Botnet Detection using Machine Learning

Shamsul Haq
Department of Computer science and Information technology,
Central University of Jammu
J&K, India
s.haq266@gmail.com

Yashwant Singh
Department of Computer Science and Information Technology,
Central University of Jammu
J&K, India
yash22222k12@gmail.com

***Abstract*:** The small program to perform any type of malicious activity that may damage the system of the legal user automatically without legal users' knowledge is called a bot (bad bot). The network of bots under the control of a botmaster is called a botnet. It is a serious threat to information, communication, and economy etc. The interaction of devices to form a botnet are smartphones, computers, IoT systems whose vulnerabilities are exploited and so the security is breached to relinquish the control to bot controllers or third-party. On the basis of C2 (command and control) structure the botnet is considered as a centralized, decentralized or hybrid type, however, the architecture of botnet includes botmaster, C2 architecture, malicious activity detector, and target. In this paper, the network traffic is focused to explore and analyze the detection of a botnet. The different set of a sample of the dataset (e.g. ISOT, ISCX, and CTU-13) consists of the features of botnet traffic and normal traffic flow in networks. The various techniques are used for the detection of a botnet, however, comparing signature-based (classification approach) and anomaly-based (decentralized, p2p/clustering approach) is the main focus in this paper for accuracy in case of unknown signature detection (high positive rates) and known signature (low false positive rates) respectively. The machine learning plays an essential role in the detection and recognition of Botnets and is therefore explored in thispaper.

In this paper, the mean of the accuracy of k-means clustering and j48 classification approach (hybrid approach) is calculated while using the two random dataset partitions whose sum are always equal to the original dataset to compute the accuracy of low positive rates and high positive rates based on the percentages of "correctly and incorrectly instances". The comparative analysis of the three techniques i.e. clustering, classification, and hybrid approach suggest that the results of classification and clustering are constant on one end i.e. lower in case of classification and higher in case of clustering and on the contrary, results of our approach are varying in nature and gives the approximation in both the processes.

Keywords: *Botnet, P2P, Malicious activities, classification, and clustering*

1. INTRODUCTION

Based on the structure of the botnet, the main two structures of Botnet are centralized and decentralized. To extend the standards the other type of hybrid botnet is developed that is very effective. It is also said that according to the "command and control" structure of Botnets, they are classified as (IRC) Internet Relay Chat-based, HTTP (hypertext transfer protocol) based, DNS or P2P (peer to peer) based botnet[1][2]. The centralized botnet approach for the command and control structure used by a botnet is an old approach. In this approach, the distribution of botmaster command is performed via bot controllers to hide the identity of the real attacker however they are found as easier to detect. They have a single point of failure and therefore is disabled by removing the centralized botmaster after detection. P2P based botnet structures are not based on the C&C server. Botmaster communicates directly to a bot peer and then the commands sent by botmaster are spread in a botnet to other bots[3][4]. In this architectural approach single point failure does not affect botnet to be disabled and therefore are very difficult for defenders to find. It is obvious that p2p botnet is very hard to find however they have also weaknesses like Sybil attack [5]. Due to the reason of such weaknesses and hard to handle HTTP based Botnets are being widely used since around 2008[6]. The HTTP based

the various botnet detection approaches, machine learning

protocol is widely used in this botnet generation for the establishment of command and control structure and unravels their malware activity among the normal traffic.

Detecting bots are not an easy task in a network of zombies. The botnet detection is the big concern even if one of the bots is detected in the same network and there is an increase in botnet detection complexity if IoT devices like are controlled by bots. To detect such type of botnet, among algorithms based on classification (supervised approach) and clustering (unsupervised approach) play an impotent role in the detection of a botnet. The algorithms of machine learning usually learn data and give a data-based prediction. In any machine learning task, the two important concerns to select are considered as appropriate feature selection and the method of machine learning algorithm. In the feature selection process, the subset of features is chosen from all possible futures to represent the data to describe the behaviour/activity of bots. The selection of features is usually dependent on the type of the data that is being used for detection of a botnet. There are different machine learning techniques that are used for detection of the botnet by the detection unusual traffic, false human interaction etc. automatically to take control and refrain the activity before any damage occurs [7][8].

