I.J. Computer Network and Information Security, **2014, 9, 43-50**
Published Online August 2014 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2014.09.06

C_2DF: High Rate DDOS filtering method in Cloud Computing

Pourya Shamsolmoali
Jamia Hamdard University/Department of Computer Science, New Delhi, India
Email: pshams@jamiahamdard.ac.in

M.Afshar Alam and Ranjit Biswas
Jamia Hamdard University/Department of Computer Science, New Delhi, India
Email: {aalam, rbiswas}@Jamiahamdard.ac.in

Abstract—Distributed Denial of Service (DDOS) attacks have become one of the main threats in cloud environment. A DDOS attack can make large scale of damages to resources and access of the resources to genuine cloud users. Old-established defending system cannot be easily applied in cloud computing due to their relatively low competence and wide storage. In this paper we offered a data mining and neural network technique, trained to detect and filter DDOS attacks. For the simulation experiments we used KDD Cup dataset and our lab datasets. Our proposed model requires small storage and ability of fast detection. The obtained results indicate that our model has the ability to detect and filter most type of TCP attacks. Detection accuracy was the metric used to evaluate the performance of our proposed model. From the simulation results, it is visible that our algorithms achieve high detection accuracy (97%) with fewer false alarms.

Index Terms—Cloud Computing, Cloud Security, Distributed Denial-of-Service (DDOS), Filtering, C_2DF.

I. Introduction

Cloud computing is a long-held imagination of computing as a utility. Armbrust et al. [1] discussed that cloud has the possibility to change a large part of the IT industry. Currently, it is growing as a computing key platform for sharing resources including infrastructure resources, application resources and software resources [2]. Regardless of the huge amount of online resources, these cloud systems are facing serious security problems.

Distributed denial of service (DDOS) is a type of DOS attacks. The only exception between DDOS and DOS attacks is DDOS sends several malicious packets from multiple hosts (zombies) to the victim node. DDOS generates more traffic than DOS attack [3]. These zombies together form a Botnet, and will generate large amount of distributed attack packets to the victim node. DDOS attacks will block the legitimate access to the server, exhaust their resources and caused considerable financial loss and have become one of the most serious security threats to the internet. While it is easy to start an attack with some attack tools and it is not easy to stop it [2, 4]. Therefore, these critical services and infrastructure need protection. Network performance degradation, revenue loss, and service unavailability at significant time are some of the issues that motivated us to offer protection for these collaborative applications. For example DDOS attacks such as SYN flood, HTTP flood, UDP flood, and buffer overflow have been posing a serious threat to resource canters [5]. DDOS attacks could harm a company's image and reputation. They could also affect the assurance of users. In recent years, DDOS attacks have been used as a tool of cyber warfare, retribution, and protest. Latest events happened in the December 2010 that disabled Visa and MasterCard Websites for more than a day [6]. Recently, many researchers on DDOS defence have been worked and lots of new techniques have been put forward. In DDOS attack There are three main branches of the research: detection of attack [4, 5, 7, 8, 9, 10, 11], filtering of attack [2, 6], and attack traceback [12]. The majority of the present DDOS defences are proposed through currency based [13] technique, where a sender is required to expend scarce resources to verify his legitimacy before sending packets. Although the currency based network shows to be more secure than the conventional open internet, they generally require the changes to both end systems and intermediary routers [4]. Packet scoring is another technique [1] that gets some attributes from TCP and IP headers and then uses classification algorithm or statistic theorems to analyze packets. It has a high filtering accuracy and easy to deployed, but it is not suitable for handling large amounts of attack traffic. Also scoring has a costly processing operation.

Wrongly use of detection methods able to recognize packets that match a known pattern or signature. But, these methods fail to detect unknown anomalies. Anomalies can be an old type of attack that has changed its pattern in an obtrusive manner to avoid detection. Or it can completely be a new form of attack. Methods of anomaly detection are used to detect the traffic patterns that differ from the modelled of normal traffic behaviour. The identified anomalies can be either a normal or attack

traffic. Though, normal traffic modelling is not an easy task in today internet that growing continuously. In recent times, mechanisms such as stateful firewall and intrusion detection or prevention systems (IDS, IPS) are used for the detection and prevention of DDOS attacks. But these methods are vulnerable to DDOS attack as the state tables in firewall were overwhelmed by moderate size DDOS attack [10]. A DDOS defence mechanism should be able to differentiate attack packets from legitimate ones with high accuracy, Minimum resource consumption and low false negative and positive rate. We are specially focused on detection and mitigation of DDOS attacks in the cloud computing which is purely on the service oriented architecture (SOA) system .In this paper, we propose a based DDOS mitigation system called the "Cloud Confidence DDOS Filtering (C_2DF)". This system generates value distributions of some attributes in the TCP and IP headers such as source IP address, packet length, sequence number, and Time to live (TTL) then uses a decision tree algorithm for detection of the DDOS packets. C_2DF has a very high filtering accuracy, easy to deploy and requires very small storage. The rest of this paper is organized as follows: section 2 introduces some related work done in security of cloud computing and the DDOS attacks that threaten this security. Section 3 gives a scenario of Cloud Confidence DDOS Filtering method. Section 4 experiments and evaluations of our proposed model. Finally section 5 covers conclusions of our work.

