

# A New Method of Testing Machine Learning Models of Detection for Targeted DDoS Attacks

Mateusz Kozlowski<sup>a</sup> and Bogdan Ksiezopolski<sup>b</sup>

*Institute of Computer Science, Maria Curie-Skłodowska University, Akademicka 9, 20-033 Lublin, Poland*


**Keywords:** DDoS, Targeted Attacks, Machine Learning, Testing Methods.


**Abstract:** Distributed Denial of Service (DDoS) is one of the most popular attacks on the Internet. One of the most popular classes of DDoS attacks is the flood-based, which sends huge amounts of packets to the victim host or infrastructure, causing an overload of the system. One of the attack mitigation systems is based on machine learning (ML) methods, which in many cases has a very high accuracy rate (0.95 – 0.99). Unfortunately, most ML models are not resistant against targeted DDoS attacks. In this article, we present the targeted attacks to the DDoS ML-based mitigation models, which have a high accuracy. After this, we propose a new method of testing ML-based models against targeted DDoS attacks.

## 1 INTRODUCTION

Distributed Denial of Service attacks have become one of the biggest problems for technology companies (Bouyeddou et al., 2020). ENISA reports indicate (ENISA, 2020) that in 2019 DDoS attacks increased by 241%. The most popular attacks are flood-based ones and based mostly on the Layer 4 OSI model (79% of DDoS attacks were syn-floods, 9% were UDP-based floods). One of the key elements in the mitigation of DDoS attacks is the detection of unwanted traffic and distinguishing this from legal traffic. DDoS detection systems are implemented with one of the following two approaches: signature-based or anomaly-based detection (Hodo et al., 2017). The former is based on training detection mechanisms to identify malicious traffic and the latter is based on training using the patterns of normal activity. The training methods can be categorized according to the detection method used: machine learning models, artificial-neural-networks, genetic algorithms, statistics, open flow or deterministic behavioral analysis. In the literature, such detection systems can easily be found, and are typically focused on models where the accuracy is above 0.99 (Braga et al., 2010; Idhammad et al., 2018; Pei et al., 2019; Santos et al., 2019; Sood, 2014).

Accuracy close to 1.00 is a great result and, if so, has the problem of DDoS attack detection been solved? The problems with the current DDoS systems vary, including isolating the attack traffic from the legal traffic. Systems are able to determine that there is an attack taking place, but it is difficult to separate one form of traffic from another and also perform the appropriate action on the attack traffic. Another problem is the targeted DDoS attacks (Bouyeddou et al., 2020; Sood, 2014). In the event of such attacks, legal traffic patterns are created, and the attack will be executed in such a way that it follows these patterns. In the event of such prepared DDoS attacks, will the effectiveness of the detection models still be close to 0.99? For this article, we conduct several targeted DDoS attacks on selected machine learning detection models that we built, which had been described in previous literature, and with which we had an efficiency of nearly 0.99. We found that the effectiveness of DDoS attack detection models has significantly decreased and, in many cases, is only a few percentage points. The results obtained prompted us to propose a new method of testing machine learning classifiers, which involves performing additional tests on the basis of specially-prepared data.

<sup>a</sup>  <https://orcid.org/0000-0001-8683-402X>

<sup>b</sup>  <https://orcid.org/0000-0003-1904-3222>

The main contributions of the paper are as follows:

- performing targeted UDP DDoS attacks on machine learning models based on single packets and time series;
- introducing the new framework of testing ML models with extra tests; and
- creating the algorithm of generating a directed DDoS attack, which can be used as extra tests in the proposed framework.

The article is organized as follows. In section 1 the introduction is presented. The section 2 describes the related work in the field. In section 3 we present the new method of testing ML models against targeted DDoS attacks. In section 4 the case studies are presented, where the UDP-based targeted DDoS attacks using single packets and time series are presented. In section 5 we present the conclusions.

## 2 RELATED WORK

DDoS detection systems can take two approaches to the data structure: a single packet selection or a flow data (a time series of packets). In general, the authors of previous studies rely on the classical method of splitting datasets, and none of them test their models on the use case of targeted attacks.

The simplest version of an anomaly-based detection system used on single packet selection methods was proposed in (Peraković et al., 2017). This version focused on building artificial neural networks using basic parameters like source IP address, destination IP address, protocol and packet length. The dataset was split among training, validation, and test set and model, and achieved a high accuracy rate of 0.95.

In (Saied et al., 2015), the authors also built an ANN model to predict the legality of packets based on 4 parameters: source IP address, port source, destination port and header length. The authors trained the model on 80% of the dataset and then validated on another 20% of samples with 0.95 accuracy. The authors did not use other online datasets in order to train their approach, as they wanted to learn about the behavior of the DDoS and in the context of genuine traffic.

