ASTES

Advances in Science, Technology and Engineering Systems Journal Vol. 6, No. 2, 295-302 (2021)

www.astesj.com

ASTESJ

ISSN: 2415-6698

Improved Detection of Advanced Persistent Threats using an Anomaly Detection Ensemble Approach

Adelaiye Oluwasegun Ishaya[1,*], Ajibola Aminat[2], Bisallah Hashim[2], Abiona Akeem Adekunle[3]

[1]*Department of Computer Science, Bingham University, 961105, Nigeria*

[2]*Department of Computer Science, University of Abuja, 902101, Nigeria*

[3]*Federal Polytechnic, Ile oluji, Ondo State, 351101, Nigeria*

ARTICLE INFO

Article history:
Received: 29 October, 2020
Accepted: 30 December, 2020
Online: 17 March, 2021

Keywords:
Anomaly detection
Traffic analysis
Packet capture

ABSTRACT

Rated a high-risk cyber-attack type, Advanced Persistent Threat (APT) has become a cause for concern to cyber security experts. Detecting the presence of APT in order to mitigate this attack has been a major challenge as successful attacks to large organizations still abound. Our approach combines static rule anomaly detection through pattern recognition and machine learning-based classification technique in mitigating the APT. (1) The rules-based on patterns are derived using statistical analysis majorly Kruskal Wallis test for association. A Packet Capture (PCAP) dataset with 1,047,908 packet header data is analyzed in an attempt, to identify malicious versus normal data traffic patterns. 90% of the attack traffic utilizes unassigned and dynamic/private ports and, also data sizes of between 0 and 58 bytes. (2) The machine learning approach narrows down the algorithm utilized by evaluating the accuracy levels of four algorithms: K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Decision Tree and Random Forest with the accuracies 99.74, 87.11, 99.84 and 99.90 percent respectively. A load balance approach and modified entropy formula was applied to Random Forest. The model combines the two techniques giving it a minimum accuracy of 99.95% with added capabilities of detecting false positives. The results for both methods are matched in order to make a final decision. This approach can be easily adopted, as the data required is packet header data, visible in every network and provides results with commendable levels of accuracy, and the challenges of false positives greatly reduced.

1. Introduction

Information security challenges have been of great concern to Information Technology (IT) experts. These challenges involve the use of malicious techniques to get unauthorized access in an attempt to disrupt service, steal information and inflict harm amongst others [1] and [2]. The effect of a successful attack has moved from just affecting machines to posing a risk to human existence and wellbeing. In 2014, 7.2 million US dollars was the estimated cost of an attack to an organization [1]. In [3], the estimated annual cost of cyber security breaches for 2015 was 3 trillion US dollars and estimated to rise to 6 trillion by 2021.

In 2009, Advanced Persistent Threat (APT) a relatively new cyberattack method named based on its traits, was discovered and has been a serious cause for concern to information security experts. Advanced Persistent Threat's most accepted definition is the National Institute for Science and Technology's (NIST) definition. NIST defined it as the use of multiple attack vectors to perpetrate targeted attacks by a well-skilled expert, exposed to huge resources. These targeted attacks negatively affect organizations through the exfiltration of confidential information, creating access for future attacks amongst others. This type of attack is successful due to the actors persistence, metamorphosis and obfuscation [4] and [5].

Advanced Persistent Threat (APT) thus refers to a targeted threat continually and gradually transforming through obfuscation methods and multiple attack techniques and vectors thereby granting an unauthorized user undetected access and control of the target systems for an extended period of time [6] and [7]. This threat majorly targeting the network plane but classified as a multi-plane threat, continually goes through metamorphosis and rapidly spreads while persistently attempting to infiltrate the target organization. Due to the rapidly increasing growth in the fields of computing and networking, APT is increasingly drawing attention among security experts.

*Corresponding Author: Adelaiye Oluwasegun Ishaya, Bingham University, +2348031599692, oluwasegun.adelaiye@binghamuni.edu.ng

Gaining popularity in the first half of 2011 due to the high level of attacks to well-known organizations, Advanced Persistent Threats (APT) cases show that huge organizations including the financial organizations, military, chemical plants, energy and nuclear industries, education institutions, aerospace, telecommunication, and governments. APT attacks that occurred in 2011 tagged by the malware utilized include; Red October, Aurora, Duqu, Ke3chang, RAS breach, Flame, Stuxnet, Snow Man, and Mini Duke amongst others [8] and [9].

Earlier studies on APT attacks aimed at identifying the inherent characteristics of APT, characterize APT attacks into phases and present generic countermeasures in an attempt to militate APT [10]-[12].

Citing the ease of penetration through APT attacks and the results of evaluative studies on these attacks show the difficulty in detecting and preventing APT attacks. The gravity of APT is visible from the occurrence of high profile APT attacks and the exfiltration of sensitive data from highly recognized organizations like Sony, Citigroup, RSA security, NASA, FBI, Fox broadcasting, etc. [7] Research shows that traditional methods for securing data were used in these organizations, but the attacks were still executed un-prevented. Researchers have pointed out and scrutinized this challenge, which is to a great extent related to the failure in preventing and detecting targeted attacks using existing conventional techniques [13] and [14].

