An advanced method for detection of botnet traffic using Intrusion Detection System

Manoj S. Koli
Department Of Computer Science and Engineering
Walchand College of Engineering
Sangli, India, 416415
Email: - manojkoli123@gmail.com

Manik K. Chavan
Department Of Computer Science and Engineering
Walchand College of Engineering
Sangli, India, 416415
Email: - manik.chavan@walchandsangli.ac.in

Abstract— **The botnet, which mainly consists of bots that are remotely controlled that provide the platform for most of the cyber threats. The effective countermeasure against such botnet is provided by IDS (Intrusion detection system). IDS regularly observes and identify the presence of active attack by inspecting the vulnerabilities in network traffic. A payload-inspection-based IDS (PI-IDS) recognizes active intrusion efforts by examining user datagram protocol packet (UDP) and transmission control protocol's (TCP) payload and matching it with known attacks .but the technique of PI-IDS is undermined if the packet is encrypted. The shortcoming of the PI-IDS is overcome by Traffic-based IDS (T-IDS), it does not check the packet payload; instead of this, it examines the header of a packet to classify the intrusion, but this technique is not suitable in today's world because network traffic grows rapidly so to check the header of each packet is not efficient and due to this detection rate also critical. So, We propose the new method in this paper T-IDS built an RDPLM (randomized data partitioned learning model) that depend on features set, and technique for feature selection, simplified sub spacing and multiple randomized meta-learning techniques. The correctness of our model is 99.984% and time for training is 21.38 s on the botnet dataset that is well-known. It is found that other Machine-learning models like deep neural network, reduced error pruning the tree detection task sequential minimal optimization, and random Tree.**

Index Terms— **Intrusion detection, machine learning, feature selection.**

I. INTRODUCTION

In today's world, most rigorous cyber-threat is botnet attack, basically, a botnet attack consists of a number of computers that are controlled remotely by the computer called bot-master [1] [3]. A botnet is a collection of huge number of bots all over the internet. Bot- master always try to keep botnet under its control and used it for different purposes they may be bad like DDOS attack, it also used to send the spam mail etc.

IDS is considered as the predictable model To improve the security of the network by knowing or identifying the intruders [4].the number of IDS built using the rule-based botnet because it works on the rules that are well-defined by the expert [5] [6]. It distinguishes botnet by comparing the normal network traffic and measuring it with the well-defined rules that are defined by experts [7] .but in today's world it is not possible to analyze the all network traffic because the network traffic increases rapidly due to this the rule-based IDS becomes slow and time-consuming.

II. RELATED WORK

In this section, we provide the details about the various Botnet detection technique. Commonly, botnet discovery methods are roughly divided into four categories: 1) rule-based; 2) anomaly-based; 3) domain name server (DNS)-based, and 4) data-mining-based [2].

In a first method that is rule-based IDS that monitors the payload of protocols that are TCP (transmission control protocol) or UDP (user datagram protocol) and correlates it with observed signatures of intrusion. If the volume of the data is less than the rule-based is accurate and efficient for identifying the activities that are malicious but if the data grows rapidly then to identify the attack or malicious activity is become complex or time-consuming. Snort is considered as the best example of rule-based method that is mostly used by the Bot-Hunter [7]

In the Anomaly-based method, the behavior of the normal network is abnormality from the anomaly behavior. This method also detects the novel attacks likely rule based. Normal networks are need to be modeled and it becomes difficult. DNS-based methods are developed on the basis that every bot must establish connection with Command &Control server to get register with the channel of botnet and communicate with the remaining bots or bot-master. As compared to the anomaly-based in a judgment botnets are detected by observing network traffic on server side of DNS and comparing it to the normal network profile.

III METHODOLOGIES

Our approaches are builds using four phases. To collect and generate the botnet dataset we have used the first phase that represents the traffic of real-world that is used to estimate review of an advanced method. To develop an ML algorithm that is scalable and effective that can deal with the large-scale network traffic and it could provide the time margin that is acceptable and it provide the real-time detection, the critically important part of this phase is to select the right features. To

reduce the dismissed and inappropriate data and to create the subset of relevant features we have developed a feature collection technique. Third, in extension to the previous phase, to reduce the number of samples of data that are used in the learning process we develop a technique for data reduction. In the fourth phase to develop the meta-learning model of multiple randomized trees, which is used to view the features that are randomly selected, is developed to detect the botnet attack.

