Survey on DDoS Attacks and its Detection & Defence Approaches

Nisha H. Bhandari

Abstract**—*****In Cloud environment, cloud servers providing requested cloud services, sometimes may crash after receiving huge amount of request. This situation is called Denial Of service attack. Cloud Computing is one of today's most exciting technologies due to its ability to reduce costs associated with computing while increasing flexibility and scalability for computer processes. Cloud Computing is changing the IT delivery model to provide on-demand self-service access to a shared pool of computing resources (physical and virtual) via broad network access to offer reduced costs, capacity utilization, higher efficiencies and mobility. Recently Distributed Denial of Service (DDoS) attacks on clouds has become one of the serious threats to this buzzing technology. Distributed Denial of Service (DDoS) attacks continue to plague the Internet. Distributed Denial-of-Service (DDoS) attacks are a significant problem because they are very hard to detect, there is no comprehensive solution and it can shut an organization off from the Internet. The primary goal of an attack is to deny the victim's access to a particular resource. In this paper, we want to review the current DoS and DDoS detection and defence mechanism.***

Index Terms**—*****Cloud Computing, Distributed Denial of Service (DDoS) attack, TTL, Hop-count, and packet marking.***

I. INTRODUCTION

Cloud computing is currently one of the most hyped information technology fields and it has become one of the fastest growing segments of IT. Cloud computing allows us to scale our servers in magnitude and availability in order to provide services to a greater number of end users. Moreover, adopters of the cloud service model are charged based on a pay-per-use basis of the cloud's server and network resources, aka utility computing.Cloud computing is a model of information processing, storage, and delivery in which physical resources are provided to clients on demand. Instead of purchasing actual physical devices servers, storage, or any networking equipment, clients lease these resources from a cloud provider as an outsourced service. It can also be defined as “management of resources, applications and information as services over the cloud (internet) on demand”. Cloud computing is a model for enabling convenient and on demand network access to a shared group of computing resources that can be rapidly released with minimal management effort or service provider interaction.[1] Cloud Computing provides different layers of computing utilities, from storage and networking to tools and applications, through three main service models: software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS).

DoS attacks do not wish to modify data or gain illegal Access, but instead they target to crash the servers and whole networks, disrupting legitimate users’ communication. DoS attacks can be launched from either a single source or multiple sources. Distributed denial-of-service (DDoS) attacks commonly overwhelm their victims by sending a vast amount of legitimate-like packets from multiple attack sites. As a consequence the victim spends its key resources processing the attack packets and cannot attend to its legitimate clients. During very large attacks, DDoS traffic also creates a heavy congestion in the Internet core which disrupts communication between all Internet users whose packets cross congested routers. [11].