In this paper, the second part is about the literature survey of different datasets, supervised and unsupervised algorithms for botnet detection. The third part presents the proposed model based on machine learning with the explanation of the data flow diagram and the related tool and data. The fourth part is based on the implementation of an algorithm and to show the different graphs for accuracy. At last, a conclusion is displayed to outline the results of this exploration/ paper.

2. LITERATURE SURVEY

The botnet detection strategies mainly based on P2P, DNS or HTTP are viewed for analysis. However, variations/limitations are been detected when tested including multiple datasets up to expectation are no longer regarded within original/referred detection techniques. Whereas, machine learning based on traffic classification

978-1-5386-6026-3/18/$31©2018 IEEE

or clustering seems to be a higher usual approach[9][10]. There are various classification or clustering algorithms up to expectation are been used for detection over centralized botnet structures, decentralized structures or hybrid type regarding structures.

TABLE 2.1 - DATASET USED IN VARIOUS ALGORITHMS

References	[23]	[22]	[21]	[20]	[20]
Algorithm	C4.5	REP TREE	SVM	ANT COLLONY SYSTEM	K-MEANS
Dataset	ISOT	ISOT	ISOT	ISOT	ISOT
Detection Rate	98.7%	98.3%	97.8%	67.8%	82.1%
False Positive	0.04%	0.01%	5.1%	23.5%	2.4%

[11] BotoOnus is considered online unsupervised detection of botnet method without a prior knowledge of net of bots (botnet). It generally extracts the vectors in a network consisting of flow features at the periodic end.

[12] Botminer, the data mining frame uses the clustering approach for botnet detection. In both p2p and centralized structures it is assumed bots behave in a similar way and therefore similar communication patterns are saved.

[13] Peershark clustering framework for botnet detection uses both clusterings and supervised machine learning approach (random forest, decision tree etc.). The main aim of peershark to detect botnet is at the dormant or active mode of illegal activity.

[14] The measurement structures of high performance may additionally provide the improving detection speed of short-lived anomalies. There is an investigation of anomalies employing traffic measurements with time stamps of fine- grained. The paper promotes the new recognition algorithm called S3 that is utilized by Bayes Net.

[15] The entropy-based approach is used to realize network anomalies while considering distribution classes as

a) flow header features which includes IP addresses, flow-sizes, ports and b) Behavioural features such as degree distribution/IPs of the source with each host communicates. The dataset captured at the Carnegie Mellon University in 2005 consists of 2.5 billion flows are used to operate analysis for better perception to use entropy based strategy in anomaly detection. There the port and address distributions are observed to be correlated in their time series and detection capabilities. The metrics of behavioral whether in or out-degree and distribution of flow size furnish the detection capabilities that are different from other distributions.

[16] Developed an approach for the traits of DNS based traffic to have a look at the botnet. The traits consist of bandwidth, burst duration and packet timing for proof of botnet command and control activity.

[17] Proposed the strategy to botnet detection activities by means of classifying the behaviour the of network traffic using methods of computing device learning classification.

[18] Developed the approach of evaluation of three feature decision algorithms (consistency primarily based subset evaluation, correlation-based feature decision, and principal element analysis) using different techniques of machine learning to classify anomalies of network traffic.

[19] Developed an efficient approach to detect periodic behaviors of traffic in the network.

[20][21][22][23] The table 2.1 with the following references suggests out the survey of various algorithms that are used to find out the false positive and detection rates.[24] The dataset that was captured in the CTU University named as CTU-13 is another botnet dataset consisting generally of the 14 features among them certain features are mostly used for highest accuracy.

3. PROPOSED MODEL

The flow of the botnet contrasts from the regular traffics, so to utilize variations and choose the features we need to have a large amount of network data first initially to extract features and after that utilization hybrid structure of clustering and classification for precision/accuracy. To have a delegate network dataset, the different challenges strike utilizes diverse calculation for detection purposes. In case of the fitting dataset, we need normal, botnet, and background/ labeled data information. In this paper first, the dataset is focused on containing a distinctive set of features. The features are pre-processed, trained, tested and occurrence of cross-validation is developed by using open source WEKA tool. The signature-based detection consisting of the classification algorithms has low false positive rate however not valid to find all Botnets as is usually used for known signatures, whereas the anomaly based detection which can be used for the unknown signatures type also, however, has a high false positive rate. So on the basis of evaluation of false positive rates; we used the hybrid detection approach of classification and clustering to find out that whether there is an impact on false positives in anomaly/clustering based detection. The proposed model that is used for the detection of accuracy is shown in Fig 3.1.