In (Pei et al., 2019), the authors used Random Forest and SVM algorithms on single packet features, an approach that has a high detection rate of 0.99 based on the classical testing framework – wherein the dataset is divided to train and test and use the cross-validation method to score the model.

The second approach of building DDoS detection system is focused on analyzing time series data. In (Braga et al., 2010) six flow features were presented, inter alia: the average of packets per flow; the average of bytes per flow; the average of duration per flow; the percentage of pair-flows; the growth of single-flows and different ports, all of which were analyzed using self-organizing maps. The authors achieved a 0.99 accuracy on the KDD-99 dataset.

In (Idhammad et al., 2018), the Random Forest ensemble learning method was used, and processed over 14 parameters, such as flow duration, number of transmitted packets, bytes, etc. The authors used CIDDS-001, a public dataset, to assess the proposed approach and achieved results with an accuracy of 0.99. The dataset was split into training and test subset with a 0.6/0.4 ratio.

In (Sood et al., 2014), the authors built ensemble learning by using classifiers such as MLP, SVM, KNN and DT-C4.5, and then combined predictions by using a majority-voting method to obtain the final output. They used NSL-KDD and KDD'99 datasets and extracted 40 features, with a split into three datasets: training, test and validation, and achieved 0.99 accuracy. NSL-KDD was used as the cross-validation method.

In (Soodeh et al., 2019) the authors proposed a novel hybrid framework based on a data stream approach for detecting attacks with incremental learning. They used the naïve Bayes, random forest, decision tree, multilayer perceptron (MLP), and k-nearest neighbors (K-NN) on the proxy side to increase the accuracy of their results. The framework was tested on NSL-KDD and KDCUP'99 datasets, split into train and test subsets with a 0.85/0.15 ratio. The authors achieved a high accuracy (more than 0.95) for each algorithm used.

## 3 A NEW METHOD OF TESTING OF ML MODELS FOR TARGETED DDOS ATTACKS

In this section, we propose a new method of testing ML models with extra tests, which are focused on testing targeted DDoS attacks. The specific steps are described in the next section of the article.

### 3.1 Step 1: Data Processing and Transformation

The first step of the process is to prepare data for a machine learning algorithm. This step includes two

phases: in the first phase, data pre-processing is performed. The second phase, which we introduce for the first time, is the creation of the targeted set of data.

### 3.1.1 Step 1A: Data Pre-processing

The preparation can be defined as a three-stage process: selecting data, pre-processing data, and transforming data. The goal of the selecting data stage is to define what kind of data can be gathered and what the format of this data will be.

If the data is to be gathered, the next stage is to pre-process the data — meaning formatting, cleaning, and sampling the data. The data gathered may not be in a format that is suitable for the model; thus, at this stage, the key is to format the data, remove or fix missing data due to leaks in the data gathering step, and finally to select only the important features.

The last stage of ‘transforming data’ is the process of applying a deterministic mathematical function to the data in order to improve the interpretability and appearance of the data.

### 3.1.2 Step 1B: Generating a Dataset for Targeted Attacks

The main goal of this step is to generate a dataset based on the statistical assumptions made on the similarity of the feature values. This dataset will be used as the traffic of the targeted DDoS attack. The dataset of the targeted attacks is generated according to Formula 1, which is a Cartesian product for a set of predetermined features used in the model.

$$S = S_{f_1}^{SLI} \times \dots \times S_{f_n}^{SLI} := \{(fv_1, \dots, fv_n) : fv_1 \in S_{f_1}^{SLI} \wedge \dots \wedge fv_n \in S_{f_n}^{SLI}\} \quad (1)$$

where:

**S** – the cartesian product

**F** = (f<sub>1</sub>, ..., f<sub>n</sub>) – the set of the features given to the ML model;

**f<sub>1</sub>** – the first feature given to the model (for example IP address);

**f<sub>v<sub>i</sub></sub>** – elements of i-feature, ex. **f<sub>v<sub>1</sub></sub>** are elements for the first feature given to the model;

**n** – the n-th feature given to the model;

**SL** – similarity level, **SL**={0.00 – 1.00};

**I** – the indicator of similarity level: top, least or bottom, **I** = {**T**, **L**, **B**};

**S<sub>f<sub>1</sub></sub>**<sup>SLI</sup> – the new set of the feature f<sub>1</sub> on similarity level equal **SLI**.

