Avrupa Bilim ve Teknoloji Dergisi
Özel Sayı 49, S. 75-82, Mart 2023

Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
Special Issue 49, pp. 75-82, March 2023

Research Article

Forensic Analysis of APT Attacks based on Unsupervised Machine Learning

Mohammed Adnan, Dima Bshara, Ahmed Awad*

Department of Networks & Information Security, Faculty of Engineering & Information Technology, An-Najah National University, Nablus, Palestine.

mohammedix88@gmail.com, deema.bshara55@gmail.com, ahmedawad@najah.edu

(2nd International Conference on Scientific and Academic Research ICSAR 2023, March 14-16, 2023)

(**DOI:** 10.31590/ejosat.1265586)

ATIF/REFERENCE: Adnan, M., Bshara, D. & Awad, A. (2023). Forensic Analysis of APT Attacks based on Unsupervised Machine Learning. *European Journal of Science and Technology*, (49), 75-82.

Abstract

Advanced Persistent Threat (APT) has become the concern of many enterprise networks. APT can remain unde- tected for a long time span and lead to undesirable consequences such as stealing of sensitive data, broken workflow, and so on. APTs often use evasion techniques to avoid being detected by security systems like Intrusion Detection System (IDS), Security Event Information Management (SIEMs) or firewalls. Also, it makes it difficult to detect the root cause with forensic analysis. Therefore, companies try to identify APTs by defining rules on their IDS. However, besides the time and effort needed to iteratively refine those rules, new attacks cannot be detected. In this paper, we propose a framework to detect and conduct forensic analysis for APTs in HTTP and SMTP traffic. At the heart of the proposed framework is the detection algorithm that is driven by unsupervised machine learning. Experimental results on public datasets demonstrate the effectiveness of the proposed framework with more than 80% detection rate and with less than 5% false-positive rate.

Keywords: Unsupervised Machine Learning, Advanced Persistent Threats (APTs), HTTP, SMTP, Forensic Analysis

* Corresponding Author: ahmedawad@najah.edu

1. Introduction

An Advanced Persistent Threat (APT) is an organized cyber attack by a group of skilled, sophisticated threat actors. [1] [2]. Attackers plan their mission carefully against strategic targets and execute out over a long time window. APTs are highly sophisticated compound attacks involving multiple phases with diverse techniques with zero-day exploits and malware. APT campaigns tend to involve multiple attack vectors as well as multiple access points. Thus, APT attacks are difficult to be identified [3] [4]. Since APTs target critical companies and other governmental organizations, they constitute one of the most serious security challenges [5].

APT exploits a variety of tactics and techniques and a large library of custom and open-source malware. It actually utilizes 13 different tactics defined by MITRE ATT&CK. This includes: reconnaissance, resource development, initial Access, execution, persistence, privilege escalation, credential access, discovery, lateral movement, defense evasion, command and control, collection, and exfiltration. To achieve their objectives, APT attackers use one or more techniques. For instance, APT38 is a North Korean state-sponsored threat group that specializes in financial cyber operations; it has been attributed to the reconnaissance general bureau. Active since at least 2014, APT38 has targeted banks, financial institutions, casinos, cryptocurrency exchanges, SWIFT system endpoints, and ATMs in at least 38 countries worldwide [6].

APTs are detected and prevented in many ways. The most common software or hardware are Security Event Information Management (SEIM) and Intrusion Detection System (IDS). However, when both the APT traffic is indistinguishable from the normal traffic, it becomes extremely difficult to detect such attacks. Furthermore, exploiting evasive techniques allow APT attacks to easily bypass the detection systems [7] [8]. Therefore, exploiting adaptive detection techniques reinforced with deep inspection might be inevitable to properly equip different enterprises with effective detection techniques against APT attacks [9].

HTTP and SMTP malicious traffic are the most common and annoying types of APT attacks. These malicious traffics analogous to genuine SMTP and HTTP traffic. They use the same TCP port 80 for HTTP and 25 for SMTP and respect the SMTP and HTTP messages structure [10] [11]. Thus, detecting such malicious traffic cannot be conducted with simple analysis of ports numbers or thorough inspection of the packet structure. A deep analysis is required to assess the behavior of the entities by combining multiple information such as the number of bytes exchanged, the duration, the time the message was sent, the time the message takes to response, and other related information.

