ADIDoS: Automatic Detection and Identification of DDoS Attacks

Pre-print planned to be submitted to ICCST2020[1] Source Code Available[2] Project Demo Available[3]

Omar Mossad
Simon Fraser University
Burnaby, Canada
omossad@sfu.ca

Parham Yassini
Simon Fraser University
Burnaby, Canada
pyassini@sfu.ca

***Abstract*—The Distributed Denial of Service (DDoS) attacks, overwhelm servers and network resources with malicious traffic and render them unusable. Despite extensive research efforts, the last two years have witnessed an increase in the severity of these attacks. Recently, a large number of datasets were created to mimic the DDoS packets and their network characteristics in order to help researchers identify these attacks. Various statistical and analytical models were developed to form a line of defence that can be implemented in Intrusion Detection Systems (IDS). In this paper, we propose a Machine Learning (ML) model that can identify DDoS packets among ordinary network traffic. Our evaluations on the CIC-DDoS-2019 dataset demonstrate that our ML model outperforms the recently proposed classifiers built on Decision Trees, Naïve Bayesian and polynomial regression. The f1-score for our model is 75% compared to 69% for the state-of-the-art decision tree classifier. Moreover, we extend the model task to classify the DDoS attacks and achieve an accuracy of 62%.**

***Index Terms*—DDoS, Machine Learning, Network Intrusion Detection, Network Security**

I. Introduction

Distributed denial of service (DDoS) attacks have been a serious threat to the availability of services across the networks for a long time. Even with the current deployed mitigation techniques, the intensity of these attacks has increased in recent years according to the Worldwide Infrastructure Security Report [1]. Rapid rise in the use of the vulnerable IoT (Internet of Things) devices also poses an emerging threat in this context. IoT devices are predicted to have 14.6 billion machine-to-machine connections by the end of 2022 [2] and considering the distributed nature of such devices, their aggregated bandwidth can be used to perform devastating DDos attacks. Mirai, Persirai and Satori [3], are some of the recent examples of botnets that have successfully attacked internet services by exploiting IoT devices' vulnerabilities. The above-mentioned facts, motivate the development of methods for timely detection of DDoS attacks in order to mitigate the cost and damage.

[1] https://easychair.org/cfp/IEEE-ICCST2020
[2] https://github.com/omossad/cmpt980.git
[3] https://youtu.be/8QBZlXfdA4w

There are two main types of approaches for detecting such attacks in the network intrusion detection systems (NIDS): the packet-based and flow-based detection. Packet-based approaches require deep packet inspection and check the payload and header information of individual packets. These methods suffer from performance problems for real-time attack detection in high speed networks. On the other hand, the flow-based methods use only information about a collection of packets belonging to a flow for detection [4].

In this paper, we aim to solve the problem of identifying DDoS attacks based on the traffic flow traces at the victim network using a machine learning model. The input of the proposed model is the flow features (e.g number of packets, port number, protocol and etc.) and the output will be the class of the flow whether it belongs to benign or malicious traffic.

The contributions of this paper can be summarized as:

- Analysis of the DDoS network traffic to select the best set of features for detection.
- Develop a machine learning model capable of identifying DDoS packets among benign traffic.
- Extend the model to categorize the types of DDoS attacks.
- Evaluation and analysis of the model performance on the CIC-DDoS-2019 dataset.

The remainder of the paper is organized as follows: Section I provides an introduction to DDoS attacks. Next, a survey on DDoS detection algorithms and datasets is given in section II. Next, we focus on the attack scenarios and defense strategy based on the 2019 DDoS dataset and the characteristics of the data is highlighted in section III-A. Section IV describes our proposed DDoS detection model. Detailed evaluations for our system compared to the state-of-the-art are demonstrated in section V. Finally, we conclude our work and suggest future improvements in section VI.