This failure has led to breaches involving confidential information and documents of organizations and government agencies. Existing methods have been ineffective in the fight against APT activities in the user, application, network and physical plane.

The continuing cases of these malwares bypassing existing security infrastructure, show that vulnerabilities and threats exist even in the midst of existing technical mitigation techniques. The solutions that utilized similar approaches suffered setbacks due to the occurrence of false positives.

This study is an extended paper titled “Mitigating Advanced Persistent Threats Using A Combined Static-Rule And Machine Learning-Based Technique” from the 15th International Conference on Electronics Computer and Computation (ICECCO) conference held in Abuja Nigeria in December, 2019 [15].

In this respect, this study proposes an ensemble anomaly detection technique combining static-rule based anomaly detection technique and an optimized ensemble machine learning algorithm. This study also assesses the efficacy of the proposed approach in mitigating APT attacks and reducing the occurrence of false positives. This approach provides a new easy mechanism for the detection and prevention of attacks to information systems.

2. Related Work

Mitigating Advanced Persistent Threats (APTs) has been a major concern for existing Intrusion detection and prevention systems. A lot of research work has been done in an attempt to provide a solution to APT like attacks. This section presents similar work done in militating the threat citing their success rates, effects and limitations.

In [12], [16]-[18] and proposed implementing traffic/data analysis using divergent techniques. This method was applied to detecting infected PDF and TIFF related files by [16] and embedded exe files in [17]. The major shortcomings were time delay and the need for human intervention. In [18] and [12] the authors suffered setbacks as the financial resources needed were high when compared to the impact on eliminating the threat. In [19], the author proposed the use of additional features to an open-source Intrusion Detection System. Additional features include data traffic analysis to detect malicious activity based on the state of the packet in transit and the port used, blacklist filters and the use of hash algorithm in protecting the integrity and confidentiality of the data within an organization.

A gene-based technique using patterns in detecting APT was employed in [20]. This approach identifies similitude with APT attacks using patterns from previous attacks. In [20], the author utilizes a network protocol behavioral pattern to form a gene-based detection system.

This work focuses on combining static-rule based anomaly detection technique and machine learning-based technique. Table 1 presents machine learning-based approaches to mitigating Advanced Persistent Threats.

Table 1: Related Works

Author	Method	Accuracy
[3]	Machine Learning technique using correlation analysis	84.8%
[21]	Random Forest Algorithm	99.8%
[22]	Simple Vector Machine	98.6%

In [3], the author adopted machine learning techniques using correlation analysis in an attempt to mitigate Advanced Persistent Threats. The machine-learning algorithm collects the output of other detection methods and classifies the Advanced Persistent Threats alerts. The results in [3] showed an accuracy of 84.8% in classifying malicious versus normal. In [21], the author used random forest algorithm to predict and detect APT achieved 99.8% accuracy. Their work showed high accuracy in properly classifying data traffic. A dataset of 1228 log events classified using Support Vector Machine algorithm showed an accuracy level of 98.67% [22]. Several machine learning algorithms have been proposed and applied to mitigating APT, but the most common algorithms used with APT detection and prevention are majorly: Simple Vector Machine (SVM), K-Nearest Neighbor (KNN), Decision Tree and Random forest [3], [14], [23] and [24]. This narrows down the machine learning algorithms to four for this study. Machine learning approach unlike the other methods proposed, showed a high level of effectiveness in mitigating APT even without prior knowledge of the attack vector utilized. The major challenge with the machine learning approach is the occurrence of false positives, which can be misleading.

Having presented and discussed related approaches to mitigating Advanced Persistent Threats as well as machine learning-based approaches in mitigating Advanced Persistent Threats and citing their shortcomings, the next section presents details on the materials and methods approach to mitigating APT.

3. Materials and Methods

This section describes procedures to be used in investigating and finding a solution to the research problem. This section also

evaluates the reason and relevance for the recommendation and application of techniques in identifying, collecting and analyzing information and data used in mastery and comprehension of the research problem. Hence, proving that the outcome of the research work is reliable, valid and reproducible.

In completing the research objectives the method to be used is broken down into two parts:

a. Static-Rule Based Anomaly Detection (Statistical Analysis).

b. Machine Learning Based Anomaly Detection

Figure 1 shows the steps and their relationships involved in completing the research objectives. The research plan is explained as follows:

1. Research Goals: The first part identifying research goals, this has been outlined in section one. The goal is to mitigate Advanced Persistent Threats.

2. Data Collection: The data to be used is secondary data. The dataset is used with both methods highlighted above which include the Static-Rule Based anomaly detection utilizing statistical analysis and the machine learning-based stages consisting of three sub-stages. A dataset local to the physical location of the researchers was not available as Nigeria is yet to identify or document any successful APT attack. The dataset utilized was obtained from Coburg University located in Germany. The dataset consists of network packet metadata. This dataset was developed by tracking network packets and documenting header details. This dataset consists of 1,047,908 instances and identified as Coburg Intrusion Detection Dataset (CIDD) [25].