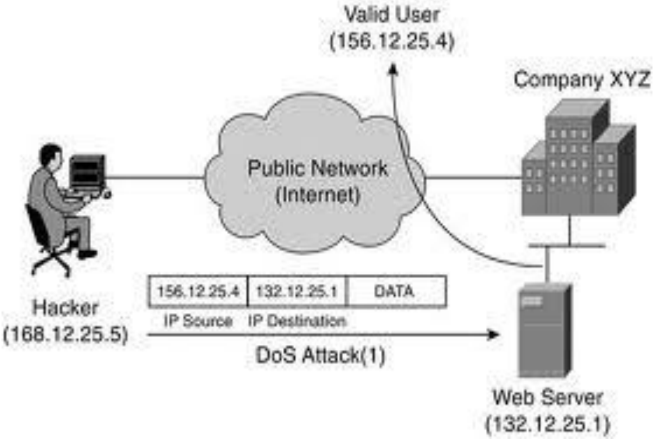

Figure 1DoS Using IP Spoofing

II. PROCEDURE FOR PAPER SUBMISSION

Innately, Internet is communication infrastructure for cloud providers that use well-known TCP/IP protocol which users' IP addresses to identify them in the Internet. Similar to copyright form and the form should accompany your final submission.Physical computer in the Internet that have IP address, a virtual machine in the Internet has an IP address as well. A malicious user, whether internal or external, like a legal user can find this IP addresses as well. In this case, malicious user can find out which physical servers the victim is using then by implanting a malicious virtual machine at that location to launch an attack [2]. Denial-of-Service (DDoS) attacks are a significant attacks in cloud environment. Distributed Denial-of-Service (DDoS) attacks are a significant problem because they are very hard to detect, there is no comprehensive solution and it can shut an organization off from the Internet. The primary goal of an attack is to deny the victim's access to a particular resource. It is very hard to detect DDoS attack when it is implemented using IP spoofing.

Manuscript received on February 2013.

Ms. Nisha H. Bhandari who is PG Scholar of Department of Computer Science & Technology, Gujarat Technological University, Ahmedabad, India.

IP spoofing has often been exploited by Distributed Denial of Service (DDoS) attacks to: 1) conceal flooding sources and dilute localities in flooding traffic, and 2) coax legitimate hosts into becoming reflectors, redirecting and amplifying flooding traffic. Thus, the ability to filter spoofed IP packets near victim servers is essential to their own protection and prevention of becoming involuntary DoS reflectors**.** DoS attack using IP spoofing is shown in fig1.Here we discuss various DDoS Attack which are implemented using IP Spoofing.

A. Smurf

The two main components to the Smurf Denial-of-Service attack are:

- The use of forged ICMP echo request packets.
- The direction of packets to IP broadcast addresses.

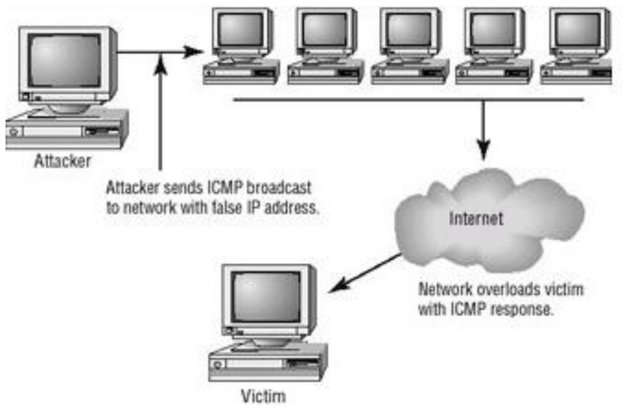

Figure 2 Smurf

The Internet Control Message Protocol (ICMP) is used to determine whether a machine on the Internet is responding properly or has connection problems. Also, ICMP can be used to handle errors and exchange control messages. To do these, an ICMP echo request packet is sent to a machine with a return address that the contacted machine will return an ICMP echo reply packet when receiving the ICMP echo request packet. On IP networks, a packet can be directed to an individual machine or broadcast to an entire network by using The IP broadcast address. In the Smurf attack, attackers are using the ICMP echo request packets directed to IP broadcast addresses from remote locations to generate DoS attacks.

When an attacker sends the ICMP echo request packets, most of the time, they create a forged packet using a spoofed IP address of attacker’s intended victim instead of his own IP address in order to hide their identity. The result is: when all the machines at the intermediary’s network respond to the ICMP echo requests, they all send replies to the victim’s machine. Although we have not intended to cause a problem on intermediary’s network, suffering similar types of traffic outbursts that a victim machine are suffering from can victimize the intermediary.[3]

B. TCP SYN Attack

TCP SYN attack is one of the most known and used resource depletion attacks. A SYN flood attack occurs during the three-way handshake that marks the onset of a TCP connection. In the three-way handshake, a client requests a new connection by sending a SYN packet to a server. After that, the server sends a SYN/ACK packet back to the client and places the connection request in a queue. Finally, the client acknowledges the SYN/ACK packet. If an attack occurs, however, the attacker sends an abundance of SYN packets to the victim, obliging it both to open a lot of TCP connections and to respond to them. Then the attacker does not execute the third step of the three-way handshake that follows, rendering the victim unable to accept any new incoming connections, because its queue is full of half-open TCP connections. Mostly the attacker sends a spoofed package to victim, what causes that the SYN/ACK package is send completely to other host, which do not respond because did not sent any SYN packets to the victim.[6].

