Tree Based Classifiers for Distributed Denial of Service Attack Classification in Biotech and Science as a Service

Velliangiri S.[1]* and Premalatha J.[2]
1. Department of CSE, Sri Venkatesa Perumal College of Engineering and Technology, Puttur, INDIA
2. Professor, Department of IT, Kongu Engineering College, INDIA
*velliangiris@gmail.com

Abstract
Securing data is the basis of information society. In Cloud network security is easily prone to attacks and the number of intrusions has increased over the last decade. To hamper such efforts, Intrusion Detection System (IDS) plays a significant role in protecting systems and networks. The foremost problem in protecting wide range high-speed networks is detecting suspicious abnormalities in network traffic patterns caused from Distributed Denial of Service (DDoS) attacks. DDoS deliberately attacks with the intention to exploit victim's bandwidth or interrupt genuine users' access to services. In this work, the tree based classifiers for DDOS attack classification is proposed.

The algorithms used for constructing decision trees are formulated by putting together the most well-known and extensively used of all machine learning methods. Simulations were carried out using cloud as software-as-a-service for genome sequencing. The web based application allows replication and share big data of genome sequencing. Simultaneous attacks were simulated and launched. The experimental results obtained show the effectiveness of tree based classifier.

Keywords: Intrusion Detection Systems (IDS), Distributed Denial of Service (DDoS), Classification and Regression Tree (CART), C4.5 and Random Tree (RT).

Introduction
Cloud computing has provided numerous benefits but on the other hand there is a considerable compromise on the stability and security of the system connected to it. Widely used static defence mechanisms such as firewalls and other network security features provide only a tolerable level of security but for better level of security, dynamic systems such as IDSs should be used. Intrusion detection is a technique where the various events happening in a computer system or network are analyzed for any sign of intrusion. Denning (1987) in order to detect, identify and trace intrusion put forth the idea of Intrusion Detection System which comes as a set of software and hardware. This system mainly works to gather and analyse network traffic and detects for any malicious patterns and warns the concerned authority. Given below are the key functions of IDS.[1]

- Monitors and analyses the data collected both from user and system activities.
- It analyses the system configuration and evaluates file and system integrity.
- For static records, it discovers the anomalous patterns.
- For recognizing anomalous patterns, static records are utilized.
- It alerts the system administrator.

Further, intrusion detection methods can be categorized into four namely: anomaly detections, misuse detections, specification-based detections, as well as model-based detections. In anomaly detection, the system establishes a standard profile to user and system activity to monitor any undesirable changes happening to the user profile with respect to the normal pattern. Misuse detection, detects intrusions that tracks significant attack patterns that exploits the weakness in system as well as application software. With the specification-based detection the exact characteristics of critical objects are abstracted as well as crafted in a manual fashion according to security requisites. Then it is evaluated against the actual behaviour of the objects. An object being intruded behaves in an abnormal manner which is easy to detect without any formal understanding. The model-based intrusion detection evaluates the execution of a process against its program model to identify any intrusion attempts. [2]

IDS can be categorized into three types: Host based IDS (HIDS): It is positioned on one machine such as server or workstation. The information is collected from several sources and is assessed within the machine. HIDS is capable of using both anomaly as well as misuse detection systems. Network based IDS (NIDS): NIDS are employed on strategic pointing network architecture. The NIDS is proficient in the capture as well as analysis of data to identify familiar assaults through comparison of patterns or signatures of the database. It also reports any detected anomalous activity by scanning traffic. NIDS are typically called "packet-sniffers", as it efficient in capturing the packets sent via the communication networks. Hybrid based IDS: It manages alerts from both network as well as host-based intrusion detection devices providing a logical complement to NID as well as HID-central intrusion detection management.

Attacks are broadly categorised into following two types.[3]

Passive Attack: A passive attack looks for traffic that is not encoded, clear-text passwords and vulnerable information that can be captured easily and used elsewhere in another attack. The various methods in passive attack includes examining network traffic, decrypting contents in traffic that are weakly encrypted, monitoring unprotected communications, and capturing authentication information like passwords. For hackers, this way of interrupting the network traffic makes it easier to watch or predict the future actions of the user. In Passive attack, important facts and figures of the user is revealed to the attacker.