It is necessary to define a set of values for each of the features (**S<sub>f<sub>n</sub></sub>**<sup>SLI</sup>) and then to define the degree of similarity (**SL**). The similarity level is defined as a

percentage share of a given value in the main dataset (**MDS**). The similarity may contribute to the most frequent values (**T**), the least frequent values (**L**), or a value in between the two (**B**). The similarity level of the elements in the dataset **MDS** of the feature **f** will be equal to **0.2T**, where all the elements in this new set will be among the 20% most frequent values in the entire main dataset **MDS** for a given feature **f**.

### 3.2 Step 2: Split Data

If the data is already transformed, the next step is to split the data into two training and test subsets or a training, validation and test subset. The first approach can be used in every algorithm and method. The second approach can be used in neural networks to perform additional test to figure out how a neural network is fitted to data. The majority of authors use the term ‘validation data,’ which is interchangeable with ‘test set’ incorrectly. In some papers, the authors split data from the training dataset into 3 parts with 0.6/0.2/0.2 ratios.

### 3.3 Step 3: Train Model

Once the data is split, the machine learning model can be defined and implemented. The process of fitting the model to the data can differ due to the selected type of algorithm. The most common algorithms used in DDoS detection problems are classifiers: support vector machines (SVM), K-nearest neighbors (KNN), Decision Trees, Random Forest and neural networks. To use these kinds of algorithms, some of the parameters need to be defined (ex., batch size, epochs number, number of layers and neurons).

**TRAINING RESULTS.** During the data training, the model returns accuracy for training and validation data, which in turn shows how the model is fitted to the data. Accuracy (**Acc**) is defined as shown in Formula 2. True positives (**TP**) is the number of samples that are correctly predicted as legal traffic; true negatives (**TN**) are samples correctly predicting an attack; condition positive (**P**) is the number of legal cases in the data; condition negative (**N**) is the number of real attack cases in the data.

$$Acc = (TP+TN) / (P+N) \quad (2)$$

where:

**Acc** – accuracy of the model;

**TP** – true positives;

**TN** – true negatives;

**P** – condition positive;

**N** – condition negative.

### 3.4 Step 4: Testing Model

If the model is fitted, it can be tested on the test subset. The model is tested on the last 20% of data from the main dataset (MDS) and for the generated targeted dataset.

#### 3.4.1 Step 4A: Evaluate Model with Test Subset – Testing Results

The test is performed in the same way as during training (Step 3) and then the accuracy is calculated according to formula 2. This way of testing models is easy to develop and achieves an accuracy score near 1.0 – the most common range is between 0.92 and 0.99. However, a model fitted in this way can be easily attacked by a targeted dataset because it can only predict an attack if the attack comes with the same parameters as in the training data. In this case, we can assume that the model is fitted to the specific data and is not able to predict new attack vectors.

#### 3.4.2 Step 4B: Evaluate Model with Targeted DDoS Dataset – Extra Testing Results

In this step, the model is tested according to the extra prepared dataset (Step 1B), which represents the targeted DDoS attack. The test is prepared the same way as in Step 3 and Step 4A, and the accuracy is calculated according to formula 2. In this step, the model is tested according to the targeted DDoS attack.

## 4 CASE STUDIES: THE TESTING ML MODELS WITH TARGETED DDOS ATTACKS

In this section, we would like to present our case studies for testing ML models using the proposed testing methods with targeted DDoS attacks. Each case study was tested by 5 different targeted DDoS attacks, which are represented by 5 different datasets. The datasets were generated according to formula 1 and are presented in Tab. 1. The datasets from 1-4 were created based on the data, which was used for model training (Step 3). Dataset number 5 was generated from the data, which was taken from publicly available data. Dataset number 5 can be created without any knowledge from the training dataset.

Table 1: The parameters of the targeted DDoS attack datasets.

Datasets – Similarity Level	Features
<b>Dataset 1: SLI = 0.2T for F<sub>1</sub></b> (Top 20% from training data)	F <sub>1</sub> = (f <sub>1</sub> , f <sub>2</sub> , f <sub>3</sub> , f <sub>4</sub> ) f <sub>1</sub> – source IP f <sub>2</sub> – source port f <sub>3</sub> – destination port f <sub>4</sub> – header length
<b>Dataset 2: SLI = 0.5T for F<sub>1</sub></b> (Top 50% from training data)	
<b>Dataset 3: SLI = 0.8T for F<sub>1</sub></b> (Top 80% from training data)	
<b>Dataset 4: SLI = 0.2L for F<sub>1</sub></b> (Last 20% from training data)	
<b>Dataset 5: SLI = 1T for F<sub>2</sub></b> (Top 100% from most popular public data)	F <sub>2</sub> – (f <sub>1</sub> , f <sub>2</sub> , f <sub>3</sub> ) f <sub>1</sub> – source IP f <sub>2</sub> – source port f <sub>3</sub> – destination port f <sub>4</sub> – header length = constant

### 4.1 UDP Targeted DDoS Attack using Single Packets

In the first case study, we wanted to test the model for UDP targeted DDoS attack. In (Saied et al., 2015), the authors used a simple multi-layer perceptron classifier for their single packet selection approach to DDoS detection. In this section, we tested the model according to the proposed method.

