A Machine Learning Model for Classifying Unsolicited IoT Devices by Observing Network Telescopes

Farooq Shaikh[1], Elias Bou-Harb[2], Jorge Crichigno[3], and Nasir Ghani[1]

[1]University of South Florida, USA
[2]Cyber Threat Intelligence Laboratory, Florida Atlantic University, USA
[3]University of South Carolina, USA

Abstract—**The Internet of Things [IoT] promises to revolutionize the way we interact with our surroundings. Smart cars, smart cities, smart homes are now being realized with the help of various embedded devices that operate with little to no human interaction. However these embedded devices bring forth a plethora of security challenges as most manufacturers still assign higher importance to the three Ps (prototyping, production and performance) than security. This inherent flaw has manifested itself in the form of various Denial of Service (DoS) attacks orchestrated with the help of unsolicited IoT devices on the Internet. We are even seeing massive throughputs without the need for amplifications affecting large scale infrastructures on the Internet. Thus, understanding the nature of these attacks and quickly identifying infected devices becomes imperative to combat this situation. In this paper we present a model to classify unsolicited IoT devices in enterprises using machine learning (ML). Namely IP header information from darknet data is collected for analysis. We then consider multiple supervised ML algorithms to classify these Layer 3 headers. We evaluate these algorithms and compare their performances in terms of accurately identifying activities of malicious IoT devices on the Internet. Our results show that Random Forest and Gradient Boosting have high recall and precision scores whereas NaiveBayes has the worst performance. We believe our model can be used by enterprises as a part of their intrusion detection system to quickly identify infected IoT devices within their own environment as well as identify scanning activities directed towards them.**

Index Terms—**Network Telescopes, Darknet Analysis, Traffic Characterization, Internet of Things, Machine learning.**

I. Introduction

There has been a massive proliferation of IoT devices in the market over the past few years, and it is predicted that this number will approach almost 30 billion devices by 2020. However market competition and technical limitations prove a hindrance in improving the security of these devices. To make matters worse, more often than not, default usernames and passwords are not changed, making these devices a prime target for adversaries to exploit. This has been shown in recent years with large scale DDoS attack on internet scale infrastructures, e.g., the attacks on Dyn and Liberia. New botnets like Hajime [1] and Reaper [2] also show how adversaries are constantly adapting to avoid detection and remain a constant threat to the ever-increasing landscape of IoT. These IoT botnets also scan the Internet for other devices using a manufacturer's default settings and can quickly grow into a powerful assembly of weapons to cause serious repercussions to multiple stakeholders.

Now Internet sinks or ”darknets” represent the unused addresses with no legitimate traffic directed towards them. An analysis of this Internet background radiation (IBR), consisting of mostly unsolicited traffic, can thus provide key insights into the methodology of attackers and be a stepping stone towards developing efficient models that can identify and notify of incoming scanning or other malicious acts.Hence we use the CAIDA datasets [3], [4] as this facility is available for research purposes. Namely CAIDA provides access to network telescopes which have collected a lot of darknet data.

Overall the first step towards defending against the Internet of vulnerable things is to analyze and classify the network characteristics of malicious devices. This analysis can aid in building models that can be deployed at enterprise and ISP perimeters to detect suspicious activities within the organization or help identify scanning or DoS activity directed towards them. We use machine learning to develop a model to :

- Characterize and label different malicious activities of IoT devices, e.g., scanning and DoS attacks, based on the network characteristics derived from network telescopes
- Train and test ML to classify the IoT devices based on network traffic features to provide close to real time classification of IP addresses
- Evaluate the performance of these algorithms

The rest of the paper proceeds as follows. Section II reviews a number of related works in the context of machine learning for threat detection in IoT and non-IoT environments. Section III then elaborates on the proposed approach and details its rationale and employed techniques. Section IV, then presents the empirical evaluations and analyzes the overall results. Finally, Section V provides concluding remarks and pinpoints several insightful topics to pave the way for future work in this impactful IoT security research area.