II. Related Work

Over the past decade, detection and prevention of denial of service attacks have been widely studied. The earlier systems employed attack signature matching and threshold-based detection methods which are good at detecting known attacks but new attacks with slight disparity can easily bypass these

signature based methods [5]. Also, another common approach is anomaly detection, where the traffic traces are compared to a baseline of "benign" traffic behaviour to detect the attacks. These methods have shown to have a better performance in detecting new attack signatures [6] but they fail to adopt to the constantly changing nature of the benign traffic in networks. This can cause high false positive rate and it would block legitimate users from accessing the services.

Recent studies have explored applying machine learning techniques for the DDoS detection. The machine learning algorithms used by these systems include Random Forests [7], Support Vector Machines, Neural Networks [8], [9] and etc.. Studies focusing on machine learning based detection methods can be divided into two major types: Anomaly-based detection and classification methods. In anomaly based detection approaches such as [10], the model is trained using the normal traffic traces only and in the test phase, the deviation from the normal traffic is used for detecting the attacks. In the classification methods such as [11], labeled data for attack and benign traffic is used for training the machine learning model.

Also, there are a few recent studies addressing the special cases of DDoS attacks that are lunched from IoT devices using machine learning [12], [13]. In both studies, they have used a flow based approach considering the special characteristics of the IoT traffic, and proposed methods for detecting DDoS near the source of the attacks on edge devices. Their results show promising accuracy for detecting the attacks. However, the model training was done using very limited data and a narrow range of attacks can be detected using their methods. Our approach can be used in a wider range of attack scenarios and can be deployed near the destination of the attack where more computational resources are available.

Sharafaldin et al. [7] have created a dataset that mimic a large variety of DDoS attacks. Additionally, they used the flow-based network metrics (e.g. number of ACKs, packet sizes, etc.) as features to classify the traffic into normal and attack classes. They have experimented with different classification methods based on Random Forests, Naïve Bayes and logistic regression algorithms. In this work, we use the dataset created by [7] to build a deep neural network that can extract the features and classify the attack types. Our claim is that deep neural networks proved to be better than conventional machine learning techniques for classification tasks. Also, we extend the task of detecting DDoS attack to the identification of the attack type as labelled in the dataset. As far as we know , none of the works in the literature have studied the classification of DDoS attack types.

III. Attack Model and Detection Strategy

A. DDoS Attacks

The DoS attackers attempt to stop users from accessing a specific service or a network resource by targeting different layers of a network. Based on the taxonomy provided by [7], DDoS attacks can be classified as two main types: Exploitation attacks and Reflection attacks. In the former type, the attacker would use the vulnerable systems distributed across the networks to send a large amount of UDP or TCP packets to the victim network. In the latter case, the requests are not directly sent to the victim network but instead, spoofed requests are sent from the attacker to some third party service and trigger that server to send large amount of responses to the victim machines.

Figure 1 demonstrates the attack scenario for the reflection based attacks. The vulnerable hijacked systems are often called "zombies" and the attack is initiated by these devices. The spoofed source IP makes the packets appear as if they were sent from the victim server, therefore the destination of the response packets will be the victim machines.

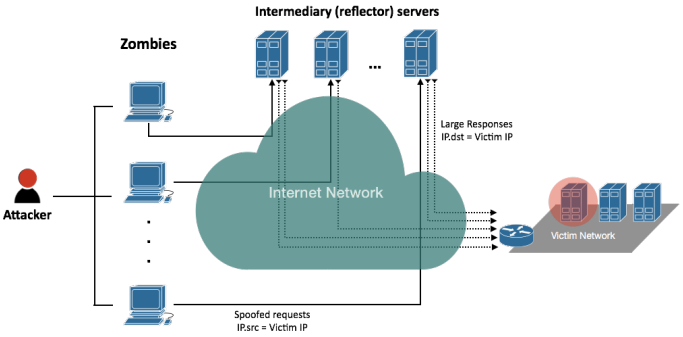

Fig. 1. Attack model for the reflection DDoS attacks

Detecting the reflection attacks, is a quite challenging task since the attackers can send the previously recorded *"benign"* traffic (e.g legitimate response from a DNS server) for this type of attack [14].

B. Dataset

The CIC-DDoS-2019 dataset [7] includes network captures done over 2 days. The captures done on January 12^{th} are considered as the training data, whereas those collected on March 11^{th} are considered as testing data.