II. Related work

A. Cloud Computing Attacks

Specht and Lee [14] argue that "DDOS attack is generally classified into bandwidth depletion and resource depletion attack". In bandwidth depletion attack, attackers flood the target with huge packet traffic that avoids the legitimate traffic and intensify the attack by sending messages to broadcast IP address. In resources depletion attack, attackers try to tie up the significant resources (processor and memory) then trying to unable the victim to process the services. Kumar and Selvakumar [15] have done research on DDOS attacks and existing attack tools. They show that the DDOS attack has the subsequent characteristics:

- Through port 80 HTTP requests are flooded.
- Flags in the UDP and TCP protocols are manipulated.
- Routing table in a host or gateway is changed.
- Source and Destination IP address and packet port number are randomly created.
- Length of packet, sequence number and window size are fixed all through the attack.

Dou et al. [2] proposed a method for filtering a DDOS attack called as CBF. This system calculates the score of a particular packet in the attack time and decides whether to discard it or not. Kumar and Selvakumar [5, 10, 16] recently have done a researches on DDOS attack detection. They evaluate the performance of a comprehensive set of machine learning algorithms for choosing the base classifier. They note that single classifier creates error on different training samples. So, by generating an ensemble of classifiers and combination of their outputs, the total error can be much reduced and the detection accuracy can be significantly improved. Varalakshmi and Selvi [11] proposed a five level DDOS defense mechanism by using an information divergence method that detects the attacker and rejects the packets in a fixed amount of time in an organized way. Du and Nakao [4] proposed architecture to mitigate a DDOS attacks by introducing a credit-based accounting mechanism, where a machine can send packets based on its credit points earned by its legitimate communication behaviours in its place of using resources in advance. Mirkovic and Reiher [17] introduce a system that discards packets based on the arrival of packets from an attacker machine Rather than DDOS detection. This helped in mitigating DDOS attack's effects. Varalakshmi et al. [18] discusses the efficiency of DDOS attacks on statistical based filtering in a common context where attackers are smart. It considers different cases such as the dynamic and static property of the attacker and the mitigating. Kim and Reddy [19] proposed a traffic detector, which can be worked in real time by monitoring the packet headers. In the proposed system a wavelet filter was used along with weighted correlation to identify DDOS attacks. Chonka et al. [12] note that HTTP Denial of service and XML Denial of service attacks are the most serious threats to cloud computing. They also offer a scheme called as Cloud Trace Back (CTB) to locate the source of these attacks. In [2, 4] the authors improved the accuracy, detection result and reduced the amount of False positive. Modi et al. [20] have done a comprehensive overview on intrusion detection system (IDS) and Intrusion prevention system (IPS) in cloud. Li and Li [15] based on normal traffic analysis introduced an adaptive system, which is used for defending against DDOS attacks. This system could be able to infiltrate the packets that are entering the network and adaptively change its configurations according to the attack severity and network conditions. Lu et al. [22] proposed a novel framework to robustly detect DDOS attacks and recognize attack packets. The key idea of their framework is to make use of spatial and temporal correlation of DDOS attack traffic. They developed a perimeter based anti DDOS system, in which traffic is analyzed only at the edge routers of ISP network.

B. DDOS attacks

DDOS attacks make use of zombies, which send requests to a target system as orders of the attacker. Mostly DOS attacks involve spoofing of the attackers' IP addresses as the victims' IP addresses, making it complex to recognize the attackers. Kim et al. [23] have done systematic review on the dark side of the internet. The authors summarized the major form of DOS attacks as mentioned bellow.