II. Related Work

Network traffic classification has been used extensively to classify different applications on the Internet. For example previous work has focused on classifying benign Internet traffic into different classes [5] [6]. The authors in [7] also perform device classification for IoT based on network traffic analysis. However notions of maliciousness are not considered. The authors in [8] present a framework for detection of DDoS backscatter based upon network features of packets found in the darknet. This approach, however, does not take into account any other types of malicious activity present in the darknet and nor does it provide any insight into the unique characteristics present in malicious IoT traffic.

A few other research endeavors have also analyzed ML applications for IoT security. The authors in [9] use ML algorithms to classify malware on the Internet by observing payload data, but this suffers from high operational overload, where defenses have to be constantly updated to the ever changing format of packet payloads. Also this solution does not focus explicitly on IoT data. The authors in [10] use ML to detect suspicious IoT devices connected to the network through the use of a static white lists within the enterprise to define the allowed devices within the network. However this suffers from the major drawback that it can be bypassed by an adversary with access to a white list. Also, the authors do not consider scanning or other malicious activities directed towards the network from outside the enterprise domain.

Meanwhile the authors in [11] use association rule extraction to detect anomalies and identify recurrent packets in the data with features similar to NetFlow. However it fails to provide automatic traffic classification and requires human expertise to analyze the data and draw useful conclusions. Again, no IoT specific provisions are made either. In [12], Hodo et al. present an artificial neural network (ANN) based intrusion detection system but focus on a single type of attack. However this scheme is relatively less efficient in it's computational complexity. ANNs are also used by the authors in [13], where they create an IoT testbed emulating node devices and using a Raspberry Pi as a gateway. However this approach suffers from the lack of sufficient data points. Although the authors introduce invalid data points to define anomalous activity, they provide no concrete explanation of what constitutes an anomaly for the devices used in the testbed.

Although the above-detailed studies address a few concerns when it comes to DoS and anomaly detection in the IoT realm, none of them provide a comprehensive framework, i.e., to proactively identify the major threats to IoT devices in an enterprise environment. In the light of the above, we develop a comprehensive model that uses popular classification algorithms using traffic information from unsolicited IoT devices found in the darknet. The goal here is to help organizations identify both intra-domain and inter-domain anomalous behavior of IoT devices.

III. Proposed Approach

This section details the proposed approach including its aims, employed methods and techniques.

A. Inferring Malicious IoT devices from the Darknet

Lack of empirical data for malicious IoT devices is a key deficiency in many research endeavors (owing to the various privacy and logistic concerns when trying to obtain such information from local IoT realms). This limitation has a profound effect within the context of ML research since it relies on sufficient representation of anomalous behavior to make reliable decisions in real world environments.

To address this concern, we leverage network telescopes to provide insights into Internet-scale maliciousness of IoT devices, i.e., through the use of the Center for Applied Internet Data Analysis' [CAIDA] dataset. The facility operates a data collection, curation and distribution infrastructure and makes it available to the scientific and research community on request. CAIDA's massive measurement archive addresses the issue of the lack of sufficient data for analysis of ML algorithms. The primary reason of selecting IP header information for extracting features is due to the fact that obtaining and processing this information is less time and resource consuming than analyzing the entire payload. Also, IoT devices being used as part of a botnet would exhibit similar network characteristics as they try to scan the Internet for other vulnerable machines.

B. Extracting Feature Vectors from the Dataset:

Overall, ML techniques focus on analyzing data and extracting useful inferences from it. In general a set of data vectors called features serve as inputs to the algorithms and the outputs largely depend upon the type of learning being performed. As a result ML schemes can provide previously unseen insights into the data.