3. Statistical Analysis/ Static-rule stage: In generating rules to detect anomalies, patterns are identified that can be used to filter data traffic and properly classify them. The approach used in obtaining these patterns is by statistical analysis. The outcome of the statistical calculations provides the basis for the formation of rules guiding the static-rule based anomaly detection model. Static-rule based anomaly detection utilizes finite sets of rules in anomaly detection. ALGORITHM 1 presents the procedure in sequence for the detection process.

3.1. Algorithm 1

Input $V = (a_i, b_i)$ *[metadata fields to be tested]*
Output: malicious traffic,
$P = \{p_1, p_2, \dots, p_n\}$ *[Metadata collected, where T is metadata of all traffic within network]*
Begin
Initialize $P = \{\ \} \in T$, $S = \{S_1, S_2\}$ *[where* S_1 *and* S_2 *represents the margin to differentiate between normal and malicious]*
For each p_i *where* $\{a, b\} \subseteq P$
$V \leftarrow \{a_i, b_i\}$
For each V (<, > or =) S_1*, V (<, > or =)* S_2 *and V (<, > or =)* S_3 *[*S_1*,* S_2 *and* S_3 *are predefined rules]*
If $D \leftarrow a_i = b_i$
Return D
End

ALGORITHM 1 provides a step by step process for detecting APT using static-rule based anomaly detection. V is gotten from the network traffic through sensors that capture traffic data, where the $V\,i$ (a_i, b_i) is used to detect anomalies for instance i. The response to the procedure is the identity of the malicious traffic D. $P = \{p_1, p_2, \dots, p_n\}$ is a subset of the entire traffic of the network collecting data for some traffic within the network. The initialization stage resets P to empty and the margins for S1 and S2 are set. The values for a, b for each i gotten from the metadata of each packet. Conditions for identifying the malicious traffic V (<,>or=) S_i are checked and stored in $D \leftarrow a_i = b_i$. D is flagged as malicious traffic, and alerts the administrator of malicious activity.

The implementation conditions required for the adoption of Static rule-based anomaly detection is evident as illustrated using the algorithm above and is proved using a statistical test for association. The hypothesis used as the basis for this test are given below

4. Research Hypothesis

H_0: There is no difference between normal and malicious traffic with respect to source port, destination port, packets and bytes.
H_1: There is a difference between normal and malicious traffic with respect to source port, destination port, packets and bytes.

4. Machine Learning: This is the second phase of this work. It is also used to improve accuracy and for greater effectiveness by combining both methods. Machine learning uses characteristics similar to that of humans who react based on knowledge to respond to events. This ML stage consists of 3 sub-stages which are recursive.

a. Data Cleaning: Misleading and inconsistent data that may affect the effectiveness of the machine-learning algorithm was removed using WEKA 3.8.3. The data affected constituted 0.06% of the whole dataset. The affected rows were deleted as other methods will likely introduce bias.

b. Feature Selection: Determining the effect of each feature on the final results is of utmost importance. When features do not contribute or contribute negatively to the final outcome, deactivating them is very necessary. This stage deals with selecting features of positive effect in achieving higher accuracy in Mitigating APT. Using univariate selection method, feature selection through Extra tree classifier and correlation matrix, flows showed no positive effect in classifying anomalous versus normal traffic. The conclusion is based on the Chi^2 score of 0.000000e+00 for the test of non-negative features see Figure 2, 0.00 score on the graph showing feature importance see Figure 3 and no correlation with other features see Figure 4.

c. Classification and testing for accuracy: Using four algorithms 80% of the dataset is used to train and 20% to test. The test is to check how well the algorithm can properly classify the instances of data so as to pick the most accurate and fastest. The best algorithm is optimized to improve its classification accuracy while checking instances of overfitting. These algorithms are: Simple Vector Machine, Random Forest, K Nearest Neighbor and Decision tree. Divergent methods are adopted by these algorithms in classifying events based on their similarities. The choice of these algorithms was based on the frequency of their utilization in APT like researches. [3], [14], [23] and [24].