The captured raw packets are available in .PCAP files and were subsequently converted into Comma Separated Values (CSV) files during the features extraction process using CICFlowMeter-V3. The CSV files record several network characteristics for each packet. These network properties of the captured traffic flows including but not limited to the *Flow Duration*, *Total Forward/Backward Packets*, *Length of Forward/Backward Packets*, etc. are considered as the input features for the classifiers.

The packets were labeled as either benign or by the attack name. There were 13 different DDoS attack types: *PortMap*, *NetBIOS*, *LDAP*, *MSSQL*, *UDP*, *UDP-Lag*, *SYN*, *NTP*, *DNS*, *SNMP*, *SSDP*, *WebDDoS* and *TFTP*. Table I summarizes the differences among these attacks. The first three rows in the table are among the exploitation DDoS attacks and the rest of the attacks are among the reflection based attacks. Although some DDoS attack types don't appear in both sets, we expect

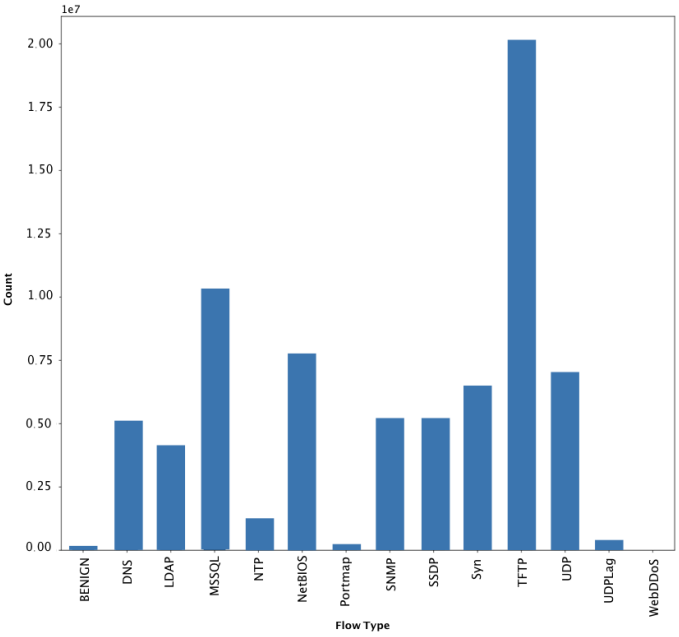

Fig. 2. Distribution of the flow types in the dataset

most models to adapt and intuitively identify these attacks. Figure 2 illustrates the total number of flows (rows in the CSV files) that belong to each specific attack type in the dataset.

C. Detection and Defense Strategy

The proposed defense mechanism is based on the traffic traces captured at the victim network gateway. A setup similar to the one that was used for collecting the data can be used in the deployment phase. A capture server inside the target network would capture all of the incoming and outgoing traffic. The flow features can be extracted from these traces in real time using the CIC Flowmeter or similar tools such as FlowScan [15]. Once the model is trained it can be deployed for detection at a capture server given the flow features. Subsequently, it can detect the malicious flows and the output of the model can be used to take proper policies against the source of the attack which is out of the scope of this paper. In addition, with the proposed attack type classification model described in the next section, it is possible to report the type of attack (e.g NTP reflection or DNS reflection) in order to choose more fine grained policies against a specific type of attack without disturbing other services on the network. Considering Network Function Virtualization (NFV) technology, intrusion detection can be done using a commodity hardware that is also providing the network with different functions such as Firewall, IDS and etc. [16]. In our proposed approach, the model can be deployed in any of these servers which can capture the network traffic and the output of the model can be used for decision making to block certain flows or limit the flow rate.

IV. Proposed Approach

In this section we describe the feature selection procedure and the architecture of the two deep neural networks developed for detection and identification of DDoS attacks. Both proposed models have similar structure but vary in the output layers.