Now selection of the most relevant features is arguably one of the most important steps in any ML design process [14]. For example, irrelevant or redundant features can significantly impact the accuracy of a ML algorithm. Hence network traffic classification research has already focused on identifying flows from raw data and extracting features based upon these flows [15], [16]. Now the darkent, as mentioned previously, has no physical devices associated with the allocated IP addresses. As such, any traffic directed to it receives no response. Therefore, the notion of flows defined for normal Internet traffic classification does not hold true here. Hence, for the purpose of this paper, we consider a flow to be identified by a source IP address and extract features over multiple packets in the given time duration for each. Essentially these feature vectors are indexed by the source IP address. Overall, statistical features in network traffic classification have proved to be very useful

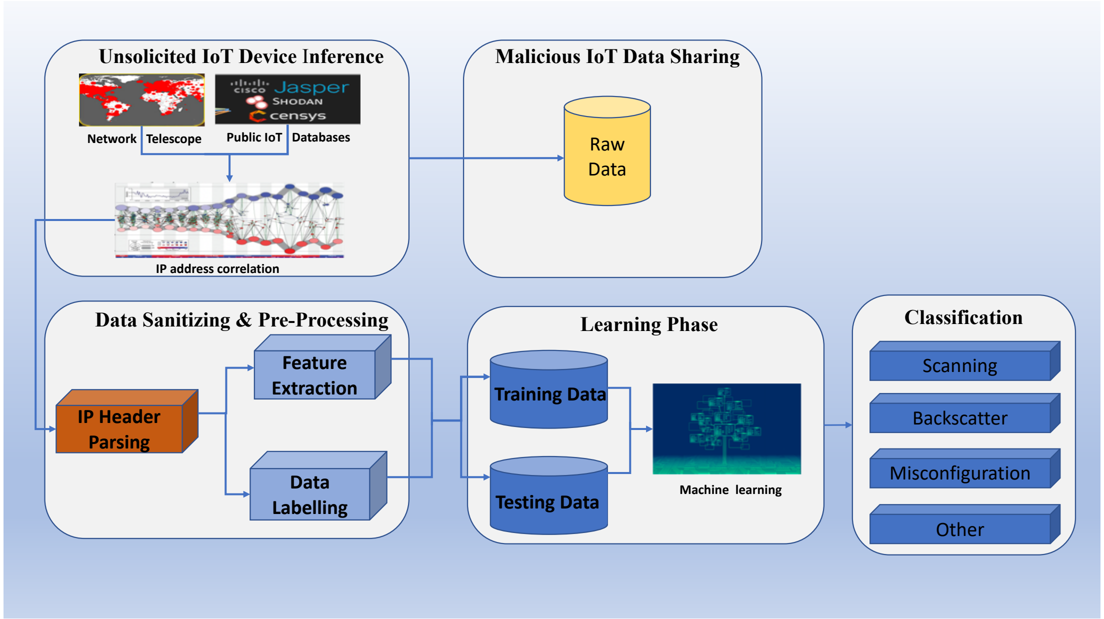

Fig. 1: Proposed methodology for Classfication

in identifying different seen and unseen applications on the Internet including the classification of IoT devices [7], [17]. For example, the authors in [8] and [18] extract features from the darknet data based upon it's statistical characteristics. We therefore, decide to adopt a similar methodology for extracting features from the darknet data. Further details on the selected features are provided in table .

C. Labelling data:

After the data extraction process, we label the data of the IoT devices found in the darknet. As explained previously, darknet data is unique in the sense that it is a one way communication (with no possible response from the unallocated IP address found there). This fact makes it difficult to use certain traditional approaches for labelling data found in these network telescopes. Hence the authors in [19] [20] present an effective scheme for identification of scanning activities in one way traffic domains like the darknet. The authors in [21], [22] also present another unique approach for categorizing data from the darknet. We use this latter approach as a basis for labelling the data we extracted from CAIDA and provide the rationale behind each label in the subsequent sections

D. Scanning

Malicious IoT devices are constantly trying infect other devices on the Internet to increase the size of their botnet and consequently increase the impact of DoS attacks orchestrated using these bots. Various different types of scanning activities can be performed by such devices, including but not limited to network scans, port scans, stealth scans etc [21]. In port scanning the attacker tries to identify the active ports on a host by sending request packets with the goal of exploiting a vulnerability associated with the service running on that port. Therefore if we find that a particular source IP address targets more than 5 ports for every unique destination IP address as well as sending over 50% of it's packets with either the SYN, FIN, FIN-ACK or NULL flags set, we label it as performing port scanning

Meanwhile in network scans the attacker aims to exploit a single service by targeting the same port on multiple hosts. These scans are often used to expand the bot network by recruiting more devices. Hence in this work, we label a source IP address as performing network scanning if the number of destination IPs with the same target port exceeds 5, and as per port scanning, over 50% of packets have similar flag settings [21].