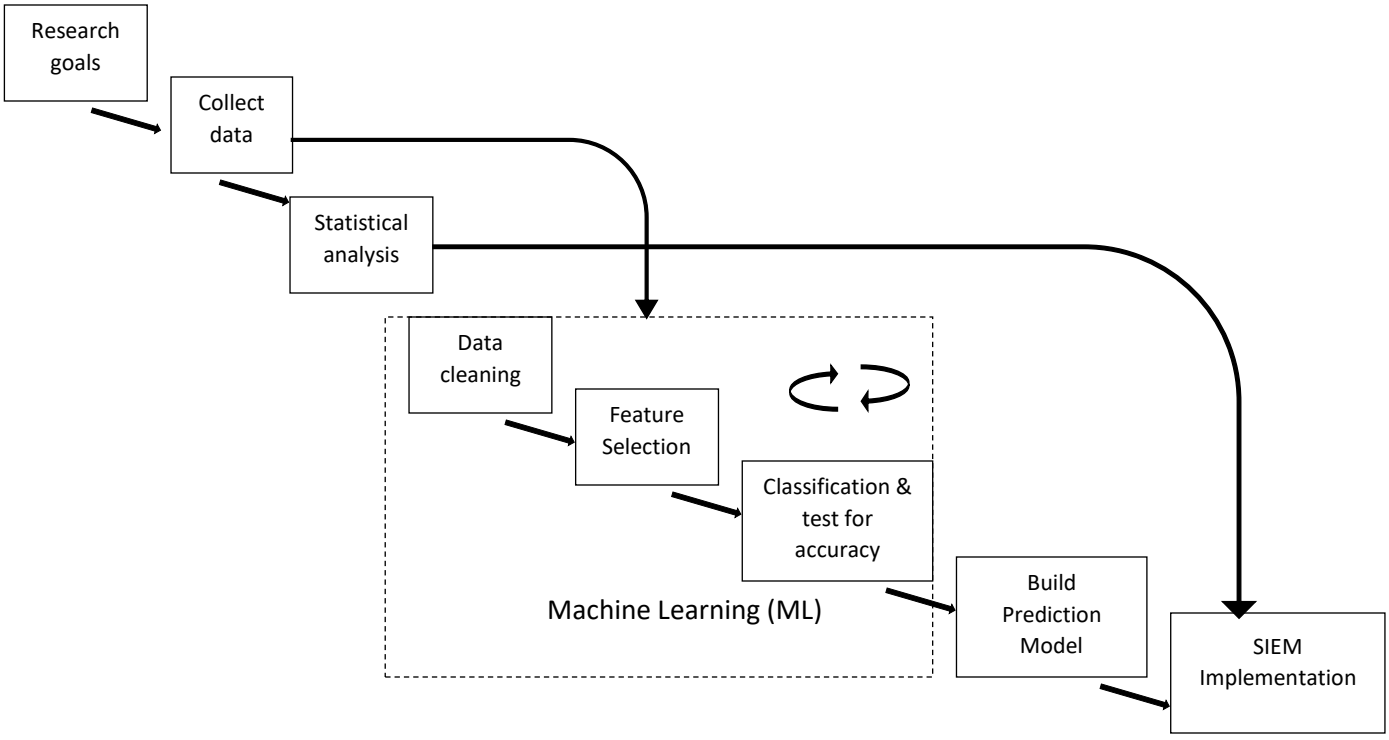

Figure 1: Research Plan

```
[1047908 rows x 7 columns]
      Specs          Score
4     Bytes  2.092691e+08
1    Src Pt  2.344792e+07
2    Dst Pt  1.646989e+07
6       Tos  1.224401e+06
3   Packets  5.388356e+04
0  Duration  1.355088e+04
5     Flows  0.000000e+00
```

Figure 2: Univariate Selection Results

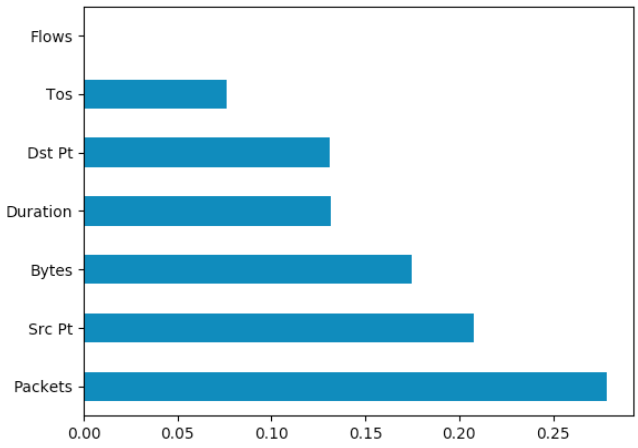

Figure 3: Graph showing Feature Importance

d. Build Prediction Model: The output of the classification stage is collected for developing the model. The algorithm selected is improved upon using load balancing due to the dataset distribution, as the instances of attack traffic are insignificant when compared to the instances of the normal traffic. The algorithm is also optimized using a trial and error approach.

e. SIEM Implementation: The final stage combines both approaches of the static-rule based anomaly detection and the enhanced machine learning-based prediction model, implementable as an SIEM module using REST API as seen in Figure 5. This stage presents the proposed solution in an attempt to mitigate Advanced Persistent Threats. The SIEM tool was considered as this solution is being implemented widely in organizations and has the capability of adding new features.

Figure 4: Correlation Matrix for the Dataset

The Dataset, which is in PCAP (Packet Capture) format, is feed at the top of Figure 5 as network traffic. The choice of using PCAP files is as a result of the information contained being easily extractible from the packet header during data transmission. The dataset is duplicated to feed both sides of the model. One copy feeds the left side of the model while the second copy feeds the right side. The left side of the model takes the dataset and the features are filtered to collect the statistically relevant data for

analysis, this happens at the defining variables stage. Using statistical analysis test for association, patterns to be used as a threshold for a static rule detection model can be gotten. These thresholds are set using rules S_1, S_2,…. S_n depending on the number of distinct patterns. With defined thresholds, new data traffic can be filtered based on the rules. The results of the filtering process are presented for decision-making based on the two methods. The results from the static rule-based anomaly detection method are presented in percentages based on how many attack patterns the captured traffic matches.