A. Features Exploration

In order to improve the inference time and performance of such models, we undertook a feature exploration phase where we evaluated the effects of the features on the classification task in order to reduce the dimensionality of the input data. This step is done once during the training phase which is not timely constrained and it can reduce the inference time for every new observation during the deployment phase where the detection time is critical. To evaluate the most effective features for classification, we employed a recursive feature elimination method with cross validation as implemented in the *scikit-learn* library [17]. First a decision tree classifier is trained on the data using all of the features and recursively, a smaller subset of features is used for classification by eliminating the less important features. The feature importance is compared by the coefficient of the features in the obtained decision function.

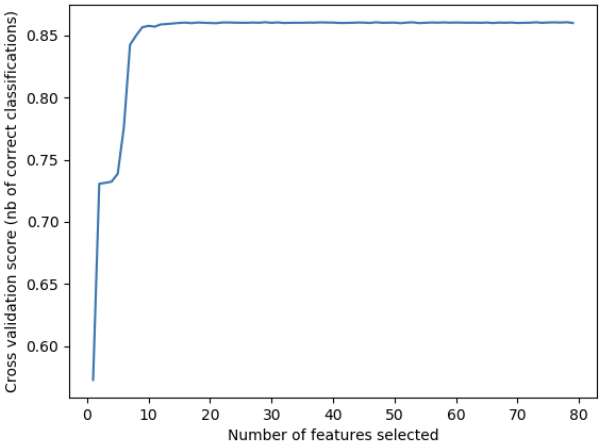

Fig. 3. Recursive feature elimination, finding optimal number of input features

Figure 3 illustrates the accuracy of the classifier versus the number of features used. We have used 28 of the most influential features out of the 87 features based on the results of this stage. The other features contain redundant information that is already covered by the features we have already used as the classifier input.

B. Binary Classification Model

The binary classification model incorporates 3 hidden layers each of size 64. For each packet, the 28 features previously described are fed to the input layer after pre-processing. We used a quantile transformer to normalize the features values. All the network layers use ReLU activation functions, whereas the output layer has a Softmax classifier. Figure 4 depicts the overall network architecture. For the hyper-parameters, we used 10 epochs and a categorical cross-entropy optimizer.

TABLE I
DDOS ATTACK TYPES

Attack Type	Description
UDP Flood	This attack is initiated on the remote hosts by sending a large number of UDP packets to the victim server
TCP SYN Flood	SYN flood is a type of TCP State-Exhaustion where the attacker creates half-open connections
UDP Lag	This attack is initiated on the remote host by sending a large number of UDP packets
PortMap	Takes advantage of RPC Portmapper to overwhelm networking services
NetBIOS	Openly accessible NetBIOS name services can be abused for DDoS reflection attacks against third parties
LDAP	Attackers abuse exposed Lightweight Directory Access Protocol servers
MSSQL	Attacks an old exploit in Microsoft SQL servers
SYN	SYN flood is a type of TCP State-Exhaustion where the attacker creates half-open connections
NTP	Attacker exploits publicly-accessible Network Time Protocol servers to overwhelm them with UDP traffic
DNS	Attacker floods a particular domain name server in an attempt to disrupt DNS services
SNMP	Simple Network Management Protocol Reflection involves eliciting a flood of responses to a single spoofed IP address
SSDP	Simple Service Discovery Protocol attack is a reflection-based DDoS attack that exploits Universal Plug and Play
WebDDoS	DDoS attack on web servers
TFTP	Amplification attack based on the Trivial File Transfer Protocol

Cross-validation was applied using 10% of the training set to prevent overfitting and add enough regularization to the model in order to adapt to unknown attack types.