Active Attack: Through active attack, the hacker directly tries to invade or intrude protected networks. Viruses, Trojan horses, worms, or stealth is used for the attack. An active attack is where protection systems are directly attacked by introducing malicious code where it modifies or steals the information. These attacks are primarily aimed at network backbone, exploiting information in transmission or attacking genuine remote user while connecting to an enclave. Active attacks have serious consequences like revealing or distributing data files, altering data or Denial of Service (DOS).

DDOS attacks pose a severe risk to the Internet and to prevent such problem many security methods are being proposed. The hackers are constantly in search of new tools to break in security systems and likewise even the researchers are modifying their systems to combat the attack. The DDOS is rapidly changing into a complex environment and has reached a point where it is difficult to see the forest for the trees. DDOS has become a complex phenomenon with increasing number of attacks. Multitudes of known attacks have complicated the problem making it hard to decipher and resolve it. On the other hand, existing defence systems have deployed wide range of strategies to handle the attacks. This in turn makes it difficult to assess their effectiveness when compared to each other and also to determine its cost.

A DDOS attack is made of four components. First, it identifies a target host for the attack. Then the daemon agents which are actually agent programs are involved to attack the chosen target. Attack daemons are generally deployed on host computers but they tend to affect both the host as well the target computers. The attacker has to get access and intrude the host computer to employ these attack daemons. The most important part of a DDoS attack is the control master program. It coordinates the attack. Finally, the actual attacker hacks the system using the control master program. Below are the steps in a DDOS attack:

- An "execute" message is sent to the control master program by the hacker.
- The control master program on receiving the "execute" command triggers the attack daemons under its control.
- Lastly the attack daemons execute the attack process on the victim.

Several researchers' have categorized DDoS attack on a wide range.[5] They are:

Attack on Bandwidth: User Defined Protocol (UDP)/ Internet Control Message Protocol (ICMP) are flooded with attacks. This creates congestion or overloading of network link by transmitting more number UDP/ICMP and SYN-flooding packets. Distributed Reflected Denial of Service (DRDoS) spoofs IP address to send fake request to loads of computers. The targeted victim, like an organization server receives all the responses for the request sent.

Attack of Host Resource: The web server is targeted in such attacks where many connections to the server are kept open and are put on hold as long as possible to bring down their service availability. In some attacks huge amount of requests are sent to disable the victim's website. Slowloris DoS Hyper Text Transfer Protocol (HTTP) and HTTP GET Flooding attack are some types of attack. In HTTP Flooding attack, the attackers send huge quantities of HTTP flood attacks from several machines concurrently. The attack frequently requests downloading of target website's pages and results in DoS state.

Attack on System/Application Weakness: Ping of Death is a type of attack which cripples network resources on the basis of a defect in the Transmission Control Protocol/Internet Protocol (TCP/IP) suite. The maximum size of a packet is 65, 535 bytes. If a user sends more than the allowed size of the packet, the receiving computer will crash down.

It is easy to spoof an identity on the network. So DDoS takes advantage of the fact and has many strategies are based on it. Identity in this context refers to finding a particular machine on the network. Internet Protocol (IP) address is a unique identifier within the internet that is used to identify a specific machine on the internet similar to that of phone numbers within the phone system. In Attacks like Domain Name System (DNS) amplification spoofing of IP address is done to impersonate as the target of a request (similar to using a victim's phone number as a call back number in the voice mail of various other phone numbers to flood the victim with phone calls). As linking IP address to real life individual is difficult, tracing DDoS attacks gets complicated. Internet is global and so even once a hostile IP address has been traced to the Internet Service Providers (ISPs) who controls it, local laws may make it difficult to prosecute an attacker. [6]

The classification task assesses a function or creates a relation between a dependent parameter as well the contingent independent parameters through mapping of data points. A classification issue is simply identifying an object as being part of a particular class and these classes are made predefined and non-overlapping in a classification problem before applying any algorithm. Classification algorithms tend to follow a rule or set of rules to arrange data into classes. The part of historical data may be used to form a classification rule or formula for making any decision and it is tested on remaining data. The part of data used to construct a model is referred as training data and the part that is used to test the model is called as test data. A model developed from training data that captures insignificant data is a well-developed model; when this model is applied to new instances, the insignificant functionalities may alter the output and leads to poor performance. Decision tree technique of classification is used to analyze data. Among many available methods, three decision tree methods are used in this work namely CART, C4.5 and RT. [7]

This work proposes a tree based classifiers in DDoS attacks. Section 2 reviews literature related to the proposed work. Section 3 explains methodology and Section 4 discusses the results of experiments conducted in proposed work. Section 5 concludes the work.