These days most organizations use SEIMs to oversee occasion logs and safeguard their organization from assaults. Nonetheless, composing discovery rules in SIEMs while considering countless measures is no longer imaginable. It raises a concern about the acceptable threshold value to tag a traffic instance as malicious or not. APTs are being executed by gifted assailants, whereas the rate of attacks is controlled to stay undetected for sufficiently long time period.

Machine learning techniques can be effectively exploited in such a complex detection task. There exists two classes of machine learning techniques: supervised and unsupervised methods. Supervised learning (SL) is practiced through feeding the system with a set of input-output examples to infer the needed function for classification. In supervised learning, each example in the training set is a pair consisting of an input vector and the desired output vector [12]. Nonetheless, this approach has two downsides in APT detection application. In the first place, the presentation of the algorithm relies exhaustive preparation of the dataset. An algorithm prepared with a particular arrangement of danger methods probably will not recognize a zero-day attack. An algorithm trained to recognize authentic traffic will be well specific for the association where the preparation dataset has been captured or caught. The subsequent issue is related to the expense of labeling datasets. Countless picture datasets exist because label pictures do not need explicit abilities. In the case of network security, the context is different. Labeling network security traffic must be performed by security specialists, which infers a dramatic expansion to the cost of this undertaking. Unsupervised machine learning algorithms, on the other hand, infer patterns from a dataset without reference to known, or labeled, outcomes and do not need any training [13]. This consequently resolves the two aforementioned drawbacks. However, for the unsupervised class, it is difficult to have the same result and the same accuracy as the supervised method or algorithm achieves.

In this paper, we propose a framework to detect APT attacks for both HTTP and SMTP traffic. We used Splunk which is SIEM to help us to analyze the data and to analyze the collection log from different sources, also, the reason that we used Splunk is that Splunk has the machine learning toolkit that helps us to test our chosen algorithm.

This paper will propose a comprehensive framework driven by unsupervised machine learning detection algorithm, to identify APT attacks for both HTTP and SMTP traffic. Our contributions are summarized as following: an unsupervised machine learning technique, we focus on HTTP and SMTP traffic. The obtained results are further analyzed to conduct forensic analysis. Our contributions are summarized as follows:

- We propose a framework, driven by unsupervised ma- chine learning algorithms to detect abnormal traffic for both HTTP and SMTP packets, through dynamically identifying the lower band, upper band, and outliers with proper choice of statistical measures.

- Three types of unsupervised machine learning algorithms are compared and evaluated in terms of Detection Rate (DR) and False Positive Rate (FPR).
- Combination of different algorithms are tested and evaluated for further enhancement of the results.
- The obtained results are further analyzd to do foreonesic analysis and obtain the attack vector as well as the targeted machines.

The rest of the article is as follows. First, we present the related work. Then, we introduce a background for machine learning algorithms and APT in section **3**. Section **4** describes our proposed algorithm, the dataset, and the comparison of the three unsupervised algorithms. Then we present in section **5** the experimental result and the attack scenario after forensic analysis. Finally, section 6 concludes the article.

2. Related Work

Several techniques have been proposed in the literature to detect generic network intrusions as well as abnormal HTTP and SMTP traffic. This has been conducted using both supervised and unsupervised machine learning.

Leon et al. introduced in [14] a way to deal with anomaly detection based on Unsupervised Niche Clustering (UNC).The UNC is a hereditary niching strategy for clustering, which can decide the number of clusters automatically. The creators describe each predicted cluster using a fuzzy membership function. Also, they utilize the Maximal Density Estimator (MDE) refinement to work on the nature of the arrangement and Principal Component Analysis (PCA) to reduce the complexity of the dataset and further improve the exhibition of the proposed approach. The model has been tested on a public dataset and has shown a detection rate of 99.2% with false alarm rate of 2.2%.

Ibrahim Ghafir et al. proposed in [15] a novel machine learning-based system, namely MLAPT, to detect and predict APT attacks in a holistic approach. The MLAPT consists of three main phases: threat detection, alert correlation, and attack prediction. The MLAPT is able to predict APT in its early steps with a prediction accuracy of 84.8%.