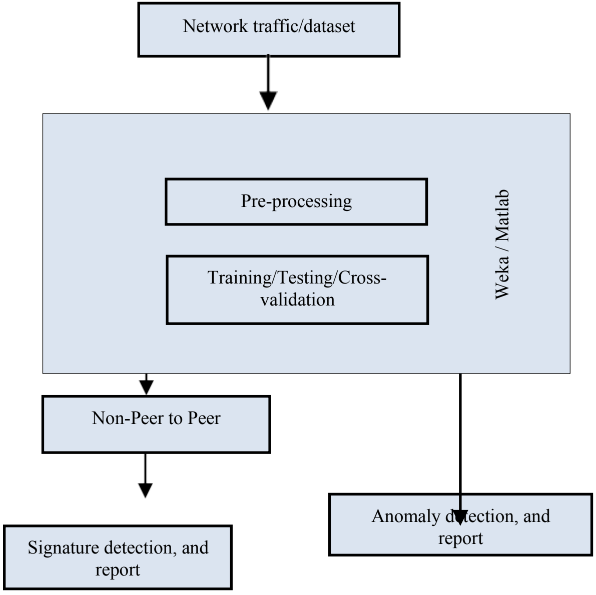

Network traffic/dataset
Pre-processing
Training/Testing/Cross-validation
Weka / Matlab
Non-Peer to Peer
Signature detection, and report
Anomaly detection, and report

FIG 3.1 - PROPOSED MODEL FOR BOTNET DETECTION

3.1 Network traffic/Dataset

The number of datasets like ISOT dataset consists of Zeus and Waledic dataset including bots obtained from the University of Georgia, US. The sample of the CTU-13 dataset that contains a number of features like source IP Address, Destination IP address etc. are extracted for the performance of botnet detection.

In this paper, the samples of datasets are used to pre-process the data. The samples of a variety of scenarios of CTU-13 are primarily used for pre-processing and accuracy in an open source WEKA tool. The loaded samples of datasets need to have the suitable features so to have the efficient results of classification and clustering algorithm with the exhibition of ROC curves also.

3.2 Weka

The WEKA tool is an open source tool while used in this paper for classification and clustering algorithms. The various types of classification algorithms such as Naïve Bayes Classifier, Ibk classifier, Rule Decision Table, Trees, and J48 classifier are compared on the basis of false positive rate and true positive rate to check out the efficient one. The survey papers of different clustering algorithms like as Hierarchical clustering, Self-organized map, k-means clustering etc. are compared, the one having lowest efficiency and the other having the high efficiency are mapped with rates of false positive. As in the case of classification, the training and testing datasets are needed or cross-validation can be used for the detection of “false positive rate and true positive rate”. The false and true positive rates for the clustering purposes show mostly the high positive rates.

The learning rates are applied to the samples of the dataset to develop 50%, 80%, and 70% dataset from the original existence and then the rates are taken out to each of the parts of original dataset (the original dataset is divided into any two parts and the sum of them is equal to the original one. E.g. 60% + 40% = 100%) while by the use of both classification and clustering sequentially or parallel.

The rates that are occurred from the classification and clustering are evaluated to find a mean. The mean is compared with the evaluation of the original sample dataset using anomaly detection/clustering algorithm mostly for p2p.

3.3 Algorithm

```
1.      Begin
2.          Identification of the dataset.
3.          Pre-processing of the data which includes
attribute selection.
4.          Perform clustering on the dataset and store result
in r11
5.          Perform classification on the dataset and store
result in r22.
6.          Set i = 30 and n=1
7.          if i <= 70 and i >= 30 repeat steps 8 to
12 8.       v1= i; v2= 100 – i
9.          r1= clustering (v1) i.e. perform clustering on v1
dataset and store its result in r1.
10.         r2 = classification (v2) i.e. perform classification
on v2 dataset and store its result in r2.
11.         mean[n] = % of incorrect
instances. 12.    i= i+10, n++;
13.         end if
14.         Compare mean[n], r11, r22 for analysis
15.     End.
```

The data flow diagram for the proposed model is shown in figure 3.2 where the loop is set to perform the hybrid-approach of classification and clustering to the respective partitions forming total dataset till the if statement is valid. Toward the end, there is a comparison of accuracies with respect to classification, clustering and hybrid approaches.