Related Works

Palmieri et al., [8] built a two-stage anomaly detection scheme on the basis of several distributed sensors all through the network. The first step in Independent Component Analysis, developed as a Blind Source Separation problem, pull out the basic traffic elements (the 'source' signals) that correspond to the independent traffic dynamics from the multi-dimensional time series received from the sensors, relating to the perceived 'mixed/aggregate' impact of traffic on their interfaces. The baseline traffic profiles are built using these components that are required in the second supervised stage. It has its basis in a binary classification strategy driven by machine learning-inferred decision trees.

Ashraf &Latif [9] with a viewpoint on intrusion and DDoS attacks developed Software-Defined Networking (SDN) along with Open Flow protocol. To mitigate such attach machine learning based techniques is suggested.

Balkanli et al., [10] studied the execution of two supervised learning methods as well as two open-source NIDS on backscatter dark net traffic. The authors utilize Bro and Corsaroopen-source systems and the CART Decision Tree as well as NB machine learning classifiers. The author designed the machine learning classifiers using variety of feature sets and various sizes of training/test sets to comprehend the significance of information pre-processing. The outcome of machine learning base approach shows a great performance can be achieved on backscatter dark net traffic without the usage of IP addresses and port numbers.

Cheikh et al., [11] put forth a new technique for detecting Denial of Service attacks with a set of classifiers and visualizing them in real time. In this method, the network parameter values (data packets) are collected, which automatically represents as simple geometric graph form to emphasize on relevant elements. The efficacy of the technique can be proved through a MATLAB simulation of network traffic drawn from the 10% KDD. It is effective when compared with other classification techniques for intrusion detection.

Tama & Rhee [12] utilized and studied the performance of Multiple Classifier System (MCS) for detecting DoS attacks. Some of the well-known base classifiers like C4.5, Support Vector Machine (SVM), as well as K-Nearest Neighbor (KNN) are integrated through combining voting scheme as well as this work, compared the output with already present ensemble learning protocols like bagging, AdaBoost, as well as rotation forest. The experiments utilizing NSL-KDD data set, MCS strategy outperform other ensemble learners and single classifiers.

Methodology

Decision tree is a classification approach in data mining. This classification approach should be understood to create a scheme from a reclassified data-set. The decision tree divides into two classes namely the normal and the attack. The object, based on the class of attack distinguishes between the normal and the attack pattern. The process is repeated for all the classes. To develop the classifier both training and testing data are used. The decision tree technology is common and is the rapid type of classification approach. The construction of decision tree follows top-down approach; divide-and-rule method is applied in this algorithm.

When building, a decision tree the choice of testing attribute is important as it decides the split up of the sample set. Different type of decision tree algorithm uses different technologies. If the size of the sample set is very huge then the tree will have more branches and more number of layers. Besides this, the abnormality and the noise present within the training-sample-set create some abnormal type of branches. Under such cases it is necessary to clip the decision tree. The decision tree algorithm does not involve huge background details making it a simple learning process for the users. In this work, CART, C4.5 and RT classifiers are described.

Classification and Regression Tree (CART): CART is a popular method to build decision trees in the machine learning domain. This classifier constructs a binary decision tree through splitting of records at every node, as per a function of a single feature. Best split is determined using the Gini index. The CART method is typically called as binary recursive partitioning. It is a binary procedure in which the parent nodes are divided exactly into two child nodes and the name recursive process is due to the fact that each child node is again treated as a parent node. [13]

In CART analysis, there is a set of rules, the main factor in dividing all nodes in a tree. The rules decide when a tree becomes complete and assigns a class outcome to all terminal nodes. Defence American Research Project

Agency (DARPA) has an intrusion dataset with 5092 cases as well as 41 parameters. The CART analysis considered up to 5092 times 41 divides for 208772 possible splits. The nodes are initially divided into two nodes similar to that of root node. Again, every input field is checked for further possible splits. If no significant split is found for a node, then the diversity of that node is reduced and it is labelled as leaf node. The algorithm used tends to grow more tress on nodes until no tree can be grown, instead of deciding whether a node is terminal or not.

