Survey of DoS attack quelling technics

Akash B. Mahagaonkar
Department of Computer Engineering,
Pune Institute of Computer Technology,
Pune, India

Amar Buchade
Department of Computer Engineering,
Pune Institute of Computer Technology,
Pune, India.

Abstract— **The cloud computing is the new buzzword in IT industry because of its various features like pay per go model, flexibility of resource allocation, low maintenance cost, short time for setup, availability and many more. The cloud service providers provides various services over the Internet connection like SaaS, PaaS, IaaS. As application of the cloud increased, attacks on the cloud are also increased. There are attacks like Denial Of Service Attack, Distributed Denial of Service Attack which stops legitimate users from accessing the cloud services provided by cloud service provider. This in result reduces the availability of the cloud services and resources. It results in degraded QoS of the cloud service provider. Hence it becomes crucial to identify ways to detect these attack on the cloud and help the cloud service provider to maintain the availability of resources to legitimate user**

Keywords-Cloud Computing, Cloud Security, DoS, DDoS.

I. INTRODUCTION *(HEADING 1)*

In todays IT environment cloud computing is the most preferred technology. As we can see there are number of new born start up companies are present in the IT industry, but still they are not capable of managing various resources that are required for their functioning in terms of finance or human resource that is required to manage that resources. The cloud computing is vital part for such companies. Numerous cloud computing service providers are present in the market and captured significant portion of the todays market. Lots of organizations are using cloud services for their day to day activities. Even if cloud computing services is growing and gaining popularity, the fear about the availability of cloud services is still an open issue.

Availability of services is adversely affected due to Denial of Service attack. A Denial of Service Attack is an step (or set of steps) executed by an attacker to make resource/resources unavailable to its legitimate users. According to Gligor denial of service attack occur when a user or group of users of a specified service is denied service to another legitimate user or group of legitimate users if the first individual or group makes the specified service unavailable to the second individual or group for a period of time that exceeds the pre-specified waiting time. Gligor [8] makes sure that timeliness of the service is considered in his definition of denial of service attack.

As shown in Fig. 1 in case of flooding based Denial Of Service attack the attacker directly attacks the victim. The attacker will send thousands of request in limited timing window. Here the attacker won’t take any help form other resources available over the Internet. The DOS attack is less severe as compare to the DDOS attack. In this case the attacker will makes use of other machines that are available over the Internet that are vulnerable to security.

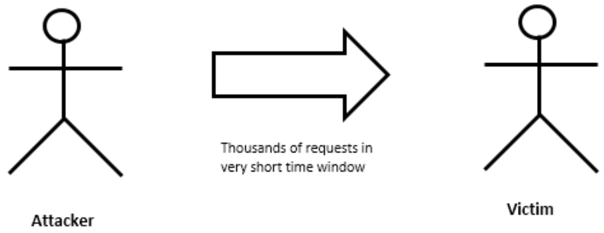

Figure 1. Flooding DoS attack.

CERT Coordination Center [4] defines three basic types of attacks.

1. Consumption of scarce, limited, or nonrenewable resources.
2. Destruction or alteration of configuration information.
3. Physical destruction or alteration of network components

A. *Targets of DoS*

- Application: An attacker attempts to prevent the application from performing its intended task by exhausting a specific resource.
- Virtual Machine: In it an attacker attempts to prevent legitimate users from accessing the virtual machines by sending thousands of illegal request for virtual machine access.
- Operating System: This is same as DoS attack on application however in application DoS attack operating system may defend other applications from the attack. For example TCP SYN flood attack attacker sends flood of TCP SYN packets to the victim without completing TCP handshake and

occupying victims complete connection state memory.

- Router: In this routers routing table is overloaded with tremendous amount of routes which in results in insufficient memory for router or insufficient CPU power to process the routes.

II. Denial of Service Defense Challenges

A. *Internal Architecture related Challenges*

- Sharing of Resources: A misbehaving user (in Cloud computing Rogue VM) can disrupt service by occupying most of shared resources. Such resource sharing featuring the users demand creates an inter-user dependency. That inter-user dependency is a root cause that helps denial of service to occur.

- Multi path Routing: Poul Heegaard et al. defined survivability as the systems ability to continuously deliver services in compliance with the given requirements in the presence of failures and other undesired events [14]. In networking it is the ability to communicate even though networks and gateways (routers) are failing is the most important feature in the design goals of the Internet. The Internet routing infrastructure is designed with the ability to route traffic with alternative paths so that the failed portion of the network can be avoided. If packets from the same source are always traveling along the particular route, then a router knows the set of source addresses that an incoming packet at its particular network interface may have as its source address. Hence a router can distinguish between spoofed and legitimate source address. This multipath routing produces competitive disadvantage at router side as the router cant distinguish between legitimate packet and spoofed packet because of the provision of multipath routing. Therefore multipath routing makes it more complicate to determine the actual source of the attack.

- Accountability: Users with sufficient privileges on a host can use IP address of another host to generate packet. This is called IP address spoofing. These packets with spoofed IP address then can be dumped into the Internet to carry out denial of service attacks. Attackers uses IP spoofing technique to hide the actual origin of attack packets.