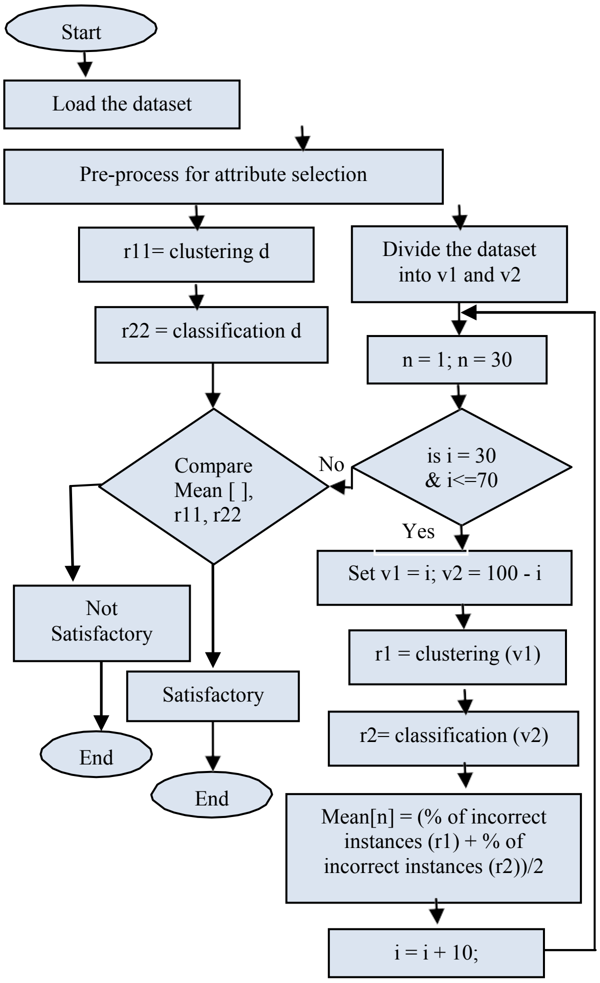


FIG 3.2 - DATA FLOW DIAGRAM OF THE PROPOSED APPROACH

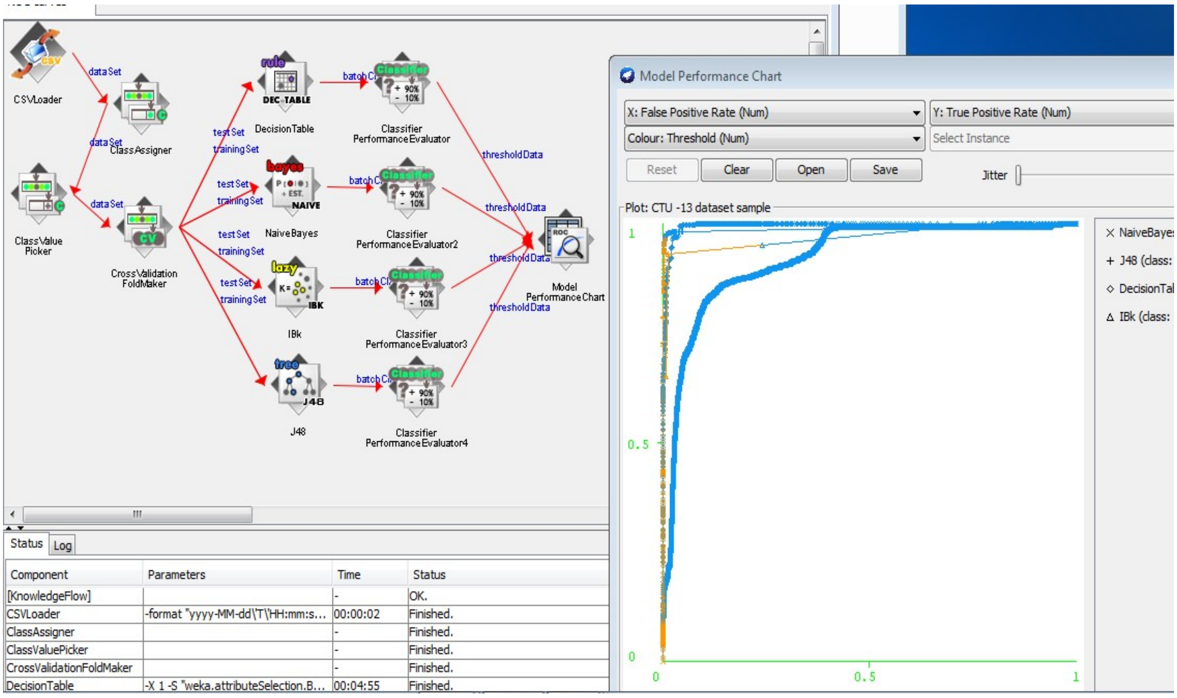
4. RESULTS AND DISCUSSION

4.1 Classification

In the first step of the experiment, we have implemented the “model performance chart” by knowledge flow of Weka tool. There are many interfaces of WEKA where many developments are performed, however, the knowledge flow interface plays an important role in the visualization of text viewer, strip chart, scatter plot matrix, model performance chart etc. In this knowledge flow interface the model for classification is developed with a) CSV loader, b) class value assigner, c) class value picker,
d) cross-validation fold maker, e) four classifiers, f) evaluation classifier, and g) match the graph in visualization respectively to show the ROC curves.

ROC curves are significant to find the reasonable accuracies. On the output, ROC curves of different classifiers are found to have variations in fig 4.1 and the variations of detailed accuracy are shown in table 4.1.

Among them, in this paper, it is found in accordance with samples of datasets (mostly of CTU-13) Trees J48 classifier has the higher accuracy and is therefore preferred for signature based. In this paper, it is sure that the J48



Component	Parameters	Time	Status
[KnowledgeFlow]		-	OK.
CSVLoader	-format "yyyy-MM-dd\'T\'HH:mm:s...	00:00:02	Finished.
ClassAssigner		-	Finished.
ClassValuePicker		-	Finished.
CrossValidationFoldMaker		-	Finished.
DecisionTable	-X 1 -S "weka.attributeSelection.B...	00:04:55	Finished.

FIG 4.1 – ROC CURVES OF CLASSIFICATION