Cho Do Xuan has exploited machine learning to propose a method of detecting APT attacks based on abnormal behaviors of network traffic [16]. In this work, two components are defined: Domain and IP of the abnormal behavior of APT attacks in network traffic. Then, these behaviors are esteemed and classified based on the Random Forest classification algorithm to conclude the behavior of APT attacks.

Thi Quynh Nguyen et al. made a comparison of the performance of four unsupervised machine-learning algorithms [17]: K-means, Gaussian Mixture Model (GMM), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and Local Outlier Factor (LOF) on the Boss of the SOC Dataset Version 1 (Botsv1). Then they present a technique that merges DBSCAN and K Nearest Neighbor (KNN) to gain 100% detection rate and between 1.6% to 2.3% false-positive rate.

In the work publised in [18], the authors introduced a semi-unsupervised anomaly detection method. They made assumption that during the learning phase (for the captured volume of HTTP traffic), Only a small fraction of the samples is labeled. Their experiments show that the proposed method achieves ratios of true positive and false positive errors below 1%.

Feature selection techniques have been exploited to analyze phishing datasets in the work published in [19]. In this context, information gain, gain ratio, Relief-F, and Recursive Feature Elimination (RFE) for feature selection have been utilized with the aid of diverse machine learning algorithms. The highest scoring classifiers have been then combined to improve the classification accuracy which has reached up to 97.4%.

Although abnormal HTTP and SMTP detection works achieve good detection results, some of them use supervised learning approaches and thus, require a training dataset which is expected to fail in detected zero-day attacks. Although some other approaches have utilized unsupervised ML, they have been applied to small datasets. The adaptation of such approaches on a wider scale is mandatory with the increasing number of APT attacks. Furthermore, most of the core fields those approaches inspect might result in high false positive rate if there are multiple servers in the network, since the proposed classifiers do not distinguish user traffic from the server traffic. Therefore, we propose in this paper a behavioral analysis approach based on probabilistic and distributed statistic algorithms for detecting malicious HTTP and SMTP traffic. We evaluate our proposed approach on public datasets: Botsv1 and Botsv2 datasets of Splunk project.

3. Background

Although translating the aforementioned MITRE ATT&CK tactics into a set of detection rules to be fed into an Intrusion Detection System (IDS) is possible, writing such rules is a very exhaustive as it needs building many highly composite indicators of compromise.

3.1. Advanced Persistent Threats (APTs)

APT attacks mainly aim to mine highly sensitive data through silently establishing a long term presence in a network. APTs generally target businesses and governments with exploiting sophisticated malware [20]. For instance, when studying APT38, the attacker has conducted spearphishing campaigns using malicious email attachments with SMTP protocol. Moreover, the attackers in the APT group encrypt most of their traffic with SSL to hide malicious traffic inside authorized network traffic. After the attacker gets into the network, APT38 has used brute force techniques over HTTP traffic to attempt account access when passwords are unknown or when password hashes are unavailable. Thereafter, APT38 has used a backdoor

with the capability to download and upload files to and from a victim's machine.

Over the years, APT groups have targeted traditional financial institutions, making the targeting of SWIFT systems their specialty. The group targets are geographically diverse, with financial institutions in Africa, Southeast Asia, India, and Latin America [6].

3.2. Machine Learning

As explained before, applying supervised learning algorithms for detecting malicious network traffic raises issues due to the exhaustivity of the training dataset and high labelling cost. There exist, however, many unsupervised anomaly detection algorithms. The core motivation of using unsupervised ML algorithms in this work is to dynamically define the thresholds needed to distinguish malicious traffic from the normal traffic. Otherwise, a threshold has to be set manually, which is more likely to increase the rate of false positives.

In this paper, we utilize two categories of unsupervised ML algorithms to detect APT attacks: Probabilistic-based algorithms and outlier detection- based algorithms.

3.1.1. Probabilistic-based ML Algorithm

This algorithm identifies anomalous events by computing a probability for each event and then detecting unusually small probabilities. The probability is defined as the product of the frequencies of each individual field value in the event. Each data item is treated based on its type as following :

- For categorical fields, the frequency of a value x represents the number of times x occurs divided by the total number of events.
- For numerical fields, we first build a histogram for all the values, then compute the frequency of a value x as the size of the bin in the resultant histogram that contains X divided by the number of events.