B. *Other Challenges*

- Difficulty of segregating requests: It is difficult to categorize incoming packets into malicious requests and legitimate requests. This is true for packets, network flows, transport layer segments, or application service request messages. Even though some malicious behavior can be detected by signature based attack detection mechanisms, attackers normally modify the characteristics of their attack messages to avoid the detection. It is known to all that attackers and defenders are locked in race, especially when it comes to signature-based attack detection. Though anomaly based detection mechanisms can detect unknown attacks, there is possibility of misidentifying normal behavior as an attack.
- Asymmetry of request and response overhead: Asymmetry of request and response overhead refers to the asymmetry in the amount of consumed resources for generating a request at the client and creating its corresponding response at the server. In most cases, a client spends trivial amount of CPU and memory resources to generate a requests, and the operations carried out by the server to produce the corresponding response incurs significantly more resource overhead in comparison.

III. Existing Systems

M. Cotton et al. proposed an approach in which the router is supposed to drop all Martian packets [1]. The Martian packet is the packet whose source or destination IP address is an IP address reserved by the Internet Assigned Number Authority [IANA] for special purposes. Limitation of this approach is we can prevent the attacker from spoofing these small set of IP addresses. The attacker can easily avoid usage of these IP addresses to do attack.

Kihong Park et al. presented route based distributed packet filtering method [2] where edge router segregates the packets as legitimate packets and attack packets depending upon from which link it received the packet. The limitations with this approach is that if multiple routes are permitted when routing packets from source address to destination address we can’t identify the attack. Hence this approach is suitable for static environment. Because according to the design goals of the Internet, Internet communication must continue despite loss of network, gateways for that purpose multiple routes should be permitted.

Jianping Wu et al. developed hierarchical architecture[3] for source address validation. It performs local subnet level, intra Autonomous System level and inter Autonomous System level address validation. This approach has various limitations such as it requires cooperation among different administrative authorities of different Autonomous Systems and it has complex implementation.

Ingress filtering approach proposed by P. Furguson et al. [4] the edge router that provides connectivity to the subscriber’s network is used for attack detection. ISP provider assigns valid IP prefix to each subscriber network. All packets that are generated from that network are checked against those prefixes. To hide it’s identify the attacker may make use of spoofed IP address. Hence that IP address won’t come under the valid prefix address and those attacking packets are dropped at edge network to prevent attack. Even though it is effective approach for DoS attack prevention it has several limitations like at specific router interface all valid IP addresses should be known before the communication takes place. Second limitation is deployment of ingress filtering is not compulsory for all ISPs. As this is router based solution special routers are required which can support ingress filtering. At last the coordination among all ISPs is required for effective implementation.

TTL (Time To Live) is an 8 bit field in IP header. The intent of the field is to catch packets that have been going around in routing loops and discard them, rather than let them consume resources indefinitely [5] Each intermediate router reduces the TTL value of packet in-transit by 1 before forwarding it. Ryo Yamada et. al identification about initial TTL values of different Operating Systems [7] is shown in Table I.

Haining Wang et al. [6] proposed hop count approach where the initial value of the TTL of packet is the smallest value mentioned in table 1 which is larger than the final TTL of the packet. For example if final TTL is 110 then the initial TTL value will be 128. Therefore hop count will be 128 110 = 18. A table with IP address and hop count is created. When a packet is received hop count value is calculated. If source IP address of that packet is not present in the table then entry is made for that IP address and associated hop count. If source IP address of that packet is already exists in the table then current hop count value and the hop count value present in the table is compared. If they are same then packet is considered as legitimate packet. If they are not same then packet is considered as attack packet. There are number of limitations with this approach one of them is as follows. In case if packet takes any different route than usual one then hop count value at the time of calculation will change and packet will be considered as attack packet, even though it is legitimate packet.

TABLE I. INITIAL TTL VALUES OF OS

OS	Protocol	Initial TTL
Linux 2.4 kernel	ICMP	255
Windows Server 2008	TCP, UDP, ICMP	128
Windows7	TCP, UDP, ICMP	128
Windows XP	TCP, UDP, ICMP	128
Linux RedHat 9	TCP, ICMP	64
FreeBSD5	ICMP	64
MacOS X (10.5.6)	TCP, UDP, ICMP	64

The second limitation comes in picture when attacker sends attack packet with the spoofed IP address of a legitimate host before any request from the legitimate packet. In this case this approach will filter attacker packets as legitimate packets and legitimate host's packets as attack packet. In next case of failure if user changes operating system of the client machine in this case the default value of TTL will also change, but hop count for each packet from this client machine will be calculated with help of old default TTL value. Hence it will classify all the packets generated from the host as attack packets.

In their paper Yih Huang and J. Mark Pullen [9] proposed an architecture. In this architecture the packet sampling is associated with the congestion control mechanism at router side to detect DoS attack. If attack has been detected in this phase then packet filters are used to drop those attack packets.

Wanchun Dou et al. proposed a confidence based filtering method [10] for DDoS attack detection in cloud environment. Their proposed approach works in two phase. Attack phase and non-attack phase. Score of each incoming packet is calculated and depending upon score the future of packet is determined.