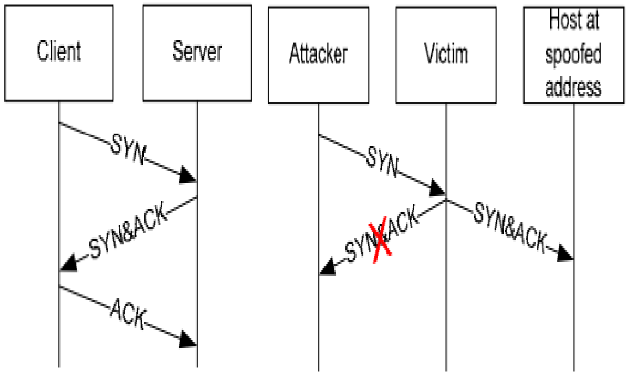

Figure 3 Package flow in three-way handshake (a) and TCP SYN attack (b)

C. UDP flood Attack

This is the second most popular DDoS attack method after TCP SYN flood. The basic idea in the UDP Flood attacks is to exploit UDP services, which are known to reply to packets. The hacker is armed with a list of broadcast addresses, to which sends spoofed UDP packets. These packets are sent to Random and changing ports of the unsuspected target location. In most of the cases the packets are directed to the echo port 7 (echoes any character it receives in an attempt to test network programs) on the target machines. However, there are attacks in which the malicious user sends packets to the chargen port. The chargen port is a port, which is used for testing purposes and generates a series of characters for each packet it receives. By connecting a host's chargen service to the echo service on the same or another machine, all affected machines can be effectively taken out of service as an excessively high number of packets are going to be produced. In addition, if two or more hosts are so connected, the intervening network can also become congested and deny service to all hosts whose traffic traverses that network (this attack generally works on NTboxes).It is obvious from the previous analysis that the result from a UDP flood attack is the creation of a nonstop flood of useless data passes between two or more systems. The target host returns ICMP port unreachable messages as a response to each spoofed UDP packets and then slows down because becomes more and more busy processing the forged IP addresses.

This "loop" is responsible for the overload of the network (may crawl to a stop) and the total exhaust of the available bandwidth. Victims of this massive amount of traffic can be also, except networks, individual system, which can lose connectivity to the Internet and in some cases, crash. [3].

III. CURRENT DETECTION AND DEFENCE MECHANISM

1) *Three-Way Handshake*

A simple solution to prevent source spoofing at end systems is to use three-way handshakes at the beginning of an interaction. If a source host spoofs its IP address, it will be unable to finish a three-way handshake. This solution works well to prevent source spoofing at end systems, but attackers are free to spoof the source address of the first packet of a three-way handshake, and they can launch DoS flooding attacks with these packets.[12].It is major drawback of given solution.

2) *Ingress /Egress Filtering*

Most of DoS and DDoS attacks use forged or spoofed source IP addresses in order to hide the attacker's originality and also indirectly generate the massive traffic from Intermediary network to target machine. As a result, a machine that the spoofed address is belonging to is also a victim of the DoS attack .A packet leaving to Internet and arriving from Internet must have a source address originating from interior network. By blocking packets with non-local source IP address from leaving an interior network, DoS attacker's source address spoofing become impossible those filtering can usually be implemented on edge routers as shown in figure 4. The edge routers need to be capable of examining the source address of every packet in real time and correctlydetermine which packet has a legitimate source IP address. However, even though this scheme is most feasible in customer network, the universal deployment is not likely to be

The edge routers need to be capable of examining the source address of every packet in real time and correctly determine which packet has a legitimate source IP address.