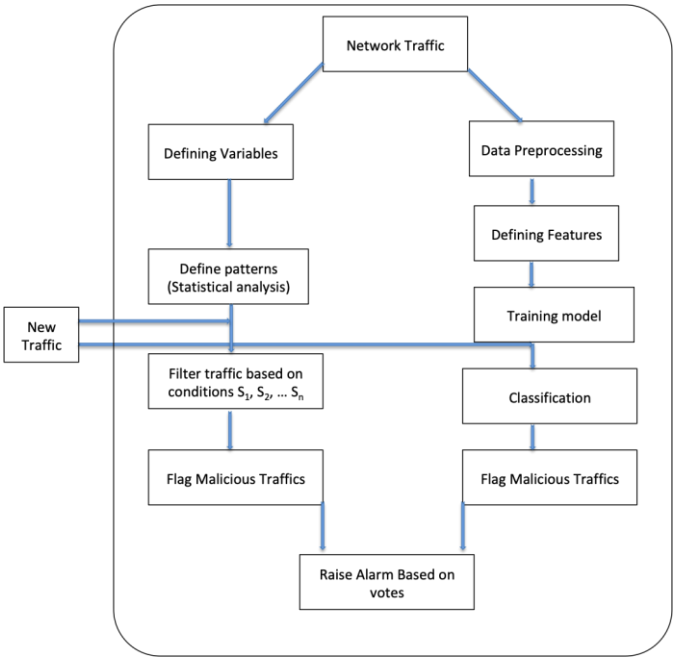

Figure 5: Proposed Model

The right side goes through the processes highlighted in 4 above. The three stages: data preprocessing/cleaning, feature selection and training the model are of importance to be able to accurately predict anomalous traffic. The model after these three stages can receive new traffic data and classify it accurately. The instance of the malicious traffic predicted is matched to the rule-based results to check if the results are true positives. The combination of static rule-based approach and machine learning is targeted towards reducing the occurrence of false positives.

5. Results

Finite set of patterns are obtained using Kruskal Wallis test for the implementation of static rule-based anomaly detection, following a normality test. Kruskal Wallis is a nonparametric test used to test for comparing k independent samples using population medians. The results are represented in

Table 2.

Table 2: Non-Parametric Test Results

Field	Median (Average rank Z)			Test statistic H	P-value
	Normal	Malicious (attacker)	Malicious (victim)		
Source Port	8082 (-27.27)	51357 (50.56)	2701 (-18.87)	2941.21	0.000
Packets	1.0 (98.14)	1.0 (-79.17)	1.0 (-63.76)	11357.29	0.000
Bytes	120 (173.47)	58 (-124.25)	54 (-121.05)	31357.13	0.000

1. **Source port:** p-value < 0.05. Reject H_0. There is evidence that at least 2 of the medians are different. Furthermore, the overall mean rank is significantly higher than both the normal and victim Z values (Z=-27.27<-1.96 & Z=-18.87<-1.96), and the Overall mean rank is significantly lower than the Z value for attacker (Z=50.56>1.96). From the results, there is significant evidence that the median port number for attacker is higher than that of normal and victim traffic.
2. **Packets:** p-value < 0.05. Reject H_0. There is evidence that at least 2 of the medians are different. Furthermore, the overall mean rank is lower than the Z value for normal (Z=-98.14>1.96), and the overall mean rank is higher than the Z value for both attacker and victim traffic (Z=-63.76<-1.96 & Z=-79.17<-1.96). There is significant evidence to show that the median number of packets for attacker and victim traffic is lower than the median number of packets for the normal traffic.
3. **Bytes:** p-value < 0.05. Reject H_0. There is evidence that at least 2 of the medians are different. Furthermore, the overall mean rank is lower than the Z value for normal (Z=173.47>1.96), and the overall mean rank is higher than the Z value for attacker and victim traffic (Z=-124.25<-1.96 & Z=-121.05<-1.96). There is significant evidence to show that the median size in bytes for victim and attacker traffic is lower than that of normal traffic.

From the results in 1, 2 and 3, having discovered patterns in network traffic vital in distinguishing between normal, victim and attack traffic we have the following rules.

Rule 1 (S_1): Flag traffic with source port within the unassigned and dynamic/private ports range.

Rule 2 (S_2): Flag traffic if size in bytes is between 0 and 58.

Rule 3 (S_3): Temporarily block and flag traffic that matches both S1 and S2.

The results from the statistical analysis done on the feature packets were dropped, as the results were not distinct. The predefined rules implemented in the static rule algorithm in section 3 number 3 to filter traffic and present suspicious traffic and match with the machine learning model results.

Having discussed the test results and highlighted rules based on patterns discovered for detecting malicious traffic, the machine learning approach and results of the modified Ensemble detection technique is presented.

The machine learning-based anomaly detection approach to militating APT attacks is due to the evidence that knowledge based on experience or previous events is of utmost importance in detecting mischievous activity without any interference.