3.1.2. Outlier Detection-based ML Algorithm

This algorithm determines values that appear to be extraordinarily higher or lower than the rest of the data. Identified outliers are indicative of interesting, unusual, and possibly dangerous events. This algorithm is compatible with two statistical measurements: Standard deviation and Median absolute deviation.

When a situation violates the expectations for a parameter, it results in an outlier. The steps for utilizing this algorithm is as following [13]:

- Select a numeric field to analyze in the packet.
- Select a statistical measurement to detect outliers. The proper metric is selected based on the distribution of the data to be analyzed as follows:
  - Standard Deviation : This method is appropriate If the data exhibits a normal distribution. Since the standard deviation method centers on the mean, it is more impacted by outliers.
  - Median Absolute Deviation : This method applies a stricter interpretation of outliers than standard deviation because the measurement centers on the median and uses Median Absolute Deviation (MAD) instead of standard deviation.
- Specify the threshold multiplier to identify the outlier envelop.

4. Proposed Approach

The proposed framework for detecting abnormal HTTP and SMTP traffic is driven by two hypotheses, as published in [17]: First, APTs are higky complex attacks perpetrated by experts of highly skilled attackers whose objective is to stay undetected for an extended period of times. Thus, most of the network traffic is genuine. The traffic corresponding to attacks is assumed to be very low (e.g., less than 3% of the traffic at maximum). In other words, if the malicious traffic rate exceeds this presumed rate, it is assumed that traditional detection systems are capable of detecting it. The second hypothesis is related to detecting anomalies without previous knowledge of the network and the traffic. This is translated in the form of detecting outliers. Consequently, the network traffic must be preprocessed to separate the different network services (DNS, HTTP(S), SMTP(S), etc.) before applying outlier detection.

Accodingly, our detection framework is summarized as following (shown in Fig. 1):

1) Categorize the network traffic by service.
2) For each service, apply unsupervised machine learning to detect abnormal traffic. In this context, three algorithms are evaluated: Histogram (probabilistic-based), standard deviation, and medium absolute deviation (outlier-detection based) algorithms (see Fig. 2).
3) Post-analyze the detected values to determine the infected machine and predict the detailed scenario for forensic analysis.

4.1 Chosen Dataset

Both experimentation and the evaluation have been conducted using Botsv1 and Botsv2 datasets from Splunk project [21] [22]. Both datasets are public. Botsv1 dataset contains HTTP-based execution and reconnaissance. On the other hand, Botsv2 contains SMTP-based spear-phishing email or initial access. Thus, our goal for the first dataset is to detect abnormal HTTP traffic which may carry executing and privi- lege escalation data. For the external scenario. Regarding the second version of the dataset, our goal isto detect SMTP traffic which may carry initial access. In this context, it is important to mention that both datasets contain evidence captured during actual computer security incidents, or from realistic lab recreations of security incidents. Both datasets consist of two parts: The original

dataset containing all data, and a much smaller version of the original dataset containing only attack data. The original dataset is available in several formats: compressed, several JSON files by source type, and several CSV files by source type (such as stream: DNS, stream: HTTP, stream: SMTP, etc).