- **Ping flood**. A ping flood is the most basic form of DOS. The attacker simply sends a huge number of ping packets to the target. If the target sends replies, the effect is amplified.
- **Smurf attack**. Similar to the ping flood attack, a smurf attack uses ping packets. The attacker sends ping packets, with the spoofed source IP address to be the victim's IP address, in the direction of computers that keep a broadcast address. All computers in the broadcast address that get the ping packet send responds to the victims' IP address. A packet that sent to the broadcast address is amplified by the number of computers that send reply packets.
- **TCP SYN flood**. TCP SYN flood is a form of attack where the attacker sends a huge number of SYN packets (connection requests) to the target, and fills up the connection queues on the target, so that the target cannot launch connections for legitimate TCP clients. A TCP connection is established using a three-way handshake between a client and a server. In TCP SYN flood the attacker behave as a client. Generate a huge number of SYN packets and not send ACK packets to the server, causing the server to consume all available TCP connection queues.
- **UDP flood**. In a UDP flood attack, the attacker sends a large number of UDP packets to random ports on the target. As the UDP does not have a congestion control system, the attacker can potentially send a very large number of packets. This attack is generally used with IP address spoofing, so that the attacker can stay away from detection.
- **Application-level attack**. The kinds of attack argued thus far all exploit network protocols or services. DOS attacks can also be accepted at the application level. For example, the attacker can command zombies to send HTTP requests to a web server to download a large file or execute expensive database operations. This will consume CPU and network resources at the server, limiting its availability to other legitimate clients.
- **DNS amplification attack**. DNS amplification attack uses DNS queries. The size of the reply to a DNS query can be much larger than the DNS query. The attacker creates a reliable domain name server, "chance.com", and registers a garbage text of large size, for example 5000 bytes, as the text Resource Record (RR) of chance .com. Next, the attacker commands zombies to send queries to their domain name servers for the text RR of chance.com, with the zombies' IP address which is spoofed to be the victim's IP address. When the domain name servers that receive queries allow recursion, they recursively query the reliable name server of chance .com for its text RR and get the reply to the source IP address, which is the address of the victim.
- **Peer-to-peer attack**. Conventional DDOS attacks use zombie computers to send a large number of requests to the victim.P2P attacks use clients linked to P2P file sharing hubs.
- **Mail bomb attack**. In a mail bomb attack, the attacker sends a large amount of e-mails to a target e-mail address to overflow the victim's mailbox or slow down the mail server. The attacker may command zombies to send e-mails to the victim e-mail address simultaneously. Attacker may create each e-mail with a different message to pass the spam filters.
- **Variable-rate and low-rate attacks**. Although it is difficult to trace back the attack, in general it is simple to know when the attack actually takes place, because the server becomes unavailable or drastically slows down. The monitoring system at the target system raises an alarm when there is an unusually large volume of traffic at a constant rate. However, the attacker may send variable-rate and low-rate traffic to the victim, making it complex for the victim to understand that actually an attack is taking place. And if the attack is not detected, the administrators may incorrectly conclude that legitimate traffic has increased and increase investment in network bandwidth.

III. Design of Proposed C_2DF System

Cloud Confidence DDOS Filtering (C_2DF) can be used in a network structure such as LAN or grid network. C_2DF is made within a virtual machine to make placement within the cloud network compatibility. C_2DF is installed at the edge routers in order to be next to the source end of the cloud network. Generally if no security services are prepared for server, the system becomes absolutely vulnerable to attacks. C_2DF can cure this by being located before the server. As a result all service requests are first sent to the C_2DF for checking. The proposed architecture is shown in Fig 1. The proposed C_2DF system consists of the following

The proposed C_2DF system consists of the following two stages:

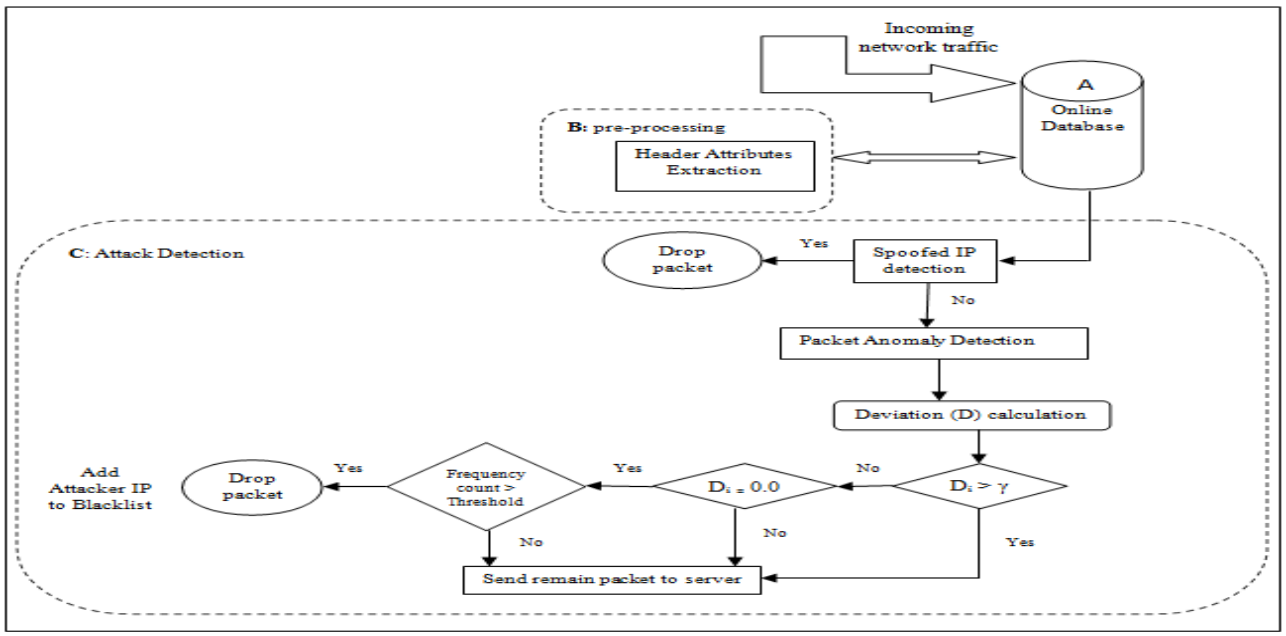

Fig. 1. Overall design of C$_2$DF

A. Pre-processing

The input to the pre-processing stage is the network traffic and the output of this stage is classified data. Pre-processing refers to the process of extracting information about packet and construction of new statistical features. The pre-processing steps are explained as follows:

In first step the system extract the seven header fields of the packet. In first step the system extract the seven header fields of the packet. These fields are Packet Length, Protocol Type, Time to Live (TTL), Source Port, Destination Port, Window size and Flag. We used these Destination Port, Window size and Flag. We used these features to find the statistical properties such as variance and standard deviation. These features quantify the behavioural characteristics of a connection in terms of number, types of various data and time.

B. Detection

The proposed DDOS detection system highly reduces the false positive and false negative rates since we analyze the entire packets. For each incoming packet during the profiling time, the system extracts the Time to Live (TTL in short) value and computes the number of hops the packet has travelled. Attacker can spoof the packet header, but not able to manipulate the Hop count value. By comparing TTL's value with IP to hop count (IP2HC in short). If no accurate matches are found, then the packet is spoofed and the system discards it directly.

The rest of the packets are forwarded to the next level of proposed DDOS detection system, known as deviation or anomaly detector. In this step the current header information compare with the profile information that is already trained in the database, to determine the information divergence between the two profiles. The packets are analyzed at this point for any anomaly in the Header of the packets. The extracted attributes and their probabilities for each field are obtained for a certain time span. This probability allocation is learned for a given period of time. This learned profile for each attribute is refreshed regularly which ensures minor changes in behaviour of genuine users. In this scenario we are using the concept of Jensen-Shannon Divergence. We are using the following Equations to compute Information Divergence (D) in header for each IP.

$$Di(P \parallel Q) = \sum P(i) \times \log \frac{P(i)}{Q(i)} \tag{1}$$

$$Di = \frac{1}{2} D(Pi \parallel Mi) + \frac{1}{2} D(Qi \parallel Mi) \tag{2}$$

$$\text{Where } M = \frac{1}{2}(P + Q) \tag{3}$$

By comparing the profile of already learned traffic and the incoming traffic for the period of observation, it gives the opportunity to distinguish the legal traffic from the attack traffic.

Information divergence is a non-commutative measure of the difference between two probability distributions P and Q. P typically represents the "True" distribution of data, or a precise calculation theoretical distribution. The measure Q usually represents a theory, model, or approximation of P. By referring to (1) this system firstly compute divergence for (P ‖M) and (Q ‖M) then by using (2) we can get the total divergence for the (ith) IP. If for the (ith) IP Information Divergence is more than learned profile (Di > γ) so the two probabilities have divergence. Therefore, P and Q denote the behaviour of different entities. But if D_i is equal to 0.0 then it indicates that there is a possibility of flooding attack. We analyze the packets which are stored in an intermediate buffer for flooding attack using the frequency counter. Each incoming packet is compared with the identity of

blacklisted IP for the exact similarity. If an exact match is identified the frequency count of packet is incremented by 1. Since in a DDOS attack, a large number of packets surround the victim in a short period of time. There is a very high possibility that the attacker sends similar packet many times which almost happens in a flooding attack. The frequency count of each packet is checked. If it exceeds the threshold value for a particular IP, the system indicates an attack and that IP is identified as the attacker. Then the system discards the packet and adds the attacker IP to the blacklist. In our proposed system we used the following two algorithms for detection and discarding of attacks, Algorithms.1, 2.