Having prepared and cleaned the dataset, and selected the features to be utilized, the next phase is data classifying and accuracy and speed test. The algorithms to be used are all supervised learning techniques. Table 3 presents the results of the classification using four algorithms and the optimized ensemble algorithm for implementation in the model see Figure 2. The results are based on the accuracy in correctly matching instances and the time taken.

Table 3: Machine Learning Approach

S/No	ML Algorithm	Accuracy (%)	Time taken (Seconds)
1	K Nearest Neighbor (KNN)	99.74	693.34
2	Support Vector Machine (SVM)	87.11	359.95 (Dataset size 5,240)
3	Decision Tree (CART)	99.84	30.56
4	Random Forest (RF)	99.90	78.95
5	Optimized Random Forest	99.95%	(Dataset size 70,000)

Using KNN algorithm, the test data, which is 20% of the dataset set of 1048575 is 209582. The dataset consists of three classes namely: attacker, victim and normal traffic. The results show that 209037 of 209582 instances of data were properly classified. This gives a 99.74% accuracy.

SVM was initially unsuccessful in classifying due to the size of the dataset and would take an infinite duration of time in performing this task as the algorithm works better with small datasets. On reducing the size gradually to 5240 instances a classification using SVM was successful with an accuracy of 87.11% as seen in Table 3. This is done using a random function to select 0.5% of the instances randomly from the 1048575 dataset to remove fears of bias.

On using the Decision tree algorithm on the 1048575 dataset, the results from the confusion matrix and classification report provided by Scikit-learn showed that, 209246 instances out of 209582 were correctly classified. This gives a 99.84% accuracy. The RF algorithm gave an accuracy of 99.90%.

The RF Algorithm was selected and modified based on the high accuracy of 99.90% recorded. The modifications included load balancing of the dataset and modifying the formula for entropy. The justification for this approach was based on the fact that the dataset was unevenly distributed and also in an attempt to improve on the existing results as entropy provides the best conditions for splitting the trees. Algorithm 2 shows the load balance technique used.

5.1. Algorithm 2

```
Input:
  train_df = (Duration, Src_pt, Dst_pt, Packets, Bytes, Tos,
  Label),
  n_trees=5,
  n_bootstrap=70000,
  n_features=4,
  dt_max_depth=None
  Output: Randomly selected datasets of size = n_boostrap
  Begin
    For i in n_trees:
      train_df1= random select 200000 from train_df where
      Label=Attacker with replacement
      train_df2= random select 200000 from train_df where
      Label=Victim with replacement
      train_df3= random select 200000 from train_df where
      Label=Malicious with replacement
      train=train_df1+train_df2+train_df3
      train= random select n_bootstrap from train with
      replacement
  return train
End
```

From Algorithm 2, the CIDD dataset of size 1048575 is first broken down into 200,000 instances per label. The 3 labels providing a dataset of 600,000 randomly selected instances. This step evenly balances the dataset presenting 200,000 for Attacker traffic, 200,000 for Victim traffic and 200,000 for normal traffic. The dataset of size 600,000 is then reduced to 70,000 the bootstrap size per tree using random selection. The dataset of size 70,000 is used to build one tree in the forest. Data is selected randomly based on labels to load balance the dataset for each tree making the learning data better distributed. The second modification done on the algorithm is modifying the formula responsible for splitting the tree based on the best condition for a split. The calculation for entropy is presented in equation 1, and the modified formula in equation 2.

$$\text{Entropy } -= \sum p * \log_2(p) \quad (1)$$

The modified formula introduces a constant 10 as seen in equation 2.

$$\text{Entropy } -= \sum p * 10 * \log_2(p) \quad (2)$$

These two enhancements when implemented increases classification accuracy from 99.90% to 99.95% and also utilizing a smaller subset of 70,000 from the 1048575 dataset. The results in Table 4 show the decision-making table.

Table 4: Decision Making Table

Static- Rule algorithm (Careful or Jump)	Optimized Random Forest algorithm (Attacker, Victim or Normal)	Decision
Carefull (Malicious)	Attacker (Malicious)	Malicious
Jump (Normal)	Victim (Malicious)	Maybe
Jump (Normal)	Normal (Normal)	Normal

Storing the results of both algorithms in a data frame provides the platform in making a decision on whether to flag traffic or not. A conditional statement is applied to the data frame fields to make decisions based on the values from both fields. Table 4 provides a sample of results using a combined approach consisting of both static-rule based anomaly detection and machine learning techniques in mitigating APT.

