secure communication. In future studies, a new dataset can be created to identify new applications and protocols, through which classification can be done. As the studies in this field continue, the ability of both network experts and cyber security experts to make analysis over encrypted traffic will increase.

References

[1] Brissaud P, François J, Chrisment I, Cholez T, Bettan O. Transparent and Service-Agnostic Monitoring of Encrypted Web Traffic. IEEE Transactions on Network and Service Management 2019; 16 (3): 842-856. doi: 10.1109/TNSM.2019.2933155

[2] Devi TR. Importance of cryptography in network security. In: International Conference on Communication Systems and Network Technologies; Gwalior, India; 2013. pp. 462-467. doi: 10.1109/CSNT.2013.102

[3] Nguyen NH. SSL/TLS Interception Challenge from the Shadow to the Light. Rockville, MD, USA: The SANS Institute (SANS Information Security Reading Room), 2019.

[4] Radivilova T, Kirichenko L, Ageyev D, Tawalbeh M, Bulakh V. Decrypting SSL/TLS traffic for hidden threats detection. In: IEEE 9th International Conference on Dependable Systems, Services and Technologies; Kiev, Ukraine; 2018. pp. 143-146. doi: 10.1109/DESSERT.2018.8409116

[5] Wood D, Stoss V, Chan-Lizardo L, Papacostas GS, Stinson ME. Virtual private networks. In: International Conference on Private Switching Systems and Networks; London, UK; 1988. pp. 132-136.

[6] Mao H, Zhu L, Qin H. A Comparative research on SSL VPN and IPSec VPN. In: 8th International Conference on Wireless Communications, Networking and Mobile Computing; Shanghai, China; 2012. pp. 1-4. doi: 10.1109/WiCOM.2012.6478270

[7] Piscitello D, Chapin AL. Open Systems Networking: Tcp/Ip and Osi. Reading, MA, USA: Addison-Wesley, 1993.

[8] Stanton R. Securing VPNs: comparing SSL and IPsec. Computer Fraud & Security; 2005 (9): 17-19. doi: 10.1016/S1361-3723(05)70254-2

[9] El-Maghraby RT, Elazim NMA, Bahaa-Eldin AM. A survey on deep packet inspection. In: 12th International Conference on Computer Engineering and Systems; Cairo, Egypt; 2017. pp. 188-197. doi: 10.1109/ICCES.2017.8275301

[10] Alshammari R, Zincir-Heywood AN. Machine learning based encrypted traffic classification: identifying SSH and Skype. In: IEEE Symposium on Computational Intelligence for Security and Defense Applications; Ottawa, ON, Canada; 2009. pp. 1-8. doi: 10.1109/CISDA.2009.5356534

[11] Di Mauro M, Longo M. Revealing encrypted WebRTC traffic via machine learning tools. In: 12th International Joint Conference on e-Business and Telecommunications; Colmar, France; 2015. pp. 259-266.

[12] Draper-Gil G, Lashkari AH, Mamun MSI, Ghorbani AA. Characterization of encrypted and VPN traffic using time-related features. In: Proceedings of the 2nd International Conference on Information Systems Security and Privacy; Rome, Italy; pp. 407-414. doi: 10.5220/0005740704070414

[13] Seddigh N, Nandy B, Bennett D, Ren Y, Dolgikh S et al. A framework & system for classification of encrypted network traffic using machine Learning. In: 15th International Conference on Network and Service Management; Halifax, NS, Canada; 2019. pp. 1-5. doi: 10.23919/CNSM46954.2019.9012662

[14] Caicedo-Muñoz JA, Espino AL, Corrales JC, Rendón A.QoS-Classifier for VPN and Non-VPN traffic based on time-related features.Computer Networks 2018; 144: 271-279. doi: 10.1016/j.comnet.2018.08.008

[15] Saqib NA, Shakeel Y, Khan MA, Mahmood H, Zia M. An effective empirical approach to VoIP traffic classification. Turkish Journal of Electrical Engineering & Computer Sciences 2017; 25 (2): 888-900. doi: 10.3906/elk-1501-126