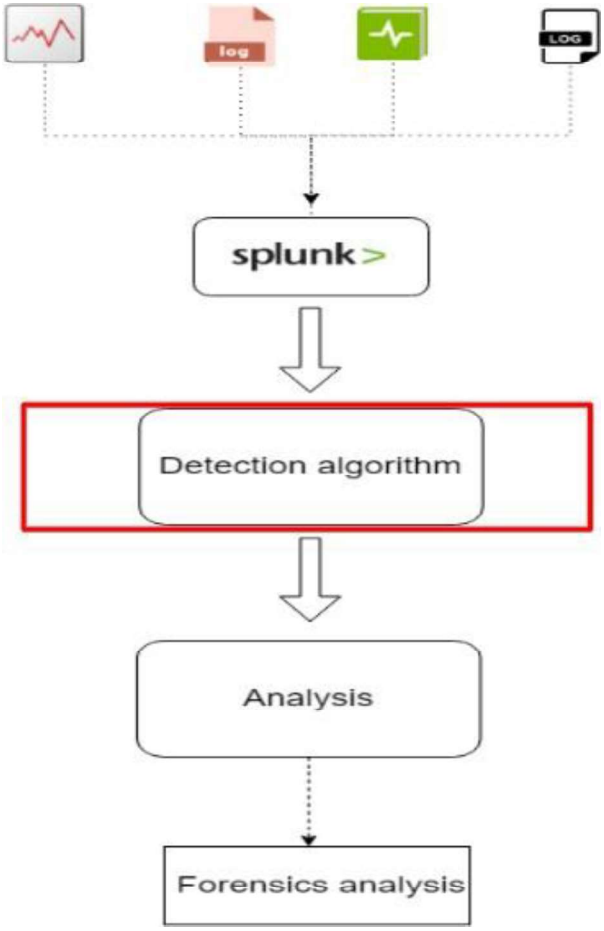

Fig. 1. **Proposed Framework**

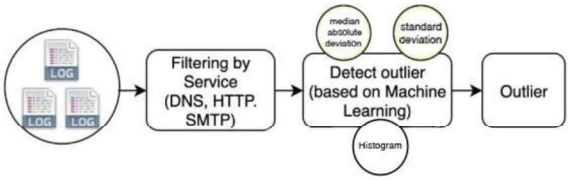

Fig. 2. **Detection Algorithm.**

4.2 Data Preparation

We deal with the datasets as an index, so we compare the original dataset, which contains the normal data with attacker data, and with the attacker dataset to choose the interesting field that changes between the two datasets. The fields of interest are: source IP, destination IP, number of bytes in, number of bytes on, client round trip time (rtt), response acknowledgment time, and the packet size.

After selecting and preparing the features, we label the Botsv1-HTTP dataset using the HTTP-attack logs that allow us to determine which connection is abnormal. The resulting labeled dataset contains 40,035 genuine HTTP connections and 23,435 attack connections. Similarly, we label the Botsv2- SMTP dataset using the SMTP-attack logs that allow us to determine which connection is abnormal. The resulting labeled dataset contains 790,683 genuine SMTP connections and 560,726 attack connections. As we conclude here, at the beginning of the APT group, the first stage will take more than 40% of the traffic. Furthermore, it will be easier than using Command & Control (C& C) server to determine the root cause at first in forensic analysis because if the attacker starts C&C, the mission will be hard to detect it using the all dataset as one bulk.

4.3 Detection Algorithm

The second step of our methodology consists of applying unsupervised ML detection algorithms to identify abnormal traffic. As mentioned before, we apply three algorithms, namely, standard deviation, medium absolute deviation, and histogram.

Once the volume of the fields of interest is obtained for both HTTP and SMTP datasets, we calculate the lower-bound and the upper-bound for standard deviation and medium absolute deviation algorithms. Regarding the histogram algorithm, the summation for each interest feature is computed. For each field of interest, any value outside the lower and upper bounds of the used statistical measure is tagged as an outlier. We use Detection Rate (DR) and the False Positive Rate (FPR) to evaluate the performance of the algorithms. The detection rate is the number of attacks detected by the system divided by the number of attacks in the dataset. The false positive rate is the number of normal connections that are mis-classified as attacks divided by the number of normal connections in the dataset. As consequence, a good algorithm should achieve a high DR value while keeping the FPR low.

4.4 Forensic Analysis

Once the malicious packets are detected by the ML algorithm, each packet is deeply inspected to envision the the scenario of the attack. Such analysis is insightful in deciding whether the attack is internal or external. Furthermore, the source machine for attacks with its intended victims are identified.

5. Experimental Results

All experiments have been conducted using Splunk. First, we try each of the aforementioned fields in the packet to identify the most influencing fields on the classification outcome. So, we found that the volume of client Rount Trip Time (RTT) , response acknowledgement time, and volume of bytes are the core fields to input to the detection algorithm.