6. Discussion

Mitigating Advanced Persistent Threats and reducing their effects to an acceptable risk level, has been an issue and threat to the existence of organizations, data as well as the safety of humans. The results from this study, show great accuracy levels and effectiveness using a hybrid approach comprising of both static and machine learning techniques in militating Advanced Persistent Threat attacks. The dataset utilized contains PCAP files classified into attacker, victim and normal traffic using divergent attack vectors per instance. These PCAP files hold packet metadata which is easily extractible by off the shelf and cloud applications like Wireshark and OpenStack. Following a hybrid anomaly detection approach, statistical analysis is used to test the dataset for association in the patterns using labels. The outcome indicated as shown in Table 3, that a high number of both the attacker and victim instances that are malicious, are often of a few bytes. Port numbers for attacker and victim traffic as also indicated in Table 3 often fall within the range of private/dynamic port numbers between 49152 to 65535. These patterns meet the requirements for implementing static-rule anomaly detection as they are finite. From algorithm 1, these rules can be implemented by replacing S_1, S_2……S_n with the conditions. This approach filters the data traffic and detects malicious activities labeling the traffic as either Jump(Normal) or Careful(malicious) as seen in Table 4. The aim is to combine two methods in an attempt to increase the accuracy level as well as reducing the chances of false positives thereby improving the effectiveness in mitigating APT. The combination of static rule anomaly detection with ML technique indicates positive results in greatly increasing the efficacy in the fight against APT as shown in section 4. The algorithm with the highest accuracy is picked based on the evaluation of multiple algorithms and considered the best algorithm for the prediction model using the PCAP dataset. The algorithms used are selected on their utilization in work done on mitigating APT. These algorithms when used on the PCAP dataset, also indicated a high level of effectiveness in mitigating APT attacks with 87.11%, 99.74%, 99.84% and 99.90% accuracy levels for SVM, KNN, CART and RF algorithm respectively as shown in Table 3. Choosing the algorithm with the best accuracy (RF) with the modifications applied increases the accuracy and effectiveness. The optimizing approach consists of dataset balancing and modifying the formula used in selecting the best point for tree branching. Accuracy increase from 99.90% to 99.95% is recorded after the modification, thereby improving the chances of effectively and accurately detecting APT attacks. Achieving a 99.95% accuracy in malicious traffic classification and a 90% chance of utilizing port numbers and bytes in detecting fraudulent traffic through static-rule based detection. The combination of these methods shows great results in providing highly accurate and an effective method in the detection, prediction and prevention of APT. This approach reduces false positives as the detection of APT traffic detected with both methods as shown in Table 4, proves that the threat is correctly identified (true positive) and hence the planned attack foiled. This result shows impressive effectiveness in militating APT when compared to the results in Table 2. This work leads [21], the best work also with 0.14%. Lower accuracies were recorded by the other algorithms. By this, our work provides the most effective method in militating APT asides the ability of the model to solve the problem of false positives. The outcome of the four algorithms selected had commendable accuracy levels but with the improved accuracy level using the modified algorithm improved the effectiveness making this the best method in militating APT.

The results of the model cannot be lower than the highest accuracy level recorded by the machine learning component of the model and hence, the accuracy level for the optimized Random Forest algorithm marks the minimum accuracy level for the model tagged at 99.95%. Implementing this model also ensures a 90% chance in accurately detecting APT traffic (True positives). This approach will reduce the chances of attackers using this method for economic sabotage, destruction of organizations reputation and cyberwars.

7. Conclusion

APT attack exploits grow every year with improved levels of sophistication and obfuscation. The challenge in effectively detecting and preventing APT attacks against large organizations and governments introduces a great risk in the loss of confidential and valuable information and services. Having looked into prevention and detection techniques, combining and modifying existing approaches on their effectiveness as well as integrate new methods is needed. From studies done, Anomaly detection proved the most promising existing mitigation method although, challenges with false-positive results occasionally occur. This study proposes a behavioral pattern recognition approach using statistical analysis to create a static rule-based model in an attempt to limit the chances of false-positives and improve the model's effectiveness in militating APT. The proposed combined model of static-rule based detection and machine learning-based techniques with an optimized algorithm, showed increased accuracy in militating APT. The ability to detect malicious traffic in 90% of the network data traffic using the static rule approach and an accuracy of 99.95% in detecting malicious traffic utilizing machine learning approaches, presents an optimized ensemble model for militating APT with greatly manageable and reduced occurrence of false-positives. The prediction model has a minimum accuracy of 99.95%, which is the accuracy level of a single component of the model. From the results, this approach to APT will solve to a large extent the challenges of cyber security experts and their fears about Advanced Persistent Threats (APT). This work provides a model for mitigating APT, implementation in organizations, testing and improvements in handling false positives are areas for further work.

Conflict of Interest

The authors declare no conflict of interest.

Acknowledgment

The support and goodwill of the following organizations are greatly appreciated:

1. Bingham University, Karu
2. University of Abuja, Abuja.

References

[1] O.I. Adelaiye, A. Showole, S.A. Faki, "Evaluating Advanced Persistent Threats Mitigation Effects: A Review," International Journal of Information Security Science, **7**(4), 159-171, 2018.