Finally, stealth scans are often used by attackers to evade intrusion prevention/detection systems. These scans involve sending a small amount of TCP or UDP packets in a short amount of time to a small number of hosts. More details on such scanning activities is provided in [21] and [22]. Hence we consider a source IP address as performing stealth scanning if it sends less than 15 SYN packets to less than 15 hosts targeting less than 5 ports in a given time frame. Similarly, if a source IP address sends less than 15 ICMP echo requests packets to less than 5 hosts in a given duration of time, we also consider it as performing stealth scans. The taxonomy performed in [21] includes these activities under the broader umbrella of scanning.

E. Backscatter

IP address spoofing is a common technique applied by adversaries in the context of DDoS attacks. Thus, when the target is flooded with requests, it sends a reply back to these spoofed IP addresses. However, this would result in no replies if the destination IPs reside in the darknet, i.e., there are no

TABLE I: Feature Selection for Malicious IoT Traffic in the Network Telescope at CAIDA

Feature	Description
average ttl	The average value of ttl for each source IP address.
total packet count	The total number of packets sent by each source IP address
number of unique destination IPs	The total number of unique destination IPs targeted by each source IP address
average of packets sent to each port	The average number of packets sent to each destination port by each source IP address.
average packets sent to each IP	The average number of packets sent to every destination IP by each source IP address
inter arrival time	The average of the inter arrival time of packets sent by each source IP
scan flags	The number of packets sent with scan flags[SYN, FIN, FIN-ACK, NULL]
udp packets	The number of UDP packets sent by each source IP address
number of unique destination ports	The total number of unique destination ports targeted by each source IP address

physical devices associated with these IP addresses. Therefore any reply packet seen in the darknet is a clear indication of backscatter traffic from a DDoS attack [8]. Accordingly, we label a source IP address as a victim of a DDoS attack if we see a reply packet from it in the darknet. Any TCP packet having the SYN-ACK, RST, RST-ACK or RST flag set is also identified as a reply packet. Finally any other IP packets seen in the darknet form IoT devices are labelled as misconfiguration.

IV. Empirical Evaluation

Empirical findings from various datasets are now presented. Namely we extracted the IP addresses of about 3 million IoT devices from various databases like Censys, Shodan, etc. These IoT devices belong to different system categories like SCADA, webcams, thermostats, DVRs etc. We then created a database of these IP addresses and compared them to the source IP addresses collected by the network telescope at CAIDA for a one hour period on the 1st day of January 2017. Note that a single hour of data at CAIDA is approximately 85 gigabytes and contains IP header information for every packet found in the traffic. Overall, the vast size of this data makes the task of training and testing our machine learning algorithms a lot simpler. In the past lack of sufficient data has made it difficult to validate the effectiveness of models in other models [7], [13].

Classification of traffic helps operators and service providers in a multitude of domains e.g., quality of service [QoS], monitoring, intrusion detection etc. However, finding the right hypothesis for making a prediction is the crux of the objective of a supervised ML algorithm. Now ensemble learners make this objective easier by combining a number of hypotheses and trying to find the best one, often with the support of multiple weak learners [23]. More specifically, learners like AdaBoost, Gradient Boosting and Random Forest have been tested successfully for network traffic classification. In [24] the authors also use ensemble learners to classify anomalous traffic and develop signatures. Similar to the work in [24], the authors in [25] use Random Forest for traffic classification, however data normalization is required before processing here.

Meanwhile the Naive Bayes scheme fits a Gaussian distribution over the data and assumes independence between the features (albeit this may not always hold true especially in the ever changing landscape of malicious Internet traffic.) However, its ability to work well with complex models and tractability make it an ideal choice for certain classification problems [26]. For example, Amor et al [27] use a Naive Bayes classifier to detect different attack scenarios and do some performance measurement as well.