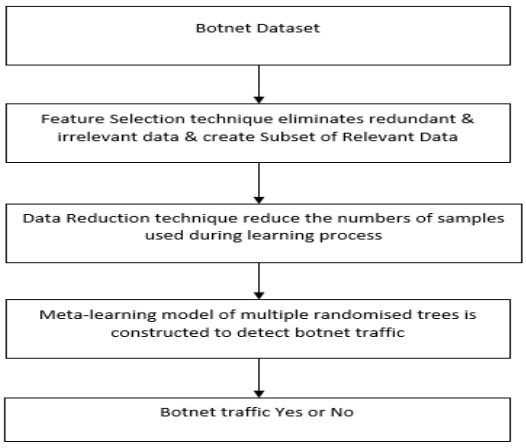

Fig.1 Overview of System

A. Data Collection: Botnet Benchmark-11Fs Dataset

To build the IDS of ML-based requires the dataset, that dataset used to by model to learn from that dataset. The dataset must consist of the all the real time network traffic that include all features. The dataset used by we to perform the experiment is ISOT that is the well-referenced dataset [8].The ISOT dataset consists of both traffic that is malicious and non-malicious, it mainly has the traffic of two P2P botnets namely Waledac and Storm from 2007 to 2009. French Chapter of Honeynet Project [10] is used to obtain the traffic that is malicious. Samely the non-malicious traffic is obtained from the TrafficLab at Ericsson Research in Hungary [11] and Lawrence Berkeley National Laboratory (LBNL) in 2005. We require the dataset that having the good number of traffic that is coming from an application that is used randomly, that include the various application such as browsing any web using HTTP, various games, and bit torrent clients.it also consist of traffic from enterprises like mid-size that is provided by LBNL.

B. Randomized Data Partitioned Learning Model (RDPLM)

Step I: (Modified Forward Selection Ranking): It is very important to select the right features of the botnet because it improves the abilities of the detection system and give proper judgment of problem [12].

There are mainly two methods for selection of features the first one is feature ranking (individual evaluation) and the second one is subset evaluation. The first method assesses the features from the dataset and according to the degrees of importance assign them weights [13].as compared to the first method second method select the subset of features based on some search method. To remove the jobless and unnecessary features the modified forward selection ranking technique is used, we remove the unnecessary features because it may contain the invalid relationship that could prevent the training process of the classifiers, As shown in fig.2., firstly we have to list out the features based on their importance and use of detecting the class type. With respect to the class type, information gain ratio is used to judge the use of each feature in the subsets of features that is provided by

$$\mathrm{IR}(C, A_j) = \big(H(C) - H(C|A_j)\big)/H(A_j)$$

Here, class is denoted by C and Aj is the jth attribute and entropy function is denoted by H (·).

$$H(\cdot) = -\sum_i P_i(\cdot)\log_2[P_i(\cdot)]$$

Dataset consist of many classes so i is the class number and P(·) is the probability operator then we have to sort the features in descending order meant to its use in feature set S1.then the feature that having the high rank make the new subset of features S2 from a feature set S1. next, likewise to the forward selection ranking [14], To measure the performance of the classifier when the S2 feature set is used, the algorithm adds the features from S1 to S2.ω is the Performance measure which is used to catch the complexity of time and the detection efficiency of a classifier estimated as follows:

$$\omega(S_i) = \mathrm{TBM}_{S_i} \times \big(100 - \mathrm{Acc}_{S_i}\big)$$

Here, TBMSi and AccSi used to indicate the time required to build the model and to detect the correctness of a classifier when the feature from set Si is used. here the main goal is to decrease the ω, the performance of the feature set Si is supposed to high than the Sj set if ω(Si) < ω(Sj). all the features are discarded regardless those who reduce the ω is put in S2; this task is done until the all the features from set S1 are tested, the selected feature set is S2 that consist of all the features. The pseudo code of step I is shown in fig.2