[2] M.M. Alani, M. Alloghani, Industry 4.0 and engineering for a sustainable future, Springer, 2019.
[3] I. Ghafir, M. Hammoudeh, V. Prenosil, L. Han, R. Hegarty, K. Rabie, F.J. Aparicio-Navarro, "Detection of advanced persistent threat using machine-learning correlation analysis," Future Generation Comput. Syst., **89**, 349-359, 2018, doi:10.1016/j.future.2018.06.055.
[4] N. Virvilis, B. Vanautgaerden, O. S. Serrano, "Changing the game: The art of deceiving sophisticated attackers," in Cyber Conflict (CyCon 2014), 2014 6th International Conference, 87-97, 2014, doi:10.1109/CYCON.2014.6916397.
[5] A. Ajibola, I. Ujata, O. Adelaiye, N.A. Rahman, "Mitigating Advanced Persistent Threats: A Comparative Evaluation Review," International Journal of Information Security and Cybercrime, **8**(2), 9-20, 2019, doi:10.19107/IJISC.2019.02.01.
[6] S. F. De Abreu, S. Kendzierskyj, H. Jahankhani, Cyber Defence in the Age of AI, Smart Societies and Augmented Humanity, Springer, 2020.
[7] M. Nicho, S. Khan, "Identifying Vulnerabilities of Advanced Persistent Threats: An Organizational Perspective," International Journal of Information Security and Privacy (IJISP), **8**(1), 1-18, 2014, doi:10.4018/ijisp.2014010101.
[8] I. Jeun, Y. Lee, D. Won, Computer Applications for Security, Control and System Engineering, Springer, 2012.
[9] K. Kimani, V. Oduol, K. Langat, "Cyber security challenges for IoT-based smart grid networks." International Journal of Critical Infrastructure Protection, **25**, 36-49, 2019, doi:10.1016/j.ijcip.2019.01.001.
[10] S. Singh, Y. Jeong, J.H. Park, "A survey on cloud computing security: Issues, threats, and solutions," Journal of Network and Computer Applications, **75**, 200-222, 2016, doi:10.1016/j.jnca.2016.09.002.
[11] A. Alshamrani, S. Myneni, A. Chowdhary, D. Huang, "A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities," in IEEE Communications Surveys & Tutorials, **21**(2), 1851-1877, 2019, doi:10.1109/COMST.2019.2891891.
[12] P.N. Bahrami, A. Dehghantanha, T. Dargahi, R.M. Parizi, K.K.R. Choo, H.H. Javadi, "Cyber kill chain-based taxonomy of advanced persistent threat actors: analogy of tactics, techniques, and procedures," Journal of Information Processing Systems, **15**(4), 865-889, 2019, doi:10.3745/JIPS.03.0126.
[13] G. Brogi, V.V.T. Tong, "Terminaptor: Highlighting advanced persistent threats through information flow tracking," in New Technologies, Mobility and Security (NTMS), 2016 8th IFIP International Conference On, 2016, doi:10.1109/NTMS.2016.7792480.
[14] G. Berrada, J. Cheney, S. Benabderrahmane, W. Maxwell, H. Mookherjee, A. Theriault, R. Wright, "A baseline for unsupervised advanced persistent threat detection in system-level provenance," Future Generation Computer Systems, **108**, 401-413, 2020, doi:10.1016/j.future.2020.02.015.
[15] O. Adelaiye, A. Ajibola, "Mitigating advanced persistent threats using A combined static-rule and machine learning-based technique," in 2019 15th International Conference on Electronics, Computer and Computation (ICECCO), 2019, doi:10.1109/ICECCO48375.2019.9043278.
[16] K. Chang, Y.D. Lin, Advanced Persistent Threat: Malicious Code Hidden in PDF Documents, 2014.
[17] I. Ghafir, M. Hammoudeh, V. Prenosil, "Disguised Executable Files in Spear-Phishing Emails: Detecting the Point of Entry in Advanced Persistent Threat," in 2nd International Conference on Future Networks and Distributed Systems, 2018, doi:10.1145/3231053.3231097.
[18] N. Virvilis, D. Gritzalis, "The big four-what we did wrong in advanced persistent threat detection?" in 2013 Eighth International Conference, 2013, doi:10.1109/ARES.2013.32.
[19] I. Ghafir, V. Prenosil, "Proposed approach for targeted attacks detection," in Advanced Computer and Communication Engineering Technology, 2016, doi:10.1007/978-3-319-24584-3_7.
[20] Y. Wang, Y. Wang, J. Liu, Z. Huang, "A network gene-based framework for detecting advanced persistent threats," in 2014 Ninth International Conference, 2014, doi:10.1109/3PGCIC.2014.41.
[21] S. Chandran, P. Hrudya, P. Poornachandran, "An efficient classification model for detecting advanced persistent threat," in 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2015, doi.:10.1109/ICACCI.2015.7275911.
[22] T. Schindler, Anomaly detection in log data using graph databases and machine learning to defend advanced persistent threats, Eibl, 2018.
[23] P.K. Sharma, S.Y. Moon, D. Moon, J.H. Park, "DFA-AD: a distributed framework architecture for the detection of advanced persistent threats," Cluster Computing, **20**(1), 597-609, 2017, doi:10.1007/s10586-016-0716-0.
[24] D. Moon, H. Im, I. Kim, J.H. Park, "DTB-IDS: an intrusion detection system based on decision tree using behavior analysis for preventing APT attacks," The Journal of Supercomputing, **73**(7), 2881-2895, 2017, doi:10.1007/s11227-015-1604-8.
[25] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, A. Hotho, "Flow-based benchmark data sets for intrusion detection," in 2017 16th European Conference on Cyber Warfare and Security (ECCWS), 361-369, 2017.