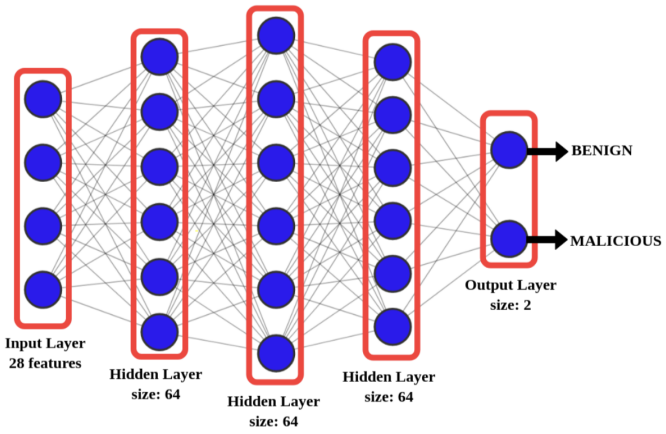

Fig. 4. Network Architecture for the binary classification model: The 28 features input is fed through 3 hidden layers using ReLU activations towards the output layer where the Softmax activation is used to determine whether the traffic was benign or malicious

C. Category Classification Model

Similar to the previous model, the categorical classifier differs mainly in the output layer. The Softmax in this case has 13 outputs representing the different DDoS attack types. Another difference is that in case of the categorical model, we chose to discard the benign traffic from the input data to make the data evenly distributed.

V. EVALUATIONS

Next, we evaluate the performance of both models and compare them against the recent literature.

A. Binary Classification Model

To evaluate the binary classification model we have to consider the True/False Positives and True/False Negatives. The importance of keeping the false positive rate low is emphasized in previous research [18]. In DDoS attacks, it is not essential to filter out all the malicious packets. However, the wrongfully labeled benign packets (i.e. the false negatives) need to be minimized in order not to affect the legitimate user traffic. In addition, each false positive requires wasting the analyst time for examining the reported incident. We showcase these values in Table II. The ratio between the True Negatives and False Negatives is quite reasonable where almost 0.1% of the benign packets were mis-labeled.

TABLE II
BINARY CLASSIFICATION DETAILED RESULTS

		Actual Class	
		BENIGN	***MALICIOUS***
Predicted Class	***BENIGN***	20,305,022 (TN)	2538 (FP)
	MALICIOUS	20,747 (FN)	36,218 (TP)

Then, we compare the overall results of the model against the classifiers developed in [7]. There were a total of 4 classifiers, namely ID3 (Decision Tree), Random Forest, Naïve Bayes and Logistic Regression. We begin by describing the evaluation metrics used:

$$Precision = \frac{True\ Positive + False\ Positive}{True\ Positive}$$

$$Recall = \frac{True\ Positive + False\ Negative}{True\ Positive}$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Table III demonstrates a considerable improvement in the f1-score compared to the best classifer (i.e. the ID3 Decision Tree). Given the enormous size of the dataset, this 7% increase is quite significant especially if applied in daily network traffic.

We have managed to reproduce the reference paper work but had to restrict the number of lines read to 1M lines only per file due to memory limitations. We chose to omit the empirical