Given: $(x_1, y_1), \ldots, (x_m, y_m)$
where $x_i \in X, y_i \in Y = \{normal, anomaly\}$
- For $j = 1, \ldots, n$:
  Calculate $IR(C, A_j) = (H(C) - H(C|A_j))/H(A_j)$;
- Rank A_j based on its $IR(C, A_j)$ to obtain ranked feature set (S_1).
- Choose the first feature from S_1 to form feature set S_2.
- Intialize $S_3 = S_2$.
- **For $k = 2, \ldots, n$:**
  $S_3 = S_2$;
  Add next feature(k) from S_1 to S_3;
  Calcualte $\omega(S_3)$; \\the performance measure of S_3
  If ($\omega(S_3) < \omega(S_2)$)
  $S_2 = S_3$;
  $\omega(S_2) = \omega(S_3)$;
  End If
  End For

End

Fig.2 Pseudo code for feature selection

Step II: (Simplified Subspace): The dataset used by a botnet is large-scale and to deal with that type of dataset, a new technique for data reduction which uses other data partitioning, such as Voronoi-based and different clustering techniques are introduced. we develop the data reduction technique that mainly consists of two stages to cluster the entire vector space as input into smaller areas (Voronoi) based on some random samples. The main goal of this first phase is to reduce the complexity of further step. We have to use K-means clustering algorithm for retrieving the centroids of each the region that represent the legitimate input vectors. We made the new reduced dataset C that consists of all centroids.so due to this time required for creating the classification model is reduced. Fig.3 describe the actions performed by step2

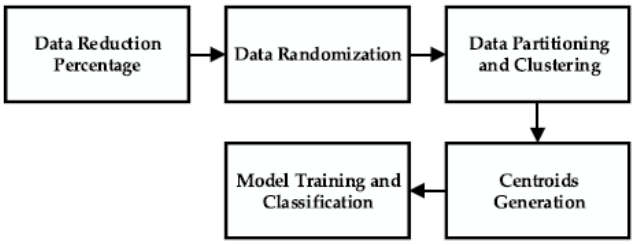

Fig.3: Steps for performing SRS

Firstly, we have to perform the sampling means simple random sampling (SRS) on the legitimate dataset to pick out n samples from a total of N samples. SRS select then samples randomly from the dataset with displacement following a binomial distribution .then we need to partition our dataset into n number of regions, based on selected samples which are representative. If we consider the partially labeled datasets, the set of representative samples selected, is supposed to be labeled. There are various algorithms to implement or construct the Voronoi diagram (VD), but Divide and Conquer and Fortune's Line Sweep give efficient results as compare to others [15]. So next, we have to cover the entire dataset that is original into the Voronoi regions that are constructed. Then we have to use K-means algorithm to detect the centroid of each region. The label of the selected subset followed the labels of the centroid, due to this the quantity of data is reduced to the threshold specified previously at a sampling percentage (n/N). As an effective and beneficial example, we used a random 2-D dataset to perform the reduction algorithm on it.

Step III: (Multiple Randomized Trees): As we discussed earlier our ISOT dataset consist of many types of data by various means like HTTP browsing and gaming, so using only one classifier is not the sufficient and beneficial as well. So we constructed a model that combined the set of methods to use network traffic by its characteristics. If we get a large collection of weedy learners, then the performance is not better, then we combine them together now it is possible that the learner can perform well[16].