5.1 Comparison between ML Algorithms

Table I displays the abnormal HTTP connection of the three algorithms for the dataset. For the Botsv1 dataset, medium absolute deviation yields poor results with DR value of 74% and the FPR of 70%. The DR of the histogram yields better which is 81.3%, and the FPR is about 3%. Regarding, the standard deviation, both DR and FPR are the best result for that dataset, with DR being 90% and FPR being 2.5%. Those results are argued in twofold: (1) The volume of the first dataset is relatively small. (2) The packet size of HTTP is relatively small if compared with SMTP. Consequently, the standard deviation algorithm outperforms others in detecting abnormal traffic.

Table I: DR and FDR Comparison for HTTP

Algorithm	DR	FPR
Histogram	81.3%	3%
Standard Deviation	90%	2%
Median Absolute Deviation	74%	33.4%
Histogram & Standard Deviation	100%	1.7%

In the next step, we try to make a combination of two algorithms to improve the result, which are histogram and standard deviation. Firstly, we calculate the standard deviation, and the output result is inputted to the Histogram. This combination achieves high DR which is 100%, and FPR 1.7%, which is an acceptable value. However, the accuracy of those results is not 100% because the dataset is relatively small.

Figure 3 shows the statistics for the combination algorithm (histogram and standard deviation) when applied to HTTP traffic with the volume of the chosen fields. Also, we indicate whether the traffic is considered to be an outlier or not.

In Botsv1 dataset, it is easy to determine the infected host using the volume of bytes. However, if there are multiple servers, a dramatic increase in the FBR is expected. We solve the multiple server problem by making classification between server and client, by choosing client RTT to make sure that the connection does not start from the server because if the connection starts from it, this means the server is already compromised.

To validate the effectiveness of the algorithms being tested, we use Botsv2 dataset to get a more exact accuracy. Table II displays the abnormal SMTP connection of the three algorithms and the combination of histogram and standard deviation for the dataset. So after applying the histogram and Standard deviation algorithm, the DR is found to be 79%, and the FPR is about 4%. We assume that the reason for this relative degradation in the result is the massive size of traffic. However, the median absolute deviation algorithm outperforms others with 96% DR and 1.2% FPR. Clearly, this algorithm progosses well for massive traffic.

Fig. 4 displays the statistics for median absolute deviation algorithm when applied to SMTP traffic (Botsv2 dataset). Notice that the detected outliers are tagged when a value of 1 is assigned for the Boolean variable is outlier.

We conclude that the reason for the obtained results is attributed to the size of the input data. If we need to obtain high DR with low FPR for a relatively small amount of traffic, a combination of histogram and standard deviation algorithms is utilized. However, median absolute deviation is recommended for massive amount of traffic.

Table II: DR and FDR Comparison for SMTP

Algorithm	DR	FPR
Histogram	77%	10%
Standard Deviation	78%	12.4%
Median Absolute Deviation	96%	1.2%
Histogram & Standard Deviation	79%	4%

5.2 Forensic Analysis

After using the proper machine-learning algorithm, we analyze APT attacks for datasets and show how machine learning helps us envision the scenario more quickly than the traditional way. For the Botsv1 dataset, after analysis and deep inspection of packets according to ML and experiment, we predict all scenarios for this attack as shown in Fig. 5, which is as follows:

At first, the attacker is in the internal network, and he knows the admin machine after doing some reconnaissance. Then the attacker targets the machine using brute force attack to get privilege escalation. After this, the attacker installs Remote Access Terminal (RAT) which poisons a backdoor Trojan that allows the remote attackers to perform various malicious activities on the compromised machine and execute it. At the end, the attacker opens a connection outside the network and closes the internal one.

For SMTP with Botsv2, the attacker tries first to send a phishing email to all companies until an employee reacts to the phishing email. Then the attacker sends spearphishing email by hiding a malicious ZIP file including a back door inside it. Also, the attacker encrypts most of their email traffic with SSL to hide malicious traffic inside authorized network traffic. Fig. 6 shows this scenario.

Both scenarios in the datasets are similar to APT38 which starts at first with a spearphishing email.Once the user executes the ZIP file, the attacker opens a backdoor and starts reconnaissance to know where the admin machine is, after that, it tries to get privilege escalation to be admin. Then it installs a ZIP file in the admin machine to get the root privilege. So, as we see that without machine learning this will take days, even

a month, but if we use machine learning, we can predict all scenarios and get the root cause in less than a day.