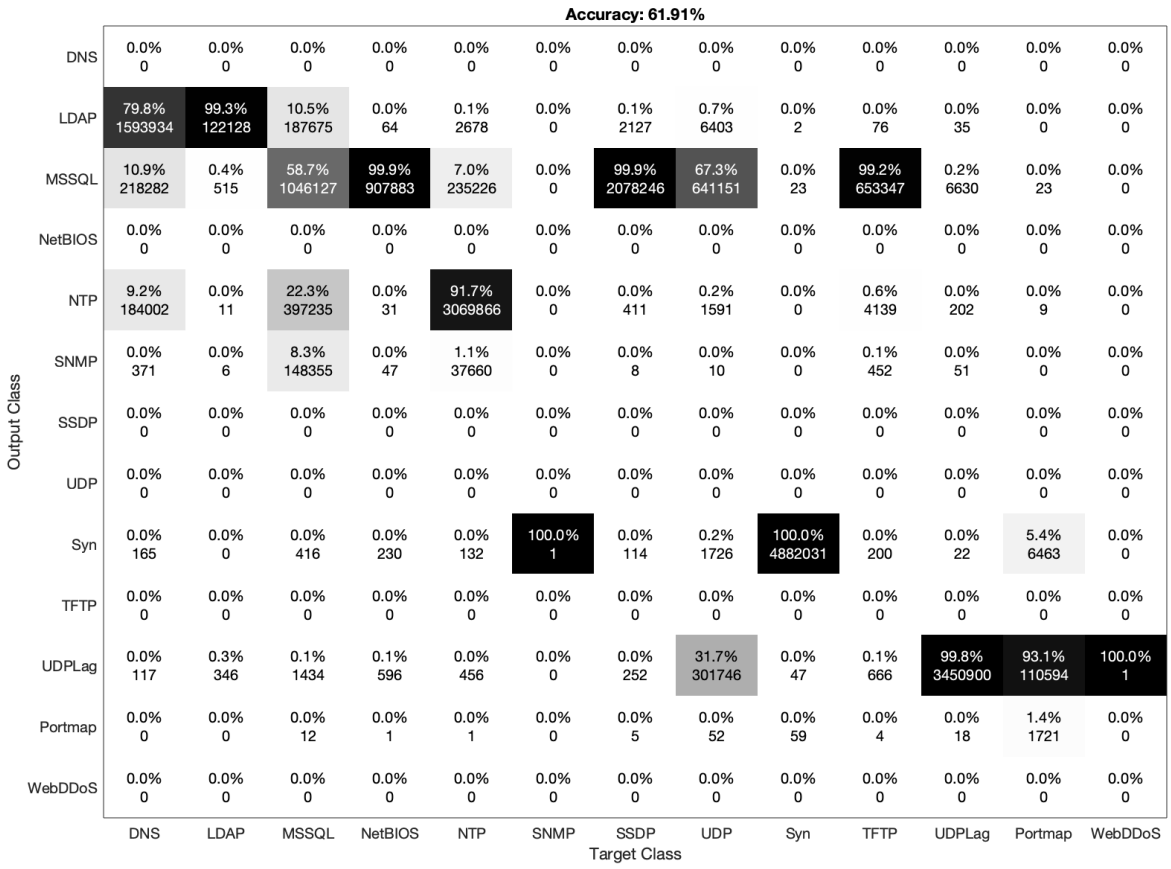

Fig. 5. Confusion Matrix for the category classification model on the CIC-DDoS-2019 dataset

TABLE III
PRECISION, RECALL AND F1-SCORE EVALUATIONS

Algorithm	Precision	Recall	F-1 score
ID3 [7]	0.78	**0.65**	0.69
RF [7]	0.77	0.56	0.62
Naïve Bayes [7]	0.41	0.11	10.0
Logistic Regression [7]	0.25	0.05	0.02
Our Algorithm	**0.93**	0.64	**0.76**

results on the 1M per file runs as they are not mitigating the actual model performance as cited in the original paper. However, we have included the necessary codes to reproduces these results in the repository.

For the training and test phases, we run our experiments on an *NVIDIA V100 PCIE 32GB* GPU. In the test phase, the average inference time of the model is 54 µs for each flow while the average flow duration for the malicious flows were around 73s. This confirms the timeliness of the attack detection in our proposed approach.

B. Category Classification Model

To our knowledge, there isn't any existing work that aim to classify the DDoS attack types. Therefore, we have decided to include the results in the form of accuracy and confusion matrix for the various DDoS classes. As shown in the Figure 2, the dataset is imbalanced in terms of attack types and our results for this task have been adversely impacted by this factor. However, the model was able to achieve an accuracy of 62% and the confusion matrix in Fig. 5 provides a detailed analysis of the predicted class labels vs the actual ones.

Due to the time limitation, we chose to include the results as is, but we plan to fine-tune this model by varying the hyper-parameters and using other set of features. Additionally, we may implement a hierarchical approach where we further categorize DDoS attacks into sub-categories.

VI. CONCLUSIONS AND FUTURE WORK

In conclusion, we built a ML model capable of identifying the DDoS attacks based on the packets characteristics. Currently, the precision and f1-score outperforms the state-of-the-art models with a significant margin. An extension was

also made to classify the DDoS attack types. Our experiments also confirms the timeliness of the attack detection in our proposed light weight model. As a future work, we plan to fine-tune the categorical model to increase the current classification accuracy. A more refined feature selection will be established to limit the inference time without affecting the model performance. We also plan to build an end-to-end DDoS detection system and test it in a realistic network scenario.