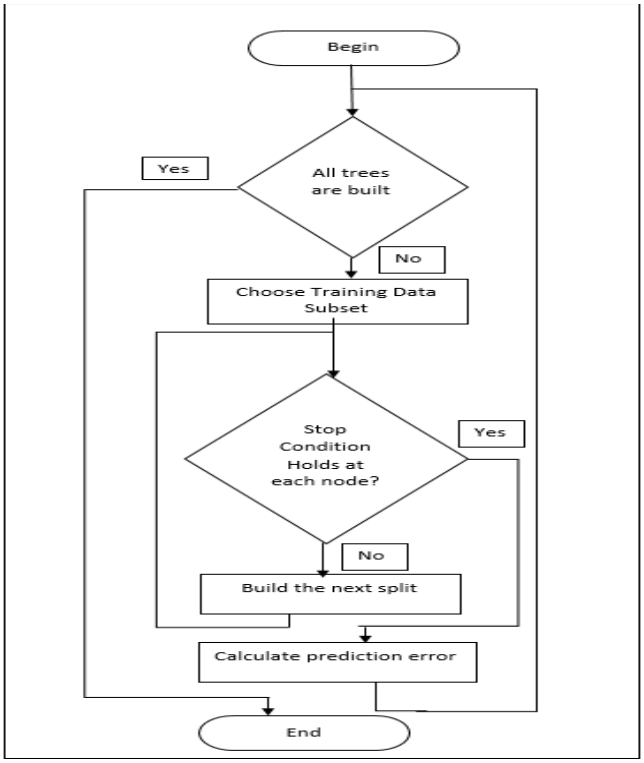

Fig.4 Flowchart of step III

IV EXPERIMENTS

In this section we have implemented the RDPLM is an advanced method for botnet detection. We implement this method using JAVA programming language. Below table 1 shows the selected 11 unique features from the original set of 41 features in the interlude of 300s based on the behavior of popular protocols and well-known behavior of botnets.

TABLE I
FEATURES DESCRIPTION

No	Name	Description	Type
1	SrcPort	Flow source port.	Disc.
2	DstPort	Flow destination port.	Disc.
3	Proto	Transport layer protocol or mixed.	Disc.
4	APL	Average payload packet length for time interval.	Cont.
5	PV	Variance of payload packet length for time interval.	Cont.
6	PX	# of packets exchanged for time interval.	Disc.
7	PPS	# of packets exchanged per second in time interval T.	Cont.
8	FPS	The size of the first packet in the flow.	Disc.
9	TBP	The average time between packets in time interval.	Cont.
10	NR	# of reconnects for a flow.	Disc.
11	FPH	# of flows from this address over the total number of flows generated per hour.	Cont.

There are various measures that we use for evaluating the classifier's performance including classification accuracy (Acc), false alarm rate (FAR), DR (also known as, sensitivity), Mac, standard deviation of F-measure (σ), time taken to build a model (TBM) and (TT). Fig.3 display the information gain ratio that is calculated. From the top of the image it displays

the information that contained in the packet and last it display the calculated Information gain ratio.

```
Tcp:  ******* Tcp offset=34 (0x22) length=28
Tcp:
Tcp:          source = 1036
Tcp:     destination = 80
Tcp:             seq = 0xD872D522 (3631404322)
Tcp:             ack = 0x0 (0)
Tcp:            hlen = 7
Tcp:        reserved = 0
Tcp:           flags = 0x2 (2)
Tcp:                   0... .... = [0] cwr: reduced (cwr)
Tcp:                   .0.. .... = [0] ece: ECN echo flag
Tcp:                   ..0. .... = [0] ack: urgent, out-of-band data
Tcp:                   ...0 .... = [0] ack: acknowledgment
Tcp:                   .... 0... = [0] ack: push current segment of data
Tcp:                   .... .0.. = [0] ack: reset connection
Tcp:                   .... ..1. = [1] ack: synchronize connection, startup
Tcp:                   .... ...0 = [0] fin: closing down connection
Tcp:          window = 65535
Tcp:        checksum = 0xAA1E (43550) [incorrect: 0xAA1C]
Tcp:          urgent = 0
Tcp:
Tcp: + NoOp: offset=25 length=1
Tcp:            code = 1
Tcp:          length = 1 [implied length from option type]
Tcp:
Tcp: + MSS: offset=20 length=4
Tcp:            code = 2
Tcp:          length = 4
Tcp:             mss = 1460
Tcp:
Tcp: + SACK_PERMITTED: offset=26 length=2
Tcp:            code = 4
Tcp:          length = 2
Tcp:
Information Gain Ration-1.0021606868913213
```

Fig.5 Calculated Information Gain ratio

Fig.6 (a) shows the accuracy of the previous algorithm and our proposed algorithm