Overall, we identified the presence of slightly over 2 million IoT IP addresses in just 1 hour on the 1st day of January 2017 from the CAIDA dataset. The IP header information extracted was stored in a database for easier processing. We extracted the features mentioned in Section III-B and labelled them according to the methodology proposed in Section III-C. Based on this, we identified 208,647 IoT IP addresses as performing scanning, 10628 sending backscatter traffic and 72794 IPs were labelled as misconfiguration traffic. Python's scikit library includes a broad range of supervised and unsupervised machine learning algorithms using Scientific Python also has an easy interface to implement them. Based on the various studies mentioned above we used the Random Forest, Gradient Boosting, Ada Boost and Naive Bayes schemes for classifying the darknet data. Namely 70% of the data for training and the remaining for testing. This partitioning was done using a random split function provided by the scikit library and we also used 5 fold cross validation on our dataset. The two metrics we used to judge the efficiency of our algorithm are *recall* and *precision*. *Precision* is defined in ML literature as the percentage of a particular class C that are truly classified as class C among all instances which are classified as class C. Mathematically precision can be defined as the ratio $\frac{tp}{tp+fp}$, where *tp* is the number of true positives and *fp* is the number of false positives. On the other hand *recall* is defined as the percentage of members of a class C that are correctly classified as belonging to class C. Mathematically this value can be represented as the ratio $\frac{tp}{tp+fn}$ where *fn* stands is the number of false negatives.

We implement the AdaBoost algorithm as described in [28] by J. Zhu et al. Namely two variants are tested depending on what parameter the algorithm uses to adapt in every iteration. In our case AdaBoost1 adapts based on predicted class label errors, whereas AdaBoost2 adapts based on class probabilities [29]. For Random Forest, we vary the tree depths to be 2 and 3 and these are denoted as Random Forest1 and Random Forest 2. Note that an increase in the depth beyond this provides no significant improvement in performance. For Naive Bayes ,we tested the algorithm with both normalized and non-normalized data and found that the latter gave higher accuracy and recall scores. Hence only results for only non-dnormalized data are being presented. The overall results for all schemes are detailed in Table II.

To further validate the efficacy of our model, we also decided to test it on data from a different time period to avoid overfitting that might arise due to a large number of samples being from the same time period. This also helps us assess how attack behavior changes over time. Namely, we extract IP header information of IoT devices found in the CAIDA dataset on the 1st day of February 2017 and also the 1 st day of March 2017. Based on our labeling methodology, we found that over 1.7 million IoT devices were found to be performing scanning and over 100,000 devices were sending backscatter traffic. Furthermore a meagre 7764 were classified as misconfiguration traffic in February, this figure reduces to 1.2 million IoT devices scanning and almost 80,000 devices sending backscatter traffic in March. The parameters used here are the same as those used for the January dataset. For training purposes, we also use the same methodologies as proposed earlier and the detailed findings presented in Tables III and IV

Based on the above results, we can infer that Gradient Boosting outperforms all other algorithms, both in terms of accuracy and precision scores. This is followed closely by Random Forest and finally AdaBoost and Naive Bayes, with Naive Bayes performing the worst. The low performance of Naive Bayes can be attributed to its independence assumption between the features, which is clearly not the case in network traffic features. We can also see in Table III that the performance deteriorates very slightly in the case of Gradient Boosting and Random Forest, whereas it drops significantly for Naive Bayes. Although Random Forest does not perform as well as Gradient Boosting, in many practical applications it is preferred over Gradient Boosting due to its distributed nature which allows for parallelization. Thus, constraints like the size of data and requirement of real-time classification can result in Random Forest being selected over Gradient Boosting. However, Gradient Boosting excels at handling outliers, selecting important variables and performs well even with missing data and this scheme has even been proven empirically to provide better accuracy [30]. Hence the trade off, as is often the case with any ML algorithm selection, lies between computation cost and prediction accuracy [29].