-classifier is better to choose as a signature based classifier to have the appropriate results and so the low false positive rates for the parts of the chosen original dataset as well as for total dataset.

TABLE 4.1- OUTPUT PERCENTAGES OF CLASSIFIERS

CLASSIFIER	CORRECTLY CLASSIFIED INSTANCES	INCORRECTLY CLASSIFIED INSTANCES
NAÏVE BAYES CLASSIFIER	7538 = 19.4348 %	31248 = 80.56525%
IBK CLASSIFIER	33287 = 85.8222%	5499 = 14.1778 %
RULE DECISION TABLE CLASSIFIER	34050 = 87.7853 %	4736 = 12.2147 %
TREES J48 CLASSIFIER	35013 = 90.2723%	3773 = 9.7277 %

with samples of datasets (mostly of CTU-13) Trees J48 classifier has the higher accuracy and is therefore preferred for signature based. In this paper, it is sure that the J48 classifier is better to choose as a signature based classifier to have the appropriate results and so the low false positive rates for the parts of the chosen original dataset as well as for total dataset.

4.2 Clustering

As for anomaly or unsupervised based the various cluster based survey explains that the K-means clustering is also the support for the similar based botnet detection behaviors. Among the various unsupervised algorithm, the k-means is not the bad one to find accuracy. So in this paper, the k-means are directly chosen to find the results by means of accuracy. The detailed view of the evaluation using k-means clustering is shown in table 4.2.

The evaluated table 4.2 is based on as follows: a) The "run information" gives the view of the instances used in a dataset and the different types of attributes to be preferred.