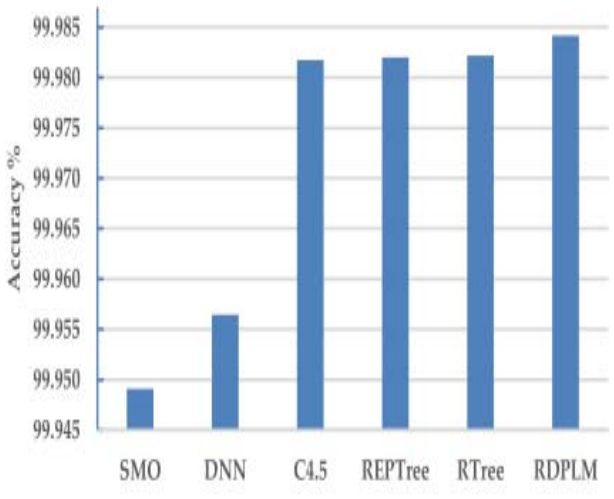

Fig.6 (a)

Fig.6 (b) shows the time required for building the model of previous methods and proposed.

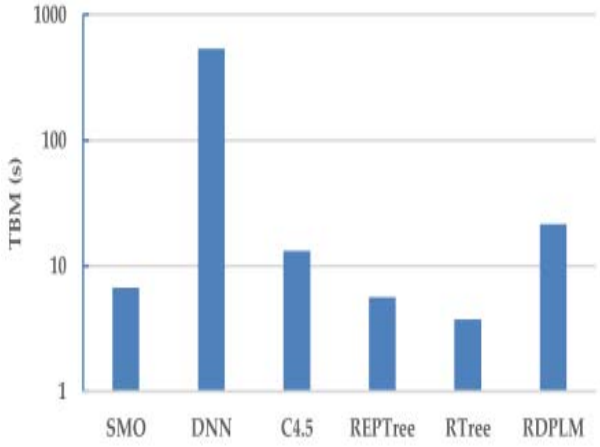

Fig.6 (b)

Fig.6(c) f measure of the algorithms

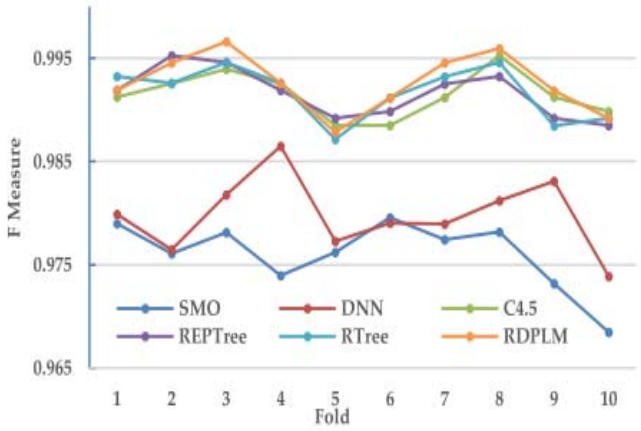

FIG.6(C)

Fig.6 (d) shows the false alarm rate of previous algorithm and our proposed RDPLM algorithm

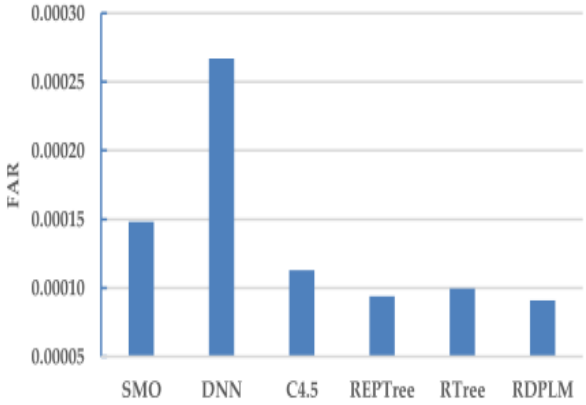

Fig.6 (d)

V CONCLUSION

This paper introduces the basic idea about the botnet and IDS that provide a chance to improve the security of existing network security by detecting, identifying and tracking the attackers. There are various existing IDS are available and that are used by ML algorithm but when you deal with large scale dataset complexity grows exponentially.