V. Conclusions and Future Work

This paper presents a novel model that uses ML techniques to classify malicious IoT traffic as performing scanning, being victims of DoS attacks, or as misconfiguration. In particular the feature vectors here are based on statistical properties extracted from darknet data from CAIDA's network telescope. This model achieved high recall and precision scores using the Gradient Boosting and Random Forest classification algorithms. We also demonstrated the reliability of the solution by selecting data from different time frames. Overall, various organizations especially large scale enterprises and ISPs, can use this solution along side their intrusion detection systems to identify anomalous IoT behaviors within and outside the domains. This scheme can help in quick identification of unsolicited IoT devices which can then be patched or wiped to help prevent their recruitment in large scale bot networks. Hence, this will reduce the probability of massive Internet-scale DDoS attacks.

However, one of the limitations of the model is that it is trained solely on darknet data which mostly represents malicious activity. Hence in the future we plan to test this framework using a more realistic setup that emulates a typical IoT environment, i.e., with both benign and malicious IoT data being used to train and test the ML algorithms.

References

[1] Hajime iot worm infects devices to head off mirai. https://www.helpnetsecurity.com/2017/04/19/hajime-iot-worm/.

[2] ”Reaper: Calm before the iot security storm?” https://krebsonsecurity.com/2017/10/reaper-calm-before-the-iot-security-storm/.

[3] T.V.Phan, N.K.Bao, and M.Park. “Distributed-SOM: A Novel Performance Bottleneck Handler for large-sized Software-Defined Networks under Flooding Attacks”. In *Journal of Network and Computer Applications*, volume 91, pages 14–25, August 2017.

[4] J.Vidal, A. Orozco, and L.Villalba. “Adaptive Artificial Immune Networks for Mitigating Dos Flooding Attacks”. In *Swarm and Evolutionary Computation*, volume 38, pages 94–108, February 2018.

[5] J.Erman, M.Arlitt, and A.Mahanti. “Traffic Classification Using Clustering Algorithms”. In *SIGCOMM'06 Workshops*, pages 281–286, September 2006.

[6] H.A.H.Ibrahim, O.R.A.Zuobi, M.A.Al-Namari, G.M.Ali, and A.A.A.Abdalla. “Internet Traffic Classification using Machine Learning Approach: Datasets Validation Issues”. In *Conference of Basic Sciences and Engineering Studies*, pages 158–166, February 2016.

[7] Y.Meidan, M.Bohadana, and A.Shabtai. “ProfilIoT: A Machine Learning Approach for IoT Device Identification Based on Network Traffic Analysis”. In *SAC*, pages 506–509, April 2017.

[8] N.Furutani, T.Ban, J.Nakazato, J.Shimamura, J.Kitazono, and S.Ozawa. “Detection of DDoS Backscatter Based on Traffic Features of Darknet TCP Packets”. In *Ninth Asia Joint Conference on Information Security*, pages 39–43. IEEE, September 2014.

[9] D.Bekerman, B.Shapira, L.Rokach, and A.Bar. “ Unknown Malware Detection Using Network Traffic Classification”. In *IEEE Conference on Communications and Network Security*, pages 134–142, September 2015.

TABLE II: Recall and Precision Values for the January Data

Metric	NaiveBayes	AdaBoost1	AdaBoost2	Random Forest1	Random Forest2	Gradient Boost
Recall	75.63	72.99	89.32	93.93	96.19	99.88
Precision	82.43	82.81	93.24	93.72	96.41	99.88

TABLE III: Recall and Precision Values for the February Dataset

Metric	NaiveBayes	AdaBoost1	AdaBoost2	Random Forest1	Random Forest2	Gradient Boost
Recall	57.53	96.16	95.74	95.63	98.04	99.97
Precision	97.56	98.74	96.40	95.55	97.77	99.61

TABLE IV: Recall and Precision Values for the March Dataset

Metric	NaiveBayes	AdaBoost1	AdaBoost2	Random Forest1	Random Forest2	Gradient Boost
Recall	60.14	97.17	96.60	93.32	98.33	99.90
Precision	95.54	99.01	95.59	93.67	96.72	99.43

[10] Y.Meidan, M.Bohadana, and A.Shabati. “Detection of Unauthorized IoT Devices Using Machine Learning Techniques”. In *arXiv:1709.04647v1*, 2017.