**STEP 1A.** In the case study presented, we used a dataset that was captured in the network infrastructure of Maria Curie-Skłodowska University in Lublin, Poland. We pre-processed this dataset by separating data using a protocol filter – only UDP packets were selected and four features extracted: IP source address, source port, destination port, and header length of the packet. Once the data was prepared, transformation was performed only on IP addresses, transforming four octets into one number.

**STEP 1B.** In this step, we generate five datasets as described in section 4.1.

**STEP 2.** The dataset from STEP 1A was split into three subsets: train, validation, and test with a 0.6/0.2/0.2 ratio.

**STEP 3.** Based on the model parameters from the (Saied et al., 2015) article, we built a multi-layer perceptron. The aim of building the model is to have the availability of classifying the packets as illegal or legal. Hence, we defined two hidden neural layers with 4 and 3 neurons and one output layer with a

single neuron. The results of the training models are presented in Tab. 2 (**TRAINING RESULTS**)

**STEP 4A.** Once the model was fitted, we performed the cross-validation on the test subset. The test results are presented in Tab. 3. One can notice that our model has an accuracy equal to 0.97.

Table 2: The training results for the MLP classifier.

Metric	Accuracy
Accuracy on the train data	0.93
Loss on the train data	0.20
Accuracy on the validation data	0.97
Loss on the validation data	0.12

Table 3: The test results for the MLP classifier.

Metric	Accuracy
Accuracy on the test data	0.97
Loss on the test data	0.12

**STEP 4B.** In this step, we tested the model for a targeted attack. The results are presented in Tab. 4. One can notice that the highest accuracy is equal to only 0.2 for dataset 5, which was created without any knowledge about the training dataset. If we create the targeted dataset with knowledge about training datasets (1-4), then the highest accuracy is equal to 0.07.

Table 4: The extra test results for the MLP classifier.

Dataset – Similarity Level	Accuracy
<b>Dataset 1: SLI = 0.2T for F<sub>1</sub></b> (Top 20% from training data)	0.00
<b>Dataset 2: SLI = 0.5T for F<sub>1</sub></b> (Top 50% from training data)	0.05
<b>Dataset 3: SLI = 0.8T for F<sub>1</sub></b> (Top 80% from training data)	0.04
<b>Dataset 4: SLI = 0.2L for F<sub>1</sub></b> (Last 20% from training data)	0.07
<b>Dataset 5: SLI = 1T for F<sub>2</sub></b> (Top 100% from most popular public data)	0.20

## 4.2 UDP Targeted DDoS Attack using Time Series

In this approach we decide to use the similar neural network architecture as in section 4.1. and use part of the features described in these publications. The

packets are classified as time series for attack and legal traffic. In this section, we would like to test the presented model according to the targeted DDoS attacks generated based on the proposed method.

**STEP 1A.** In our case study, we used the public dataset FGRP\_SSDP DDoS Attack (Dataset FGRP\_SSDP DDoS Attack, 2020). We pre-processed this argus dataset by extracting features into a csv file: IP source address, source port, destination port, number of packets in flow, total packet size in bytes and flow duration. Once the data was prepared, the transformation was performed only on the IP addresses, transforming four octets into one number.

**STEP 1B.** In this step we generate five datasets, as described in section 4.1.

**STEP 2.** The dataset from STEP 1A was split into three subsets: train, validation, and test with a 0.6/0.2/0.2 ratio.

**STEP 3.** Based on the model parameters from section 4.1, we built a multi-layer perceptron. The model configuration used in this case study is the same as in the previous section. The results are presented in Tab. 5. The accuracy is very high.

Table 5: The training results for the MLP classifier for case study 2.

Metric	Accuracy
Accuracy on the train data	0.97
Loss on the train data	0.09
Accuracy on the validation data	0.99
Loss on the validation data	0.01

**STEP 4A.** Once the model was fitted, we performed the cross-validation on the test dataset. The result of this step is presented in Tab. 6. The accuracy for the test is very high and is equal to more than 0.99.

Table 6: The test results for the MLP classifier for case study 2.