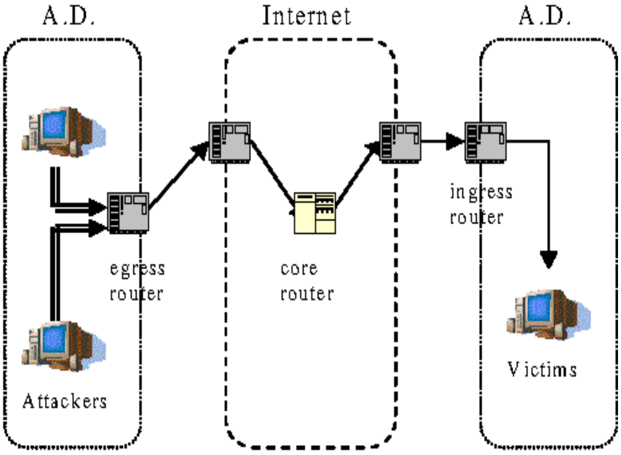

Figure 4 Ingress/Egress Filtering

However, even though this scheme is most feasible in customer network, the universal deployment is not likely to be accomplished because of administrative burden ,potential router overhead and complications with existing services that depend on source address spoofing .Moreover, if the interior network is quite large, or each sub-network does not have the address filtering capability ,the attackers could still forge addresses from hundreds of thousands of hosts within a valid interior network.[3].

3) *Hop Count Filtering*

Hop-Count Filtering is use to weed out spoofed IP packets at the very beginning of network processing, thus effectively protecting victim servers' resources from abuse.

The rationale behind hop-count filtering (HCF) is that most randomly spoofed packets, when arriving at victims, do not carry hop-count values that are consistent with the IP addresses being spoofed. The hop-count information is indirectly reflected in the Time-to-Live (TTL) field of the IP header, since each intermediate router decrements the TTL value by one before forwarding a packet to the next hop. As a receiver, an Internet server can infer the hop-count information and check for consistency of source IP addresses. Exploiting this observation, HCF builds an accurate IP-to-hop-count (IP2HC) mapping table, while using a moderate amount of storage, by clustering address prefixes based on hop-count. To capture hop-count changes under dynamic network conditions, it devise a safe update procedure for the IP2HCmapping table that prevents pollution by attackers. The same pollution-proof method is used for both initializing IP2HC mapping table and inserting additional IP addresses into the table. To minimize collateral damage, HCF has two running states, *learning* and *filtering*. Under normal conditions, HCF stays in the *learning* state, watching for the rationale behind hop-count filtering (HCF) is that most randomly spoofed IP packets, when arriving at victims, do not carry hop-count values that are consistent with the IP addresses being spoofed. The hop-count information is indirectly reflected in the Time-to-Live (TTL) field of the IP header, since each intermediate router decrements the TTL value by one before forwarding a packet to the next hop. As a receiver, an Internet server can infer the hop-count information and check for consistency of source IP addresses. Exploiting this observation, HCF builds an accurate IP-to-hop-count (IP2HC) mapping table, while using a moderate amount of storage, by clustering address prefixes based on hop-count. To capture hop-count changes under dynamic network conditions, It devise a safe update procedure for the IP2HC mapping table that prevents pollution by attackers. The same pollution-proof method is used for both initializing IP2HC mapping table and inserting additional IP addresses into the table. To minimize collateral damage, HCF has two running states, *learning* and *filtering*. Under normal conditions, HCF stays in the *learning* state, watching for abnormal TTL behaviours without discarding any packets. Even if a legitimate packet is incorrectly identified as spoofed, it will not be dropped. Therefore, there is *no* collateral damage in the *learning* state. Upon detection of an attack, HCF switches to the *filtering* state, in which HCF discards those IP packets with mismatching hop-counts.. HCF has its own limitations. An attacker may circumvent HCF entirely by not using spoofed traffic, or partially by bombarding a victim with much more attacking traffic than seen before. Also, a "determined" attacker may find a way to build an IP2HC mapping table that is accurate enough for most spoofed IP packets to evade HCF.