TABLE 4.2 - OUTPUTS OF INCORRECTLY CLUSTERED INSTANCES

```
=== Run information ===

Instances: 114077

Attributes:  24

Proto, State, sTos, sHops, dHops, sTtl, dTtl, TcpRtt, SynAck, AckDat,

SrcPkts, DstPkts, SrcByte, DstBytes, SAppBytes, DAppBytes, Dur,

TotPkts, TotByte, TotAppByte, Rate, SrcRate, DstRate

Test mode:  Classes to clusters evaluation on training data

" Clustering model (full training set)"

kMeans

======

Number of iterations: 6

Within cluster sum of squared errors: 58215.57105464449

Final cluster centroids:

Cluster#

Attribute      Full Data        0          1

(114077.0) (9729.0) (104348.0)

==================================================

Time taken to build model (full training data): 2.3 seconds

" Model and evaluation on training set"

Clustered Instances

0       9729 (9%)

1     104348 (91%)

Incorrectly clustered instances:          62639.0       54.9094 %
```

TABLE 4.3- MEAN CALCULATION OF PERCENTAGE EVALUATED OF INCORRECTLY INSTANCES

KMEANS CLUSTER				J48 TREE CLASSIFIER				MEAN OF THE PERCENTAGE OF A AND PERCENTAGE OF B (%)
Total number of Instances		Incorrectly clustered instances/(A)		Total number of Instances		Incorrectly classified instances/(B)		
Number of instances	Percentage (%)	Number of instances	Percentage (%)	Number of instances	Percentage (%)	Number of instances	Percentage (%)	
34223	30	16969.0	49.5836	79854	70	7909	9.9043	29.74395
45630	40	22720.0	49.7918	68447	60	7052	10.3029	30.04735
57038	50	24321.0	42.64	57039	50(second half)	5847	10.2509	26.44545
57039	50 (second half)	27443.0	48.1127	57038	50	5759	10.0968	29.10475
68447	60	34207.0	49.9759	45630	40	4658	10.2082	30.09205
79854	70	26699.0	33.4348	34223	30	3552	10.379	21.9069

TABLE 4.4 - COMPARISON TABLE FOR THE COMPARISON GRAPH

MACHINE LEARNING APPROACHES	PERCENTAGE OF INCORRECTLY INSTANCES					
CLASSIFIER APPROACH (J48 TREE CLASSIFIER)	9.7277	9.7277	9.7277	9.7277	9.7277 %	9.7277
CLUSTER APPROACH (K-MEANS CLUSTER)	54.9094	54.9094	54.9094	54.9094	54.9094	54.9094
HYBRID APPROACH	29.74395	30.04735	26.44545	29.10475	30.09205	21.9069

b) The “test mode” classes the instances of around 114077 into 0 and 1class containing 9729 and 104348 respectively.
c) At the end of the evaluation, it is found at the “evaluation on training set” the incorrectly classified instances are 54.9094% (“the total dataset for evaluation is used”).

4.3 Hybrid approach

The hybrid approach based on a combination of classification and clustering for the calculation of accurate measurement is defined by the mean percentage of incorrectly classified instances and incorrectly clustered instances as shown in table 4.3.

The classified approach is evaluated by the j48 classifier and clustering approach by the k-means cluster while by using the variant partitions of the dataset forming the total dataset.

- Mean = (percentage of A + percentage of B) /2. Here the mean is occurred by using the distributed dataset samples. The samples of the dataset designed for the classifier (signature based/supervised algorithm) and cluster (unsupervised) in WEKA are such that the sum of their instances is equal to the total dataset.
- The percentage of incorrectly clustered instances calculated by k-means cluster = A.
- The percentage of incorrectly classified instances calculated by j48 classifier = B.

4.4 Comparison graph

The required outputs for the comparison are taken out by the following:

The classifier approach is chosen and the output is evaluated and taken out from the fig 4.1 (ROC curves) and table 4.1 respectively. The evaluation of k-means is shown in table 4.2, and the evaluated hybrid approach is shown in table 4.3. The outputs of the classifier, cluster and hybrid approaches are applied to draw the chart for a comparison of measurement.

Based on machine learning approaches that are implemented and because of their yield tables, table 4.4 is drawn in accordance of them by means of rates of incorrect instances and dataset partitions to draw the comparison graph (fig 4.2) of the cluster, classification, and hybrid approach.