Metric	Results
Accuracy on the test data	0.99
Loss on the test data	0.01

**STEP 4B.** In this step, we tested the model for a targeted attack. Similar to case study number one, we used the five datasets described in Tab.1. The results are presented in Tab. 7. One can notice that the

highest accuracy is equal to 0.50 for dataset 1. In this case, we need to have information about the 0.20 of the most popular traffic in the tested network. If we have knowledge about 0.50 of the traffic, then the accuracy is equal to 0.16. For dataset number 3 (knowledge about 0.80 of traffic), the accuracy is equal only to 0.05. If we create the targeted dataset without any knowledge about the training datasets, then the accuracy is equal to 0.41.

Table 7: The extra test results for the MLP classifier for case study 2.

Dataset – Similarity Level	Accuracy
<b>Dataset 1: SLI = 0.2T for F<sub>1</sub></b> (Top 20% from training data)	0.50
<b>Dataset 2: SLI = 0.5T for F<sub>1</sub></b> (Top 50% from training data)	0.16
<b>Dataset 3: SLI = 0.8T for F<sub>1</sub></b> (Top 80% from training data)	0.05
<b>Dataset 4: SLI = 0.2L for F<sub>1</sub></b> (Last 20% from training data)	0.02
<b>Dataset 5: SLI = 1T for F<sub>2</sub></b> (Top 100% from most popular public data)	0.41

## 5 CONCLUSIONS

The machine learning methods used for the detection and mitigation of DDoS attacks are very effective, especially for unknown attacks. Many models exist in the literature, which have very high accuracies, according to the tests based on the datasets split into train and test or train, validation and test subsets. In this article, we performed targeted UDP DDoS attacks on machine learning models based on single packets and time series. We have shown that models with very high accuracy (0.97 and 0.99) in standard tests are not resistant to a targeted DDoS attack. The prepared tests require different levels of knowledge about the traffic, and one of the levels assumes that the attacker has no knowledge about the network. For ML models, which analyze single packets, the accuracy for targeted attacks is equal to a maximum of only 0.20. In accuracy for ML models, which analyze traffic as the time series, the accuracy for targeted attacks is a maximum equal to 0.50. In our article, we have proposed a new method of testing ML models for targeted DDoS attacks. We have created the algorithm for generating a targeted DDoS attack, which assumes different knowledge levels about the tested traffic. In this article, we would like

to show that it is important to extend the testing of the machine learning.

## REFERENCES

Bouyeddou, B., Kadri, B., Harrou, F., Sun, Y.: DDOS-attacks detection using an efficient measurement-based statistical mechanism. In: Engineering Science and Technology, an International Journal, Volume 23, Issue 4 (2020).

ENISA Threat Landscape 2020 - Distributed denial of service. <https://www.enisa.europa.eu/publications/enisa-threat-landscape-2020-distributed-denial-of-service>

Hodo, E., Bellekens, X., Hamilton, A., Tachtatzis, C., Atkin, R. – son: Shallow and deep networks intrusion detection system: A taxonomy and survey,” In: arXiv preprint arXiv:1701.02145 (2017).

Braga, R., Mota, E., & Passito, A.: Lightweight DDoS Flooding Attack Detection Using NOX/OpenFlow. In: 35th Annual IEEE Conference on Local Computer Networks. Denver, Colorado (2010).

Idhammad, M., Adfel, K., & Belouch, M.: Detection System of HTTP DDoS Attacks in a Cloud Environment Based on Information Theoretic Entropy and Random Forest. In: Security and Communication Networks, Volume 2018.

Pei, J., Chen, Y., & Ji, W.: A DDoS Attack Detection Method Based on Machine Learning. In: Journal of Physics, Conference Series 1237 032040 (2019).

Santos, R., Souza, D., Santo, W., Ribeiro, A., Moreno, E.: Machine learning algorithms to detect DDoS attacks in SDN. In: Concurrency Computat Pract Exper. 2019; e5402. John Wiley & Sons, Ltd. (2019).

Sood, A., Enbody, R.: Targeted Cyber Attacks. In: Syngress (2014).

Peraković, D., Periša, M., Cvitić, I., & Husnjak, S.: Model for Detection and Classification of DDoS Traffic Based on Artificial Neural Network. In: Telfor Journal, Vol. 9, No. 1 (2017).

Saied, A., Overill, R. E., & Radzik, T.: Detection of known and unknown DDoS attacks using Artificial Neural Networks. In: Elsevier B.V (2015).

Soodeh, H., Mehrdad, A.: The hybrid technique for DDoS detection with supervised learning algorithms. In: Elsevier B.V (2019).

Dataset FGRP\_SSDP DDoS Attack. University of Southern California-Information Sciences Institute (2020).