Moreover, the actual deployment of those legitimate packets that are incorrectly identified as spoofed. HCF requires further work in tuning its parameters and handling the IP2HC inaccuracy caused by the Network Address Translator (NAT) boxes and possible hop-count instability. Nevertheless, HCF does greatly enhance the capability to counter DDoS attacks by depriving an attacker of his powerful weapon, random IP spoofing [8].

4) IP Trace back

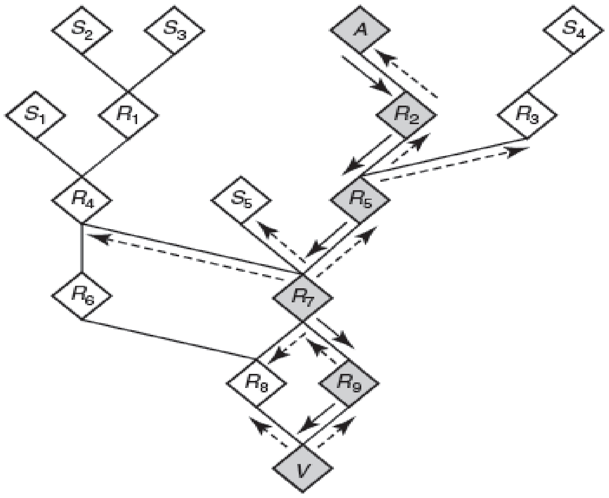

Figure 5 IP Trace back

Most existing trace back techniques start from the router closest to the victim and interactively test its upstream links until they determine which one is used to carry the attacker's traffic. Ideally this procedure is repeated recursively on the upstream router until the source of traffics' reached. This technique has a critical assumption that an attacker will be remained while tracing mechanism is completed. However, the most attacks are relatively short: 50% of attacks are less than 10 minutes, 80% are less than 30 minutes and90% last less than an hour. Therefore, using this technique is not feasible in real time detection system. Another drawback of this scheme is that even though we use an automated tool, it creates enormous network and management overhead such as storing packet information along with routing paths, and inter-communication between routers of different ISPs.

5) Pi: A Path Identification Mechanism

Pi (short for Path Identifier), a new packet marking approach in which a path fingerprint is embedded in each packet, enabling a victim to identify packets traversing the same paths through the Internet on a per packet basis, regardless of source IP address spoofing. In *PI method* which routers mark information on packets en-route to the victim, who can then use that information to reconstruct the path that the packets take from the attacker through the Internet, despite IP address spoofing? The path information obtained by the trace back mechanism can then be used to install network filters upstream from the victim to block attack traffic. Embeds in each packet an identifier based on the router path that a packet traverses. The victim need only classify a single packet as malicious to be able to filter out all subsequent packets with the same marking. The Pi mark is deterministic, so that a marking for a particular path remains the same and each packet traversing that path will carry the same mark. The Pi mark is generated piecemeal by the routers along the path from end-host to victim. Each packet from the attackers could be identified by this *distinctive marking*, then the victim could drop such packets with the same marking by comparing incoming packet markings against the markings in the attack list. Thus it allow the victim to develop rapidly responsive packet filters to protect itself during such attacks. So in this approach DDoS attack is modelled in two phases. In the first phase, the *learning phase*, all packets are assumed to be analysed by the victim, using the packet identification function that determines whether the packet is an attack packet or a legitimate user's packet. In other words, the victim is temporarily given the power to differentiate between legitimate users' packets and attackers' packets. The victim is thus able to generate an attack markings list. In the second phase, the *attack phase*, the victim is presumably no longer able to apply its packet identification function and is forced to use the Pi filter based on the information it has gathered in the learning phase. Pi marking is the most general, flexible and powerful of the packet marking schemes to date, and shows significant potential in reducing or eliminating the DDoS threat. [4]