[16] Zhang Y, Zhao S, Zhang J, Ma X, Huang F. STNN: a novel TLS/SSL encrypted traffic classification system based on stereo transform neural network. In: IEEE 25th International Conference on Parallel and Distributed Systems; Tianjin, China; 2019. pp. 907-910. doi: 10.1109/ICPADS47876.2019.00133

[17] Chari M, Srinidhi H, Somu TE. Network traffic classification by packet length signature extraction. In: IEEE International WIE Conference on Electrical and Computer Engineering; Bangalore, India; 2019. pp. 1-4. doi: 10.1109/WIECON-ECE48653.2019.9019918

[18] Pradhan A, Behera S, Dash R. Hybrid RBFN based encrypted SSH traffic classification. In: 5th International Conference on Signal Processing and Integrated Networks; Noida, India; 2018. pp. 264-269. doi: 10.1109/SPIN.2018.8474059

[19] Yang Y, Kang C, Gou G, Li Z, Xiong G. TLS/SSL encrypted traffic classification with autoencoder and convolutional neural network. In: IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems; Exeter, UK; 2018. pp. 362-369. doi: 10.1109/HPCC/SmartCity/DSS.2018.00079

[20] Al-Obaidy F, Momtahen S, Hossain MF, Mohammadi F. Encrypted traffic classification based ML for identifying different social media applications. In: IEEE Canadian Conference of Electrical and Computer Engineering; Edmonton, AB, Canada; 2019. pp. 1-5. doi: 10.1109/CCECE.2019.8861934

[21] Khatouni AS, Zincir-Heywood N. Integrating machine learning with off-the-shelf traffic flow features for HTTP/HTTPS traffic classification. In: IEEE Symposium on Computers and Communications; Barcelona, Spain; 2019. pp. 1-7. doi: 10.1109/ISCC47284.2019.8969578

[22] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research 2002; 16: 321-357. doi: 10.1613/jair.953

[23] Quinlan JR. Induction of decision trees. Machine Learning 1986; 1: 81-106. doi: 10.1007/BF00116251

[24] Breiman L. Random forests. Machine Learning 2001; 45: 5-32. doi: 10.1023/A:1010933404324

[25] Sukhpreet SD, Abdullah-Al N, Robert A. Effective intrusion detection system using XGBoost. Information 2018; 9 (7): 1-24. doi: 10.3390/info9070149.

[26] Huajun C, Ranjin D, Zhen L, Huan X. Android Malware classification using XGBoost based on Images. In: IEEE 4th Information Technology and Mechatronics Engineering Conference Patterns; Chongqing, China; 2018. pp. 1358-1362. doi: 10.1109/ITOEC.2018.8740537

[27] Di W, Peiqi G, Peng W. Malware detection based on cascading XGBoost and cost sensitive. In: International Conference on Computer Communication and Network Security; Xi'an, China; 2020. pp. 201-205. doi: 10.1109/CCNS50731.2020.00051

[28] Zhuo C, Fu J, Yijun C, Xin G, Weirong L, Jun P. XGBoost Classifier for DDoS attack detection and analysis in SDN-based cloud. In: IEEE International Conference on Big Data and Smart Computing; Shanghai; 2018. pp. 251-256. doi: 10.1109/BigComp.2018.00044

[29] Chen T, Guestrin C. XGBoost: a scalable tree boosting system. In: KDD 2016: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; San Francisco, CA, USA; 2016. pp. 785-794. doi: 10.1145/2939672.2939785

[30] Jiong W, Boquan L, Yuwei Z. XGBoost-Based Android Malware Detection. In: 13th International Conference on Computational Intelligence and Security (CIS); Hong Kong, China; 2017. pp. 268-272. doi: 10.1109/CIS.2017.00065

[31] Uddin MF. Addressing accuracy paradox using enhanched weighted performance metric in machine learning. In: Sixth HCT Information Technology Trends; Ras Al Khaimah, United Arab Emirates; 2019. pp. 319-324. doi: 10.1109/ITT48889.2019.9075071

[32] Powers DMW. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. Adelaide, Australia: Technical Report School of Informatics and Engineering, Flinders University, 2007.

[33] Göksel U, Mustafa A, İbrahim AD, Murat D. Perimeter network security solutions: a survey. In: 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies; Ankara, Turkey; 2019. pp. 1-6. doi: 10.1109/ISMSIT.2019.8932821