In CART a complex tree is constructed and then it is pruned back to the original tree depending on the outcomes of cross-validation or set test validation. The tree is pruned back depending on its performance for a given set of test data. The cross validation is used to choose a tree that can perform well on unpredictable data. The CART protocol is resilient to missing data. While building a tree, if a value is not present for a certain predictor in a certain record, then that record is not utilized in deciding the optimum split. The CART utilizes all possible information in hand to decide on the optimal split. When CART is utilized for predicting new data, missing values will be handled by the surrogates. Surrogates are split values as well as predictors which imitate an actual divide in the tree are it is utilized instead of missing prepared predictor data.

The steps involved in CART are as follows: 1. Rules for splitting data at nodes have their basis on values of a parameters. 2. Further splitting of tree is stopped at the leaf/terminal node. 3. Lastly at each leaf/terminal node there is a prediction for target variable. The merits are: 1. CART does not depend on the data of a particular type of distribution. 2. There is no significant impact due to outliers in input data.

C4.5: C4.5 is a well-known machine learning algorithm and a new variant of ID3 protocol. The decision tree formed may be clearly understood by the user. When a tree is constructed using this algorithm there is sufficient data gain while building which helps to keep the tree as small as possible. Pruning is a process carried out to reduce the size of trees and get a smaller tree. It also reduces the complexity of the classifier and increases the prediction accuracy.

The C4.5 algorithm can be classified as follows [14]:

Stage 1: Build decision tree
Protocol: C4.5 creates a decision tree from the provided training data.
Input: Training sample set T, set of potential features and Features-list.
Output: A decision tree.
Generate a root node N:

- If T is part of C, then return N as a leaf node, and mark it as C.
- If feature-list is empty or the remaining sample set of T is below the given value, then return N as leaf node, and mark it as a class that occurs frequently
- Compute the IG ratio for all features in the features-list
- If test-feature is the test feature of N, then test feature is equal to the feature that has the greatest IG ratio in features-list
- Find the division threshold if the testing attribute is continuous
- For all new leaf nodes developed by node N.

Prune the tree after calculating the classification error rate of all nodes.

Stage 2: Extract classification rules
In a decision tree, every branch corresponds to a test output, and every leaf node signifies a category or category distribution. If all paths from root to leaf nodes are traced, the conjunction of every feature-value comprises the antecedent of rules whereas the leaf node comprises the consequent of rule. Based on this the tree is converted into IF-THEN rule.

Stage 3: Determine network behaviour
For any new network behaviour, find out if it intrudes or not based on classification rules.

Random Trees (RT): Leo Breiman and Adele Cutler introduced the term Random Forests (RF) also referred as RT as a collection of decision tree classifiers. Decision tree uses the chain of simple decisions obtained on the basis of the results of sequential tests for assigning class labels. The decision tree branches are made of sets of decision sequences wherein nodes are tested and the leaves correspond to the class labels. The training data set is recursively split into more number of homogeneous sub-sets on the basis of tests applied to one or more values of the input features vector. Prediction or label assignment is carried out at the terminal nodes of the tree through usage of an allocation strategy. [15]

In RT, input feature vector is categorized with each tree in the forest and the final prediction is made on the basis of majority voting. The trees are trained with the same variables, however with differing sets of training samples. The training sets are chosen through usage of bootstrap process on the original training data set wherein for every training dataset, the same quantity of vectors as in the original set are arbitrarily chosen with replacement. An arbitrary sub-set of the parameters is employed at every node of the trained trees for finding the most suited split. The size of sub-sets created at every node is fixed for every node as well as tree by a training variable. There is no pruning of trees. The classification error is calculated for every tree using out-of-bag data. Then it is composed by the left-out vectors in the training stage of the single tree classifiers by sampling with replacement.