Algorithm.1. Hope-count Algorithm

Input: Packet Header Attributes
Output: drop spoofed packets
Begin
For each packet:
Extract the final *TTL* ($T_{\hat{g}i}$) and the source IP address S_i;
Retrieve the initial *TTL (Tij);*
Compute the hope-count $Hc_i=Tij-T_{\hat{g}i}$;
Get the stored hope-count (His) for the indexed S_i;
If ($Hc_i \neq Hs_i$)
The packet is spoofed so drop it;
Else
The packet is legal so Forward it;
End If
End For
End

Algorithm.2. Packet Anomaly Detection Algorithm

Input: packet Header Attributes
Output: drop illegitimate packet and Identify Attacker IP
Begin
For each sample *(t)*
If learning period
Define probabilities of each value for header Attributes for every *IP* and store them;
Else
Define probabilities of each value for the header Attributes for every IP;
Define the D_i *for* IP_i;
If $Di \approx 0.0$
Possible for flooding attack
Check for flooding using frequency counter;
If flooding attack (*frequency counter* > *threshold*)
DDOS attack detected
Drop matching packets;
Else
Forward to destination;
End If
Else
Add the Attacker *IP* to Blacklist;
End If
End If
End For
End

IV. Performance Evaluation

The proposed model is evaluated with respect to implementation. For generating network traffic and DOS attack we created a cloud Lab. We have chosen a HP proliant ML 330 G6 server with following features: Intel Xeon E5606, 6 GB RAM, 2×300 GB SCSI Hard Drives, we also selected VMware ESXI 5.0.0 Hypervisor as virtual machine manager (VMM) and windows 7 as guest operating system. We also have 4 clients with following features: Intel core 2 duo (2.26 GHZ), 2GB RAM.

On each client machine we installed virtual machines with the random IP addresses for generating traffic. One of these client is generating Normal traffic consist of FTP access, Web page access, e-mail access and UDP traffic. The performance of server at the time of normal traffic is shown in Fig 2.