So, in this paper, we introduced three new steps for constructing the botnet IDS for networks.to deal with the unnecessary features and to reduce the size of the dataset, we used feature ranking and Voronoi clustering technique. In addition to that, we also use the characteristics of the network flow to detect the botnet intrusions despite the packet payload content, which helps it to encryption of packet. The result shows that the proposed method is produced very high detection accuracy (99.984%) also it reduce the computational cost of the model. This is used to prevent the processing delays of data, and this is the important requirement of ML-based IDS when dealing with the large scale of networks.

ACKNOWLEDGEMENT

I would like to thank my Guide Mr. M. K. Chavan for their continuous help and encouragement. Also a word of appreciation for Department of CSE, Walchand College for providing useful resources.

REFERENCES

[1] M. Abu Rajab, J. Zarfoss, F. Monrose, and A. Terzis, "A multifaceted approach to understanding the botnet phenomenon," in Proc. 6th ACM SIGCOMM Conf. Internet Meas., New York, NY, USA, 2006. pp. 41–52

[2] M. Feily, A. Shahrestani, and S. Ramadass, "A survey of botnet and botnet detection," in Proc. 3rd Int. Conf. Emerg. Security Inf., Syst. Technol., SECURWARE, Athens, Greece, 2009, pp. 268–273.

[3] B. Al-Duwairi and L. Al-Ebbini, "BotDigger: A fuzzy inference system for botnet detection," in Proc. 5th Int. Conf. Internet Monitor. Prot. (ICIMP), Barcelona, Spain, 2010, pp. 16–21

[4] D. E. Denning, "An intrusion-detection model," IEEE Trans. Softw. Eng., vol. SE-13, no. 2, pp. 222–232, Feb. 1987

[5] J. Zhang and M. Zulkernine, "Network intrusion detection using random forests," in Proc. PST, St. Andrews, NB, Canada, 2005, pp. 53–61

[6] M. Roesch, "Snort—Lightweight intrusion detection for networks," in Proc. USENIX LISA, Nov. 1999

[7] G. Gu, P. A. Porras, V. Yegneswaran, M. W. Fong, and W. Lee, "BotHunter: Detecting malware infection through IDS-driven dialog correlation," in Proc. USENIX Security, Boston, MA, USA, 2007, pp.

[8] S. Saad et al., "Detecting P2P botnets through network behavior analysis and machine learning," in Proc. 9th Annu. Int. Conf. Privacy, Security Trust (PST), Montreal, QC, Canada, 2011, pp. 174–180

[9] S. T. Sarasamma and Q. A. Zhu, "Min–max hyperellipsoidal clustering for anomaly detection in network security," IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 36, no. 4, pp. 887–901, Aug. 2006.

[10] (Jan. 2015). French Chapter of Honenynet. [Online]. Available:

[11] http://www.honeynet.org/chapters/france.

[12] G. Szabó, D. Orincsay, S. Malomsoky, and I. Szabó, "On the validation of traffic classification algorithms," in Passive and Active Network Measurement. Berlin, Germany: Springer, 2008, pp. 72–81

[13] F. Zhang, P. P. K. Chan, B. Biggio, D. S. Yeung, and F. Roli, "Adversarial feature selection against evasion attacks," IEEE Trans. Cybern., to be published

[14] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," J. Mach. Learn. Res., vol. 3, pp. 1157–1182, Mar. 2003

[15] S. Zaman and F. Karray, "Features selection for intrusion detection systems based on support vector machines," in Proc. 6th IEEE Consum. Commun. Netw. Conf. (CCNC), Las Vegas, NV, USA, 2009, pp. 1–8.

[16] F. Aurenhammer and R. Klein, "Voronoi diagrams," Handbook of Computational Geometry, vol. 5. Amsterdam, The Netherlands: Elsevier North Holland, 2000, pp. 201–290. J. R. Quinlan, "Induction of decision trees," Mach. Learn., vol. 1, no. 1,pp. 81–106, 1986.