RT is a collection of tree predictors known as forest. The classification process is as below: the RT classifier takes the input feature vector, classifies the vector with every tree in the forest. The class label which receives the maximum votes is given as output. In the event of regression, the classifier response is the average of the responses of every tree in the forest. Most of the machine learning protocols consider simple classifiers that fit the given data as the best approximation to the target function as complex models tend to overfit the training data leading to poor generalization (Tanagra tutorials).

Random forests though structurally the same as classical decision trees they are trained in a different way. Training does not consider an extensive search of the possible test candidates as only a randomized sub-set in order will create many different and independent RTs. An RT is grown through an iterative process. In every iteration, a set of L test candidates is arbitrarily created. The test candidates are employed on every training sample at a node and the entropy gain is calculated on the related split. From this a well-suited candidate is selected and the left and right branches are added to the existing node. The process is continued till the leaf nodes maintain training data of a single class only.

Random forest uses a particular set of arbitrarily selected features at every node in the decision tree. RT is a predictive model which utilizes a set of binary rules. RT is utilized for classification or regression applications. It is simple to understand the decision rules. The classification is rapid once the rules are formed.

Result and Discussion

The evaluation process is divided in three steps: in the first step the packets are captured from the network using a Linux based sniffer placed on a monitoring host, which is based on the popular libpcap library and while in capture mode a filter was used to monitor traffic for the www service only. In the second step, the captured packets for some scenario are grouped into timeframes and the statistical features for each timeframe and overlap time sizes are produced. In the final step, the features data were used to train and evaluate the CART, C4.5 and RT. The CART, C4.5 and RT classifiers are evaluated. The table 1 and 2 and figure 1 and 2 shows the classification accuracy and false positive rate.

Table 1
Classification Accuracy

Number of second	CART	C4.5	Random Tree
2	90.33	91.57	96.3
4	93.35	90.79	92.69
6	92.9	93.74	95.92
8	96.11	95.4	96.51
10	92.58	92.93	91.99

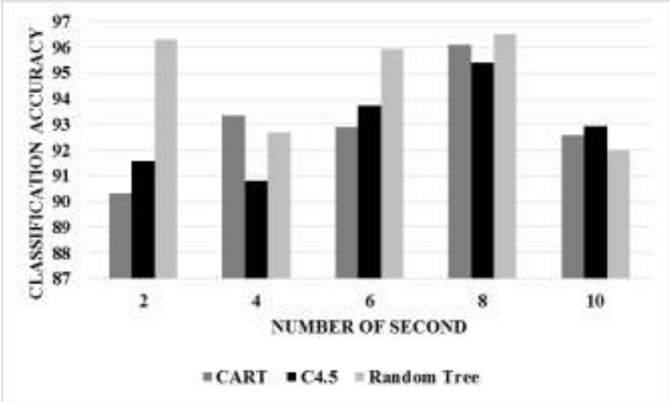

Figure 1: Classification Accuracy

From the figure 1, it can be observed that the RT has higher classification accuracy by 6.39%, 0.7%, 3.19%, 0.41% & 0.63% when compared to CART and by 5.03%, 2.07%, 2.29%, 1.15% & 1.01% when compared to C4.5 for 2, 4, 6, 8 and 10 number of second respectively.

Table 2
False Positive Rate

Number of second	CART	C4.5	Random Tree
2	0.1988	0.1752	0.165
4	0.1896	0.1668	0.1678
6	0.1907	0.1589	0.1462
8	0.1763	0.1569	0.1474
10	0.1576	0.1432	0.1322

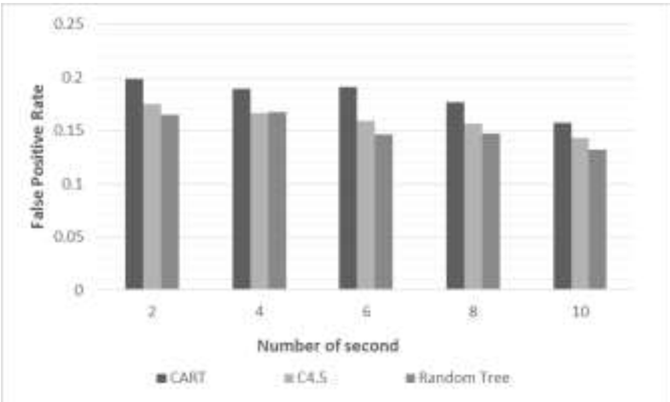

Figure 2: False Positive Rate

From the figure 2, it can be observed that the RT has lower false positive rate by 18.58%, 12.2%, 26.42%, 17.86% & 17.53% when compared to CART and by 6%, 0.6%, 8.33%, 6.24% & 7.99% when compared to C4.5 for 2, 4, 6, 8 and 10 number of second respectively.