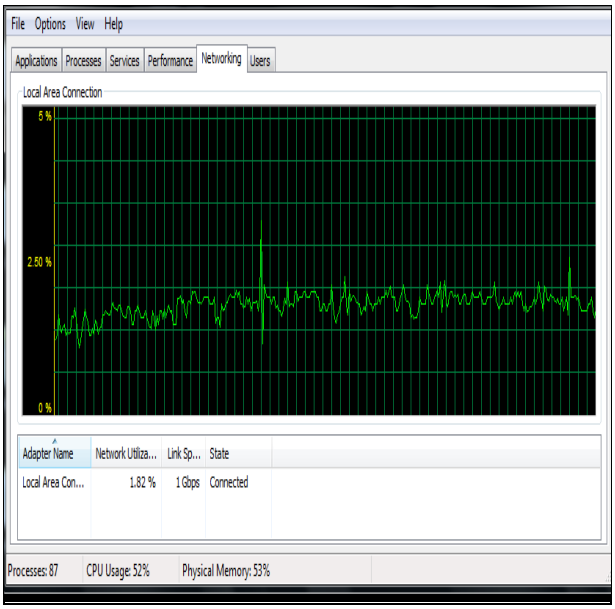

Fig. 2. server performance in normal traffic

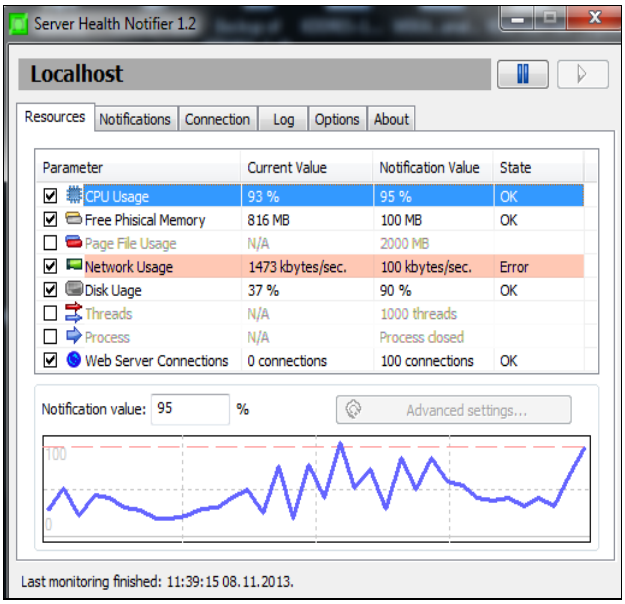

Fig. 3.CPU performance

The rest of clients generate attacks traffic. In each client machine, 6 virtual machines were created with spoofed IP addresses. We used Netwag tool [24] to generate well known DDOS attacks such as a Smurf attack, TCP SYN attack and etc. we have also tried using hping, packet crafting tool. In our proposed model, only the TTL, Total length, Source port, Destination port, Protocol type, Window size and Flag of packet are extracted. To capture the packets and access all its header information, a packet capturing tool JPCap is used. JPCap is an open source java Library for transferring and capturing network packets. By using JPCap, we can establish applications to capture packets from a network interface and explore them in java. Fig 3 shows CPU performance plots of the hypervisor during Smurf attacks and TCP SYN flood attacks. For simulation, different machines were used to send non-stop requests to create DDOS attack. In beginning all non spoofed packets are allowed in the learning period and once profiled are learned, deviations are recognized which result in rules being framed and therefore afterward DDOS attack packets which matches the rules are discarded. Once an attack is detected, rules are framed in order to stop similar packets from infiltrate the system. This might decrease the amount of false positive to small extent.

In our first experiment, we used 1000 data points randomly from our own lab dataset to test the detection percentage and the False alarm rate of proposed model.
The detection rate is the ability of the system to detect attacks over the total amount of attacks. The false alarm is the number of data incorrectly predicted as attack traffic. The equation for detection percentage (DP) and false alarm rate (FAR) is as follows:

$$DP = \frac{TP}{TP+FN} \tag{4}$$

$$FAR = \frac{FP}{FP+TN} \tag{5}$$

False positive and detection rate of traffic that is Trained and tested by our proposed system is showed in Fig. 4. The result shows that the C_2DF was able to detect over 95% of attack traffic with a 4.5% of false positive rate.

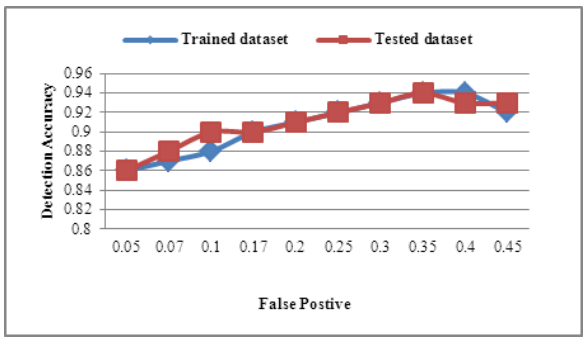

Fig. 4. Detection Accuracy and False positive Rate

As it is not possible to get 100% of detection rate without getting a false positive alarm, we can achieve the maximum possible detection rate by changing the threshold to the maximum value. By increasing the threshold value we improve detection accuracy but thereby increasing the false positive rate leading to the system inefficiency. Classification of any attack based on predefined classes of attacks can be answered successfully through using machine learning techniques [25]. All these techniques are accessible in the data mining community. From the presented list we have selected PART [26], Support Vector Machine [27], Decision Tree [28], Multilayer Perceptron [29] and Bagging [30] to classify into attack type. Support vector machine (SVM) and Multilayer Perceptron are function based techniques, PART and Decision Tree are rules based techniques and Bagging is basically Meta data mining techniques. We used an old dataset, KDD CUP (1999) to check the performance of our model against the existing ones. The details of performances are showed in table 1. The classification accuracy shows the amount of attacks which are classified correctly by the data mining techniques. The number of unclassified instances measures the technique's limitations, which means failures in classifying a number of attacks.

Table 1. Simulation results of different algorithms. For KDD CUP

Classifiers	Correctly Classified Instances	Incorrectly Classified Instances
Decision Tree	92.2	7.8
Bagging	94.5	5.5
Multilayer perceptron	95.3	4.7
PART	98.9	1.1
SVM	97.3	2.7
C_2DF	97.2	2.8

As mentioned in above table the detection accuracy in C_2DF is above 97% but the false positive rate is above 2% .Decision Tree has 93% of detection rate with more than 8% of false positive. Bagging classification algorithm has more than 95% of detection but the false positive is above 6%. Multilayer perceptron has above 96% of detection and the false positive is more than 5%. PART detection rate is above 98% but the false positive rate is more than 1%. For SVM algorithm the detection rate is 98% and the false positive is above 2%. By this comparison we observe that PART and SVM algorithms in detection accuracy have better performance in comparison to our proposed model. But in false positive rate our proposed model has almost same performance in comparison to SVM algorithm. PART has better performance than our proposed model. The results show that our proposed model has problem in detection rate of attack traffic. This seems to indicate that the DDOS defence system setting needs to be re-adjusted to improve it efficiency.