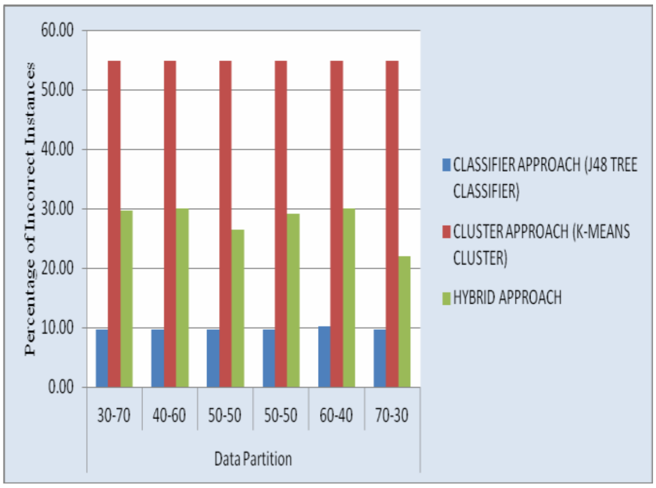


FIG. 4.2 - COMPARISON OF PERCENTAGES OF INCORRECT INSTANCES

Figure 4.2 analyses the approaches of signature (classifier), anomaly (cluster) and a hybrid approach. The signature-based/classifier approach shows the very lower percentage of incorrect instances and therefore for the well-known signature. There is a high positive rate percentage in anomaly approach; however, the hybrid shows the variations of incorrect instances but lower than an anomaly and higher than signature-based approach.

5. CONCLUSION

The bot can be defined as a malicious program that runs automatically once executed in a computer system. The net of bots is considered as a botnet that is controlled by a botmaster. The zombies are controlled remotely via an attacker under the command and control infrastructure named as botmaster or bot controller. On the basis of information security, it tries to exploit the vulnerabilities so to have loss of information, confidentiality, and authenticity. In the case of cybersecurity, it is able to provide any type of virus, Trojan, spamming emails, DDoS etc. Detecting bots are not an easy task in a network of zombies. The botnet detection is the big concern even if one of the bots is detected in the same network and there is an increase in botnet detection complexity if IoT devices like are controlled by bots. To detect such type of botnet, among the various botnet detection approaches, machine learning algorithms based on classification (supervised approach) and clustering (unsupervised approach) play an impotent role in the detection of a botnet. In this paper, a hybrid approach for detecting Botnets in the network has been devised. The approach consists of dividing the dataset into two smaller sets and then applies to cluster on one set and j48 classification on the other set. The result of both the processes i.e. clustering and classification are then combined on the whole dataset. The comparative analysis of these three techniques i.e. clustering, classification, and hybrid approach suggest that the results of classification and clustering are constant on one end i.e. lower in case of classification and higher in case of clustering and on the contrary, results of our approach are varying in nature and gives the approximation in both the processes.

In future scope, the results generated by the proposed hybrid algorithm are further needed to be implemented with the efficient data partitioning policy for better result generation, as the random data partitioning policy being implemented does not show many variations.

REFERENCES

[1] G. Gu and W. Lee, "BotSniffer : Detecting Botnet Command and Control Channels in Network Traffic BotSniffer : Detecting Botnet Command and Control Channels," 2008.

[2] M. Feily, A. Shahrestani, and S. Ramadass, "A survey of botnet and botnet detection," *Proc. - 2009 3rd Int. Conf. Emerg. Secur. Information, Syst. Technol. Secur. 2009*, pp. 268–273, 2009.

[3] C. R. Davis, S. Neville, J. M. Fernandez, J. M. Robert, and J. McHugh, "Structured peer-to-peer overlay networks: Ideal botnets command and control infrastructures?," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5283 LNCS, pp. 461–480, 2008.

[4] L. T. Borup, "Peer-to-peer botnets: A case study on Waledac," ^{Math}. *Model.*, 2009.

[5] D. Il Jang, M. Kim, H. C. Jung, and B. N. Noh, "Analysis of HTTP2P botnet: Case study waledac," *Proc. - MICC 2009 2009 IEEE 9th Malaysia Int. Conf. Commun. with a Spec. Work. Digit. TV Contents*, no. December, pp. 409–412, 2009.