6. Conclusions

In this paper, we present an Advanced Persistent Threat (APT) detection framework based on unsupervised machine learning for detecting malicious SMTP and HTTP traffic. We used the logs provided by Splunk and extracted the information to build datasets. We compared the performance of yhree algorithms standard deviation, median absolute deviation, and histogram without sampling (as one block). Combining standard deviation and histogram gave the best result for relatively szmall traffic with 100% DR and 1.7% FPR. For large amount of data, the medium absolute deviation algorithm outperforms others. After detecting the outliers, we analyzed the result to find the root cause of the attack using forensic analysis, which would have taken much more time without using machine learning.

src_ip	sum(bytes)	sum(response_ack_time)	sum(bytes_in)	sum(bytes_out)	absDev	isOutlier	lowerBound	median	medianAbsDev	upperBound
1.160.115.70	58541	208927221	39623	18918	198240461	1	-10512282	10686760	10599521	31885802
1.160.117.198	413794	1521648047	280097	133697	1510961287	1	-10512282	10686760	10599521	31885802
1.160.118.132	244301	922754717	165258	79043	912067957	1	-10512282	10686760	10599521	31885802
1.160.127.3	211215	778302606	143016	68199	767615846	1	-10512282	10686760	10599521	31885802
104.47.32.205	493	299608	204	289	10387152	0	-10512282	10686760	10599521	31885802
104.47.32.208	499	315588	210	289	10371172	0	-10512282	10686760	10599521	31885802
104.47.32.231	500	224675	211	289	10462085	0	-10512282	10686760	10599521	31885802
104.47.32.41	84192	30657317	21764	62428	19970557	0	-10512282	10686760	10599521	31885802
104.47.32.43	107359	61598882	27907	79452	50912122	1	-10512282	10686760	10599521	31885802
104.47.32.45	6321	382220	5995	326	10304540	0	-10512282	10686760	10599521	31885802
104.47.32.47	138446	75496067	113274	25172	64809307	1	-10512282	10686760	10599521	31885802
104.47.32.49	20526	41451091	3032	17494	30764331	1	-10512282	10686760	10599521	31885802
104.47.32.51	1057	537925	464	593	10148835	0	-10512282	10686760	10599521	31885802
104.47.32.56	2916	307833	2612	304	10378927	0	-10512282	10686760	10599521	31885802

Fig 4: Algorithm Statistics for SMTP

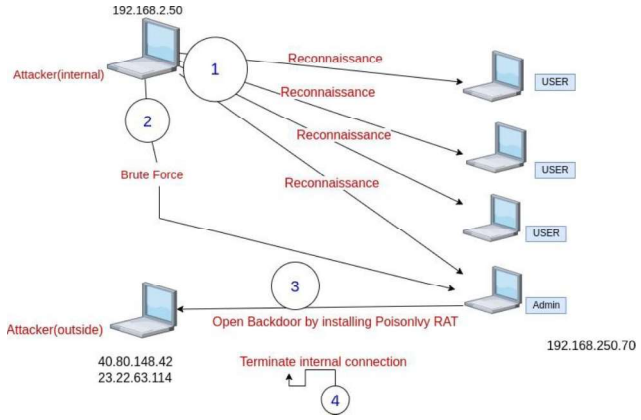

Fig 5: Forensic Analysis for HTTP Scenario

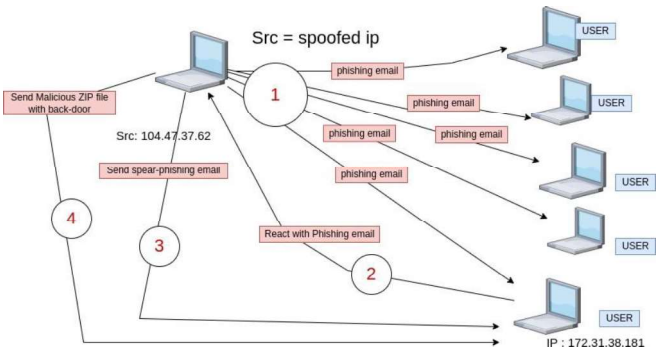

Fig 6: Forensic Analysis for SMT Scenario

References

[1] A. Benzekri, R. Laborde, A. Oglaza, D. Rammal, and F. Barre`re, "Dynamic security management driven by situations: An exploratory analysis of logs for the identification of security situations," in 2019 3rd Cyber Security in Networking Conference (CSNet), 2019, pp. 66–72.