V. Conclusion

Serious services are frequently badly affected by DDOS attacks, in spite of the usual deployment of network attack prevention mechanisms such as Intrusion Detection systems and Firewall. Some intrusion detection systems could detect only attacks with recognized signatures. Predicting the upcoming attacks is impossible. Hence, we have proposed a comprehensive solution for DDOS attacks in reliable cloud computing. In this model we proposed two level of filtering for detecting the attackers. Firstly the system extracts the seven header fields of each packet entering the system. In next level the system compare the value TTL with the value stored in the table of IP to hop count (IP2HC). If the values do not match, the packet is spoofed and the system drops it. In final level we used the concept of Jensen-Shannon Divergence. The incoming packets header information compare with the profile information that already is stored in the database, to find out the information divergence between the profiles. The packets are comparing for any anomaly in the header of the packets. The core of our work is to present an efficient false positive reduction technique to reduce the false alarms. From the simulation experiments, it is evident that the C_2DF results in high detection rate of 97%. Further, it has been observed from the end results that C_2DF has better performance than Decision Tree, Bagging and Multilayer Perceptron Algorithm. But in comparison to SVM and PART it has almost the same performance. In future work we have plan to re-adjust the setting of C_2DF to improve it detection accuracy.

Acknowledgment

We would like to thank Dr. Ashley Chonka and Research consul of Iran embassy in New Delhi for his helpful comments that helped to improve the quality of this work.

References

[1] M. Armbrust, Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, Matei Zaharia. A view of cloud computing. Communication of ACM, *53(4), (2010), pages: 50–58.*

[2] Doua, W., Chen, Q., Chen, J. A confidence-based filtering method for DDoS attack defence in cloud environment. Future Generation Computer Systems, 29(7), (2013), pages: 1838–1850.

[3] Lo, Chi-Chun., Huang, Chun-Chieh., Ku, Joy. A Cooperative Intrusion Detection System Framework for Cloud Computing Networks. In proceeding of International Conference on Parallel Processing, (2010), pages: 280-284.

[4] Du, P., Nakao, A. OverCourt: DDoS mitigation through credit-based traffic segregation and path migration. Computer Communications, 33(18), (2010). 2164–2175.

[5] Raj Kumar, P. A., Selvakumar, S. M2KMIX: Identifying the Type of High Rate Flooding Attacks using a Mixture of Expert Systems. International journal of Computer Network and Information Security, 4(1), (2012), pages: 1-16.

[6] Lent, R. Evaluating a migration-based response to DoS attacks in a system of distributed auctions. Computers & Security, 3(1), (2 0 1 2), pages: 327-343.

[7] Chonka, A., Abawajy, J. Detecting and Mitigating HX-DoS attacks against Cloud Web Services. IEEE Conference on Network-Based Information Systems, (2012), pages: 429-434.

[8] Chonka, A., Singh, J., Zhou, W. Chaos Theory Based Detection against Network Mimicking DDOS Attacks. IEEE Communications Letters, 13(9) (2009)pages: 717-719.

[9] Chonka, A. Xiang, Y. Zhou, W. Huang, X. Protecting Cloud Web Services from HX-DoS attacks using Decision Theory. IEEE conference on communications: advanced internet and cloud, (2012) pages 85-91.

[10] Raj Kumar, P. A., Selvakumar, S. Distributed denial of service attack detection using an ensemble of neural classifier. Computer Communications, 34(11), (2011), pages: 1328–1341.

[11] Varalakshmi, P., Thamarai Selvi, S. Thwarting DDoS attacks in grid using information divergence. Future Generation Computer Systems, 29(1), (2013), pages:429–441.

[12] Chonka, A., Xiang, Y., Zhou, W. Alessio Bonti. Cloud security defence to protect cloud computing against HTTP-DOS and XML-DOS attacks. Journal of Network and Computer Applications, 34(4), (2011), pages: 1097-1107.

[13] Walfish, M., Balakrishnan, H., Karger, D., Shenker, S. Dos: fighting fire with fire. ACM Workshop on Hot Topics in Networks (HotNets), (2005).

[14] Specht, S. M., Lee, R. B. Distributed Denial of Service: Taxonomies of Attacks, Tools, and Countermeasures. IN Proceedings of the International Workshop on Security in Parallel and Distributed System, (2004), pages: 543–550.

[15] Raj Kumar, P. A., Selvakumar, S. Detection of distributed denial of service attacks using an ensemble of adaptive and hybrid neuro-fuzzy systems, Computer Communications, 36(3), (2013), pages: 303–319.