[6] Z. Zhaosheng, J. F. Zhi, L. Guohan, R. Phil, C. Yan, and H. Keesook, "Botnet research survey," *Proc. - Int. Comput. Softw. Appl. Conf.*, pp. 967–972, 2008.

[7] ALLERIN, "Machine learning for anomaly detection | Artificial Intelligence |." .

[8] L. Mai and M. Park, "A comparison of clustering algorithms for botnet detection based on network flow," *2016 Eighth Int. Conf. Ubiquitous Futur. Networks*, pp. 667–669, 2016.

[9] M. Stevanovic and J. M. Pedersen, "An efficient flow-based botnet detection using supervised machine learning," *2014 Int. Conf. Comput. Netw. Commun.*, pp. 797–801, 2014.

[10] Z. Ghahramani, "Unsupervised Learning," *Adv. Lect. Mach. Learn.*, vol. 3176, pp. 72–112, 2003.

[11] M. Yahyazadeh and M. Abadi, "BotOnus An Online Unsupervised Method for Botnet Detection," *ISeCure*, vol. 4, no. 2, pp. 125–136, 2013.

[12] "64 BotMiner_ Clustering Analysis of Network Traffic for Protocol- and Structure-Independent Botnet Detection." .

[13] P. Narang, C. Hota, and V. N. Venkatakrishnan, "PeerShark: flow-clustering and conversation-generation for malicious peer-to-peer traffic identification," *Tijdschr. voor Urol.*, vol. 2014, no. 1, pp. 1–12, 2014.

[14] J. Kline, S. Nam, P. Barford, D. Plonka, and A. Ron, "Traffic anomaly detection at fine time scales with Bayes Nets," *Proc. - 3rd Int. Conf. Internet Monit. Prot. ICIMP 2008*, pp. 37–46, 2008.

[15] G. Nychis, V. Sekar, D. G. Andersen, H. Kim, and H. Zhang, "An empirical evaluation of entropy-based traffic anomaly detection," *Proc. 8th ACM SIGCOMM Conf. Internet Meas. Conf. - IMC '08*, p. 151, 2008.

[16] R. Villamarín-Salomón and J. C. Brustoloni, "Identifying botnets using anomaly detection techniques applied to DNS traffic," *2008 5th IEEE Consum. Commun. Netw. Conf. CCNC 2008*, no. 1, pp. 476–481, 2008.

[17] D. Zhao, I. Traore, A. Ghorbani, and B. Sayed, "Peer to peer botnet detection based on flow intervals," *Inf. Secur. …*, pp. 87–102, 2012.

[18] P. Narang, J. M. Reddy, and C. Hota, "Feature selection for detection of peer-to-peer botnet traffic," *Proc. 6th ACM India Comput. Conv. - Comput. '13*, pp. 1–9, 2013.

[19] B. AsSadhan and J. M. F. Moura, "An efficient method to detect periodic behavior in botnet traffic by analyzing control plane traffic," *J. Adv. Res.*, vol. 5, no. 4, pp. 435–448, 2014.

[20] K. Huseynov, K. Kim, and P. D. Yoo, "Semi-supervised Botnet Detection Using Ant Colony Clustering," *31th Symp. Cryptogr. Inf. Secur.*, p. 7, 2014.

[21] S. Saad *et al.*, "Detecting P2P botnets through network behavior analysis and machine learning," *2011 Ninth Annu. Int. Conf. Priv. Secur. Trust*, pp. 174–180, 2011.

[22] D. Zhao *et al.*, "Botnet detection based on traffic behavior analysis and flow intervals," *Comput. Secur.*, vol. 39, no. PARTA, pp. 2–16, 2013.

[23] P. Narang, S. Ray, C. Hota, and V. Venkatakrishnan, "PeerShark: Detecting peer-to-peer botnets by tracking conversations," *Proc. - IEEE Symp. Secur. Priv.*, vol. 2014–Janua, pp. 108–115, 2014.

[24] S. Buriya, A. K. Patel, S. S. Yadav, S. Buriya, A. K. Patel, and S. S. Yadav, "Botnet Behavior Analysis Using Naïve Bayes Classification Algorithm Without Deep Packet," vol. IX, no. Viii, pp. 45–54, 2015.