VII. Lessons Learnt

During this project we had the chance to learn more about the already deployed techniques for network intrusion detection. Also, we gained a deep understanding about different types of DDoS attacks and how they are launched using different protocols such as NTP, DNS and etc.. During the implementation phase, we got familiar with the tools for collecting flow data from a set of packets (e.g CICFlowmeter and NetFlow). More importantly, we gained hands on experience in developing deep learning classification models and exploring different features of the network traffic flow.

Acknowledgment

We would like to acknowledge the efforts and guidance provided by the CMPT980 course instructors. By organizing the project deliverables in the form of regular milestones, it has significantly helped to achieve a satisfactory state at the end.

References

[1] "Netscout's 14th annual worldwide infrastructure security report," https://www.netscout.com/report/, accessed: 2020-03-13.

[2] "Cisco visual networking index: Forecast and trends, 2017–2022 white paper," https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html, accessed: 2020-01-30.

[3] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "Ddos in the iot: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[4] P. Gogoi, M. H. Bhuyan, D. Bhattacharyya, and J. K. Kalita, "Packet and flow based network intrusion dataset," in *International Conference on Contemporary Computing*. Springer, 2012, pp. 322–334.

[5] J. Mirkovic and P. Reiher, "A taxonomy of ddos attack and ddos defense mechanisms," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, pp. 39–53, 2004.

[6] J. Andress, *The basics of information security: understanding the fundamentals of InfoSec in theory and practice*. Syngress, 2014, ch. 10, pp. 151–169.

[7] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *2019 International Carnahan Conference on Security Technology (ICCST)*. IEEE, oct 2019. [Online]. Available: https://doi.org/10.1109%2Fccst.2019.8888419

[8] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep learning approach for network intrusion detection in software defined networking," in *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*. IEEE, 2016, pp. 258–263.

[9] X. Yuan, C. Li, and X. Li, "Deepdefense: identifying ddos attack via deep learning," in *2017 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2017, pp. 1–8.

[10] R. Abdulhammed, M. Faezipour, A. Abuzneid, and A. AbuMallouh, "Deep and machine learning approaches for anomaly-based intrusion detection of imbalanced network traffic," *IEEE Sensors Letters*, vol. 3, no. 1, pp. 1–4, 2019.

[11] M. Aamir and S. M. A. Zaidi, "Clustering based semi-supervised machine learning for ddos attack classification," *Journal of King Saud University-Computer and Information Sciences*, 2019.

[12] K. Bhardwaj, J. C. Miranda, and A. Gavrilovska, "Towards iot-ddos prevention using edge computing," in *USENIX Workshop on Hot Topics in Edge Computing (HotEdge 18)*. Boston, MA: USENIX Association, Jul. 2018. [Online]. Available: https://www.usenix.org/conference/hotedge18/presentation/bhardwaj

[13] R. Doshi, N. Apthorpe, and N. Feamster, "Machine learning ddos detection for consumer internet of things devices," in *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2018, pp. 29–35.

[14] A. G. Bardas, L. Zomlot, S. C. Sundaramurthy, X. Ou, S. R. Rajagopalan, and M. R. Eisenbarth, "Classification of UDP traffic for ddos detection," in *Presented as part of the 5th USENIX Workshop on Large-Scale Exploits and Emergent Threats*. San Jose, CA: USENIX, 2012. [Online]. Available: https://www.usenix.org/conference/leet12/classification-udp-traffic-ddos-detection

[15] D. Plonka, "Flowscan: A network traffic flow reporting and visualization tool." in *LISA*, 2000, pp. 305–317.

[16] S. K. Fayaz, Y. Tobioka, V. Sekar, and M. Bailey, "Bohatei: Flexible and elastic ddos defense," in *24th {USENIX} Security Symposium ({USENIX} Security 15)*, 2015, pp. 817–832.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[18] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *2010 IEEE Symposium on Security and Privacy*, 2010, pp. 305–316.