Thomas Vissers et al. [11] derived a system which uses several attributes for outlier detection in incoming requests to web service in cloud environment. With extracted features they constructed normal profile. The profile is derived using Gaussian model. These models are based on dataset constructed from logged features of previous requests. Using this model they segregate the packet in normal and abnormal category.

Umar Tariq et al. presented node centrality algorithm [12] to denote autonomy capacity to access HTTP node. The autonomy capacity is calculated using the mean of traffic flow and routed path between nodes which is again trains the decision technique application for normal and intrusive activities.

IV. EVALUATION OF ATTACK CLASSIFICATION

The performance of DoS attack detection using machine learning technics can be evaluated by using four indexes calculated as the following equations.

$\text{Accuracy} = (TP+TN)/(TP+TN+FP+FN)$
$\text{Precision} = TP/(TP+FP)$
$\text{Recall} = TP/(TP+FN)$
$F1 = (2\times\text{Precision}\times\text{Recall})/(\text{Precision}+\text{Recall})$

In which TP, FN, FP and TN refer respectively to the number of true positive instances, the number of false negative instances, the number of false positive instances and the number of true negative instances, as defined in the Table II.

TABLE II. Confusion Matrix.

	Predicted Attack	**Predicted Normal**
Actual Attack	TP	FN
Actual Normal	FP	TN

V. CONCLUSION

In recent years number of attacks on cloud service providers are increased. Even though cloud service provider can pool large number of resources to provide service to legitimate users, security of these resources from attack is also important. To maintain fluency in service, achieve customer satisfaction, to maintain reputation and presence in the competitive market conditions the security of the resources is highest priority for cloud service provider. Hence development of more advanced techniques to detect and mitigate attack in cloud environment is crucial.

ACKNOWLEDGMENT

I am indeed thankful to my guide Prof. Amar Buchade for his guidance and assistance to complete this paper. I am grateful. for his valued support and faith on me. I extend my special thanks to Head of Department of Computer Engineering, Prof. Girish P. Potdar who extended the preparatory steps of this paper-work.

REFERENCES

[1] M. Cotton, L. Vegoda, IETF RFC 5735, January 2010.

[2] K. Park, H. Lee, ”On the Effectiveness of Route-Based Packet Filtering for Distributed DoS Attack Prevention in Power-Law Internets”, SIGCOMM01, August 2001.

[3] J. Wu, G. Ren, X. Li, ”Source Address Validation: Architecture and Protocol Design”, IEEE International Conference on Network Protocol, 2007.

[4] P. Ferguson, D. Senie, ”Network ingress filtering: defeating Denial of Service attacks which employ IP source address spoofing: RFC 2827”, May 2000.

[5] L. L. Peterson, B. S. Davie, ”Computer Networks: a systems approach”, 5th ed., Morgan Kaufmann, 2013.

[6] H. Wang, C. Jin, K. G. Shin ”Defense Against Spoofed IP Traffic Using Hop Count Filtering”, IEEE Transactions On Networking, VOL. 15, NO. 1 FEBRUARY 2007.

[7] R. Yamada, S. Goto ”Using abnormal TTL values to detect malicious IP packets”, Proceedings of the APAN Network Research Workshop 2012

[8] C. Yu and V. D. Gligor ”A specification and verification method for preventing denial of service”, IEEE Transaction on Software Engineering, 1990.

[9] Y. Huang and J. M. Pullen ”Countering Denial-of-Service Attacks Using Congestion Triggered Packet Sampling and Filtering”, 10th International Conference on Computer Communications and Networks 2001.

[10] W. Dou, Q. Chen, J. Chen ”A Confidence Based Filtering Method for DDoS Attack Defence In Cloud Environment”, Future Generation Computer Systems September 2013.

[11] T. Vissers, T. S. Somasundaram, L. Pieters, K. Govindarajan, P. Hellinckx ”DDoS Defense System For Web Service In Cloud Environment” Future Generation Computer Systems, March 2014.

[12] U. Tariq, Y. Malik, B. Abdulrazak ”Defence and Monitoring Model for Distributed Denial of Service Attack”, 2nd International Workshop on Internet of Ubiquitous and Pervasive Things June 2012.

[13] B. Forouzan, S. Fegan ”Data Communication and Networking”,4th ed. New York, USA: Mc Graw-Hill, 2007.

[14] P. Heegaard, K. Trivedi ”Network Survivability Modeling”, Performance Modeling of Computer Networks: Special Issue in Memory of Dr. Gunter Bolch, Science Direct, June 2009.

Akash B Mahagaonkar received Bachelor’s degree in Computer Engineering from Savitribai Phule Pune University in 2013 (email:akash.mahagaonkar@gmail.com). He is currently pursuing Master of Engineering from Savitribai Phule Pune University.

Amar Buchade presently working as assistant professor in the Department of Computer Engineering at Pune Institute of Computer Technology, Pune. He received B.E. and M.E. in Computer Engineering from WCOE, Sangli, India. His research area includes Distributed System, Cloud Computing and Security.