[11] D.Apiletti, E.Baralis, T.Cerquitelli, and V. D.Elia. “Characterizing Network Traffic by means of the NetMine Framework”. In *Computer Networks*, volume 53, pages 774–789, April 2009.

[12] E. Hodo, X. Bellekens, A. Hamilton, P. L. Dubouilh, E. Iorkyase, C. Tachtatzis, and R. Atkinson. “Threat Analysis of IoT Networks using Artificial Neural Network Intrusion Detection System”. In *2016 International Symposium on Networks, Computers and Communications (ISNCC)*, pages 1–6, May 2016.

[13] J.Canedo and A.Skjellum. “Using Machine Learning to Secure IoT Systems”. In *14th Annual Conference on Privacy, Security and Trust*, pages 219–222, December 2016.

[14] A.L Blum and P. Langley. ”Selection of Relevant Features and Examples in Machine Learning”. In *Artificial Intelligence*, volume 97, pages 245–271, 1997.

[15] S.Zander, T.Nguyen, and G.Armitage. “Automated Traffic Classification and Application Identification using Machine Learning”. In *IEEE Conference on Local Computer Networks 30th Anniversary*, pages 250–257, December 2005.

[16] T.T.T Nguyen and G.Armitage. “A Survey of Techniques for Internet Traffic Classification using Machine Learning”. In *IEEE COMMUNICATIONS SURVEYS and TUTORIALS*, volume 10, pages 56–76, 2008.

[17] A.Sivanathan, D.Sherratt, H.H.Gharakheili, A.Radford, C.Wijenayake, A.Vishwanath, and V.Sivaraman. “Characterizing and Classifying IoT Traffic in Smart Cities and Campuses”. In *IEEE Infocom Workshop on Smart Cities and Urban Computing*, pages 559–564, May 2017.

[18] T Ban. “3-3 Data mining applied to Darknet Traffic Analysis”. *Journal of the National Institute of Information and Communications Technology*, 63:45–54, 01 2016.

[19] N. Brownlee. One-way Traffic Monitoring with iatmon. In *Passive and Active Network Measurement Workshop (PAM)*, volume 7192, pages 179–188. Cooperative Association for Internet Data Analysis (CAIDA), March 2012.

[20] J. Treurniet. “A Network Activity Classification Schema and Its Application to Scan Detection”. In *IEEE/ACM Transactions on Networking*, volume 19(5), pages 1394–1404, 2011.

[21] J. Liu and K. Fukuda. “Towards a Taxonomy of Darknet Traffic”. In *Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 37–43, September 2014.

[22] M. Wustrow, E. Karir and M. Bailey. “Internet Background Radiation Revisited”. In *10th ACM SIGCOMM Conference on Internet Measurement*, pages 62–74, November 2010.

[23] R. Polikar. “Ensemble based systems in decision making”. In *IEEE Circuits System*, volume 6, pages 21–45, 2006.

[24] J. Zhang, M. Zulkernine, and A. Haque. “Random Forests Based Network Intrsuion Detection Systems”. In *IEEE Trans. Syst. Man Cybern*, volume 38, pages 649–659, August 2008.

[25] F. Ghaibian and A. Ghorbani. “Comparative Study of Supervised Machine Learning Techniques for Intrusion Detection”. In *5th Annu. Conf. Commun. Netw. Serv. Res*, pages 350–358, May 2007.

[26] J. Han, M. Kamber, and J. Pei. *“Data Mining”*. Morgan Kaufmann Publishers, 2000.

[27] N.B. Amor, S. Benferhat, and Z. Elouedi. “Naïve Bayes vs. Decision Trees in Intrusion Setection Systems”. In *ACM Symp. Appl. Comput*, pages 420–424, March 2004.

[28] J. Zhu, H. Zou, and T. Hastie. ”Multi-class AdaBoost”. In *Statistics and Its Interface*, volume 2, pages 349–360, 2009.

[29] T. Hastie, R. Tibshirani, and J. Friedman. *”Elements of Statistical learning”*. Springer, 2 edition, 2009.

[30] JO Ogutu, HP Piepho, and T. Schulz-Streeck. “A comparison of random forests, boosting and support vector machines for genomic selection”. In *BMC*, May 2011.