6) Traffic Analysis

Many researchers in the academic fields have proposed many different approaches to analyse the traffic patterns in order to inferring attacking packets and its characteristics. Most of methods are detecting the pattern of illegitimate packets or their source using probabilistic and statistical analysis. A critical point of those researches is that a large scale attacks can readily be identified by observing very abrupt changes in the network traffics and most of packets will have a certain type of pattern so that we can classify them according to each pattern. For example, at first, randomly collect sample, classify the collected packets, and then normalize the data or build a temporary DB. Second, using a specific algorithm and modelling, find a pattern for bad-will packets from the data sampling. Most of differentiated methods are developed in second phase such as using Time series, Data Mining and more complicated mathematical models.However, even though those methods can provide quite reasonable solution to detect bad-will packets, we cannot be fully confident that every attacking packet can be detected or only illegitimate packets are likely to be detected since these mechanisms are relying on probabilistic model [3].

IV. OBSERVATION

We can observe that many of the methods need to be implemented simultaneously and collaboratively on several nodes, making them difficult to implement. In Path Identification technique router's IP address that the Pi uses to mark the path is too large to write into the packet's limited space. The disadvantage of writing routers' IP addresses into the limited space may result the same path identification for different paths. With these observations and concerns in mind, implementing an effective defines method becomes a critical investment that requires serious consideration to reach a balance between benefits and costs: the location, simplicity, performance, and cost of a defence system are correlated and an efficient system is one which optimizes these factors.

V. CONCLUSION

The survey of the all relevance detection and defense techniques against DDoS with IP spoofing we can conclude that methods are differ in their region of action, the amount of legitimate traffic they drop, their ease of implementation, and the type of attack they are effective against, each method has certain features that make it more suitable to implement in one situation than another.

REFERENCES

1. Ayesha Malik, Muhammad MohsinNazir Security Framework for Cloud Computing Environment: A ReviewJournal of Emerging Trends in Computing and Information SciencesVOL. 3, NO. 3, March 2012
2. FarzadSabahi Cloud Computing Security Threats and Responses
3. GulshanShrivastava and KavitaSharma ,"The Detection & Defense of DoS & DDoS Attack:A Technical Overview" Proceeding of ICC, 27-28 December 2010
4. Yaar, A., A. Perrig and D. Song, 2003. Pi: A Path Identification Mechanism to Defend against DDoSAttacks. Proceedings of Symposium on Security and Privacy, pp: 93-107
5. *Wesley M. Eddy, Verizon Federal Network Systems*"Defense Against TCP SYN Flooding Attack " December 2004 Available: http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_9-4/syn_flooding_attacks.html
6. Simona RAMANAUSKAITĖ "Modeling and research of Distributed Dienal of service attack "Available: http://vddb.laba.lt/fedora/get/LT-eLABa-001:E.02~2012~D_20120723_10503170003/DS.005.1.01.ETD
7. Fu-Yuan Lee *, ShiuhpyngShiehDefending against spoofed DDoS Attack s with path fingerprint International Journal of Computer application (0975 – 8887)
8. W. Haining, et al., "Defense Against Spoofed IP Traffic Using Hop-Count Filtering," Networking, IEEE/ACM Transactions on, vol. 15, pp. 40-53, 2007
9. P. A. R. Kumar and S. Selvakumar, "Distributed Denial-of-Service (DDoS) Threat in Collaborative Environment - A Survey on DDoS Attack Tools and Traceback Mechanisms," in Advance Computing Conference, 2009. IACC 2009. IEEE International, 2009, pp. 1275-1280.
10. I. B. Mopari, et al., "Detection and defense against DDoS attack with IP spoofing," in Computing, Communication and Networking, 2008. ICCCn 2008. International Conference on, 2008, pp. 1-5.
11. JelenaMirkovic, Max Robinson, Peter Reiher, George Oikonomou ,"Distributed Defense Against DDoS Attacks" Available:http://www.isi.edu/~mirkovic/publications/udel_tech_report_2005-02.pdf
12. XinLiu ,"Mitigating Denial-of-Service Flooding Attacks with Source Authentication" Available:http://www.cs.duke.edu/~xinl/dissertation.pdf