[2] (2015) Introduction to Cybercrime. [Online]. Available: interpol.int/en/Crimes/Cybercrime

[3] Q. Zhang, H. Li, and J. Hu, "A study on security framework against advanced persistent threat," in 2017 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC), 2017, pp. 128–131.

[4] (2022) Advanced persistent threat (apt) attacks. [Online]. Available: https://www.cynet.com/advanced-persistent-threat-apt-attacks

[5] M. Khosravi-Farmad, A. A. Ramaki, and A. G. Bafghi, "Moving target defense against advanced persistent threats for cybersecurity enhancement," in 2018 8th International Conference on Computer and Knowledge Engineering (ICCKE), 2018, pp. 280–285.

[6] (2022) Tactics, techniques, and procedures. [Online]. Available: https://attack.mitre.org/

[7] T.-H. Cheng, Y.-D. Lin, Y.-C. Lai, and P.-C. Lin, "Evasion techniques: Sneaking through your intrusion detection/prevention systems," IEEE Communications Surveys Tutorials, vol. 14, no. 4, pp. 1011–1020, 2012.

[8] H. Kılıc¸, N. S. Katal, and A. A. Selc¸uk, "Evasion techniques efficiency over the ips/ids technology," in 2019 4th International Conference on Computer Science and Engineering (UBMK), 2019, pp. 542–547.

[9] D. X. Cho and H. H. Nam, "A method of monitoring and detecting apt attacks based on unknown domains," Procedia Computer Science, vol. 150, pp. 316–323, 2019, proceedings of the 13th International Symposium "Intelligent Systems 2018" (INTELS'18), 22-24 October, 2018, St. Petersburg, Russia. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050919304041

[10] (2019) Smtp security. [Online]. Available: https://mailtrap.io/blog/smtp- security

[11] (2017) Http attacks. [Online]. Available: https://blog.radware.com/security/2017/11/http-attacks

[12] S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach. London: Prentice Hall, 2010.

[13] M. Mohri, A. Rostamizadeh, and A. Talwalkar, Foundations of Machine Learning.

[14] E. Leon, O. Nasraoui, and J. Gomez, "Anomaly detection based on unsupervised niche clustering with application to network intrusion detection," in Proceedings of the 2004 Congress on Evolutionary Com- putation (IEEE Cat. No.04TH8753), vol. 1, 2004, pp. 502–508 Vol.1.

[15] I. Ghafir, M. Hammoudeh, V. Prenosil, L. Han, R. Hegarty, K. Rabie, and F. J. Aparicio-Navarro, "Detection of advanced persistent threat using machine-learning correlation analysis," Future Generation Computer Systems, vol. 89, pp. 349–359, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X18307532

[16] C. D. Xuan, "Detecting apt attacks based on network traffic using machine learning," Journal of Web Engineering, vol. 20, no. 1, pp. 171– 190, 2021.

[17] T. Q. Nguyen, R. Laborde, A. Benzekri, and B. Qu'hen, "Detecting abnormal dns traffic using unsupervised machine learning," in 2020 4th Cyber Security in Networking Conference (CSNet), 2020, pp. 1–8.

[18] R. Kozik, M. Choras´, R. Renk, and W. Holubowicz, "Semi-unsupervised machine learning for anomaly detection in http traffic," in CORES, 2015.

[19] A. Zamir, H. Khan, T. Iqbal, N. Yousaf, F. Aslam, A. Anjum, and M. Hamdani, "Phishing web site detection using diverse machine learning algorithms," The Electronic Library, vol. ahead-of-print, 01 2020.

[20] Z. Rahman, X. Yi, and I. Khalil, "Blockchain based ai-enabled industry 4.0 cps protection against advanced persistent threat," IEEE Internet of Things Journal, pp. 1–1, 2022.

[21] (2020) Botsv1 dataset. [Online]. Available: https://github.com/splunk/botsv1

[22] (2020) Botsv2 dataset. [Online]. Available: https://github.com/splunk/ **bots**