Conclusion

In the current study, the role of IDS which is an essential part in the protection of computer data is detailed. DDoS attacks are huge-scale coordinated attacks on the provision of services of victimized systems or networks. They are primarily launched in an indirect manner via a huge set of compromised computer agents on the Internet. The efficacy of various tree based classifiers are evaluated for classifying the DDoS attack. The CART, C4.5 and RT classifiers are suggested. Outcomes prove that the RT improves classification accuracy in the range of 0.63% to 6.39% for CART and by 1.01 to 5.03% for C4.5 when compared with 2, 4, 6, 8 and 10 number of second respectively.

References

1. Shanthini J., Data mining techniques for efficient intrusion detection system: A survey. *International Journal on Engineering Technology and Sciences (IJETS)*, **2(6) (2015)**

2. Ashoor A.S. and Gore S., Importance of Intrusion Detection system (IDS), *International Journal of Scientific and Engineering Research*, **2(1)**, 1-4 **(2011)**

3. Mangrulkar N.S., Patil A.R.B. and Pande A.S., Network Attacks and Their Detection Mechanisms: A Review, *International Journal of Computer Applications*, **90(9)**, doi: 10.5120/15606-3154 **(2014)**

4. Bhuyan M.H., Kashyap H.J., Bhattacharyya D.K. and Kalita J.K., Detecting distributed denial of service attacks: methods, tools and future directions, *The Computer Journal*, doi: 10.1093/comjnl/bxt031 **(2013)**

4. Bhaya W. and Manaa M.E., Review Clustering Mechanisms of Distributed Denial of Service Attacks, *Journal of Computer Science*, **10(10)**, 2037 **(2014)**

5. Zuckerman E., Roberts H., McGrady R., York J. and Palfrey J., Distributed denial of service attacks against independent media and human rights sites, The Berkman Centre **(2010)**

6. Yadav R. and Garg K., Knowledge Based Analysis of Statistical Tools in Attack Detection, *IOSR Journal of Engineering (IOSRJEN)*, **2(7)**, 54-57 **(2012)**

7. Palmieri F., Fiore U. and Castiglione A., A distributed approach to network anomaly detection based on independent component analysis, *Concurrency and Computation: Practice and Experience*, **26(5)**, 1113-1129 **(2014)**

8. Ashraf J. and Latif S., Handling intrusion and DDoS attacks in Software Defined Networks using machine learning techniques, In Software Engineering Conference (NSEC), 2014 National, IEEE, 55-60 **(2014)**

9. Balkanli E., Alves J. and Zincir-Heywood A.N., Supervised learning to detect DDoS attacks. In Computational Intelligence in Cyber Security (CICS), 2014 IEEE Symposium, IEEE, 1-8 **(2014)**

10. Cheikh M., Hacini S. and Boufaida Z., Classification of DOS Attacks Using Visualization Technique, *International Journal of Information Security and Privacy (IJISP)*, **8(2)**, 19-32 **(2014)**

11. Tama B.A. and Rhee K.H., Performance Analysis of Multiple Classifier System in DoS Attack Detection, In International Workshop on Information Security Applications, Springer International Publishing, 339-347 **(2015)**

12. Vijayasankari S. and Ramar K., Hybrid Feature Selection for Modelling Intrusion Detection System and Cyber Attack Detection System, *International Journal of Applied Information Systems (IJAIS)*, **3(6)**, 16-22 **(2012)**

13. Kosamkar V., Improved Intrusion Detection System using C4, 5 Decision Tree and Support Vector Machine (Doctoral dissertation, Mumbai University) **(2013)**

14. Albayati M. and Issac B., Analysis of Intelligent Classifiers and Enhancing the Detection Accuracy for Intrusion Detection System, *International Journal of Computational Intelligence Systems*, **8(5)**, 841-853 **(2015)**.