[16] Raj Kumar, P. A., Selvakumar, S. M2KMIX: Identifying the Type of High Rate Flooding Attacks using a Mixture of Expert Systems. International journal of Computer Network and Information Security, 4(1), (2012), pages: 1-16.

[17] Mirkovic, J., Reiher, P. Taxonomy of DDoS attack and DDoS Defence Mechanisms, ACM SIGCOMM Computer Communication Review, 34(2), (2004), pages: 39–53.

[18] Varalakshmi, P., Thamarai Selvi, S., Javed Ashraf, A., Karthick, K. B-tree based trust model for resource selection in grid. In proceeding of Signal Processing Communications and Networking, (2007), Pages: 222–227.

[19] Kim, S., Narasimha Reddy, A.L. Statistical techniques for detecting traffic anomalies through packet header data, IEEE/ACM Transactions on Networking, 16(3), (2008), pages: 562–575.

[20] Modi, C., Patel, D., Borisaniya, B., Patel H., Patel, A., Rajarajan, M. A survey of intrusion detection techniques in Cloud. Journal of Network and Computer Applications, *6(1), (2013), pages: 42–57.*

[21] Li, M., Li, M. An Adaptive Approach for Defending against DDoS Attacks. Mathematical Problems in Engineering, *(2010).*

[22] Lu, K., Wu, D., Fan, J., Todorovic, S., Antonio Nucci. Robust and efficient detection of DDoS attacks for large-

scale internet. International Journal of Computer and Telecommunications Networking, *51(18), (2007), pages: 5036–5056*.
[23] Kim, W., Jeong, Ok-Ran, Kim, C., So, J. The dark side of the Internet: Attacks, costs and responses. Information Systems, *36(3), (2011), pages: 675–705*.
[24] Netwag Tool, http://ntwag.sourceforge.net/.
[25] Khorshed,Md.T., Shawkat Ali, A. B. M., Wasimi, S. A. A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing. Future Generation Computer Systems, *6(28), (2012), pages: 833-851*.
[26] Frank, E., Witten, I.H. Generating accurate rule sets without global optimization. In Proceedings of International Conference on Machine Learning, *(1998), pages: 144-151*.
[27] Platt, J.C. Fast training of support vector machines using sequential minimal optimization. Advances in kernel methods, MIT Press *(1999), pages: 185 – 208*.
[28] Quinlan, J.R. Book review: C4. 5 Programs for Machine Learning, *16(3), (1994), pages: 235-240*.
[29] Lopez, R., Onate, E. A variational formulation for the multilayer perceptron. Lecture Notes in Computer Science, In proceeding of Artificial Neural Networks, *4131, (2006), pages: 159–168*.
[30] Bauer, E., Kohavi, R. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. Machine learning, *36, (1999), pages: 105-139*.

Pourya Shamsolmoali. He is a researcher in Computer Science Dept., Hamdard University, New Delhi, India. He received his B.Sc. and M.Sc. degree in Computer Science from Hamdard University, in 2008 and 2010 respectively. He has published several research papers in Cloud Computing field. His research interest includes Distributed System, Network system, Network Security, wireless Ad hoc network, mesh network, sensor network, Cloud and Grid Computing, Middleware, Data Mining and Machine Learning.

M. Afshar. Alam. He is a Professor in Computer Science; he was Head of Computer Science Department, Faculty of Management and Information Technology, at the Hamdard University, New Delhi, India. In 1997-2000, he founded the Department of Computer Science, Hamdard University. He was also founder of Computer Centre at Hamdard University. He received his Master degree in Computer Science from the Aligarh Muslim University, Aligarh and Ph.D. from Jamia Millia Islamia University, New Delhi. His research interests include Fuzzy logic, Software engineering, Networking, Network Security, Cloud Computing and Bioinformatics. He is the author of a book on Software re-engineering and over 50 publications in International/ National journals, conference and chapter in an edited book. He is a member of expert committee AICTE, DST, UGC and Ministry of Human Resource Development (MHRD), New Delhi, India.

Ranjit Biswas is Head of Department of Computer Science in Hamdard university, he was a Professor at ITM University, His area of specialization is Fuzzy Logic in Computer engineering, rough theory and approximate reasoning, bag data base, Fuzzy data base, parallel architecture, artificial intelligent and pattern recognition.

How to cite this paper: Pourya Shamsolmoali, M.Afshar Alam, Ranjit Biswas,"C2DF: High Rate DDOS filtering method in Cloud Computing", IJCNIS, vol.6, no.9, pp.43-50, 2014. DOI: 10.5815/ijcnis.2014.09.06

