A Comprehensive Study on Distributed Denial of Service Attacks and Defense Mechanisms

Pavithra K.C
IV Semester, M.Tech.
Dept. of CSE
MITE, Moodabidri

Snitha Shetty
IV Semester, M Tech.
Dept. of CSE
MITE, Moodabidri

Dr. Nagesh H.R
Professor & Head
Dept. of CSE
MITE, Moodabidri

ABSTRACT
The advances in information technologies in the internet are increasing the possibility of attacks exponentially. Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks are increasing rapidly across the internet world. Denial of Service (DoS) is an attack on availability of a service. The attack aims at denying of an approved service to a legitimate user. When a group of attackers perform DoS attack on a common target, the attack is known as DDoS. The most common method of performing a DoS or a DDoS attacks is to flood the target or network with unwanted traffic, causing interruptions to the communication of legitimate users. The attacks are evolving in a way, the frequency and the severity, sophistication of DDoS attacks are increasing very fast. Existing methods of DoS/DDoS attack, defence mechanisms are outdated and even the latest surveys do not reflect the significant developments in this area in recent years. This paper will explain, in simple terms, the various types of DDoS attacks and the technologies that are used to prevent and mitigate the attacks. This comprehensive study of attacks and their defence mechanisms would provide the researchers with a better understanding of the problem and the possible solutions.

Keywords
DoS, DDoS, Botnet, Defence, IP spoofing.

1. INTRODUCTION
Internet was designed to allow easy sharing of information between various interconnected computers and networks. It was not designed with security technologies to safeguard the information. There are pathogens and digital equivalents of viruses, which cause a major threat to the valuable information that is available. The Internet has become a channel for people and businesses to regularly access useful information. The limitation of internet is that it is vulnerable to disruption. Malicious users are often able to obtain confidential information or halt normal computer operation, with various motives ranging from financial greed, revenge and political aims.

DoS and DDoS attacks cause lot of trouble around the world daily, by causing significant downtime for websites or using disruption to breach security, causing reputational and financial damage. During DoS attacks, attackers bombard their target with a massive amount of requests or data exhausting its computing resources or network and preventing legitimate users from having access. More simply, a DoS attack is when an attacker uses a single machine's resources to exhaust those of another machine, in order to prevent it from functioning normally. When the traffic of a DoS attack comes from multiple sources, it is called a Distributed Denial of Service (DDoS) attack. The result of a DDoS attack is amplified and the problem of defence is made tougher by using multiple attackers. In such cases, it is harder to detect and block attackers manually, so special defences are needed to detect and defend against such large-scale attacks. Attackers infect thousands of computers spread across the world with sophisticated malware technologies in order to gain unauthorized access to such systems. Attackers do not really control their attacking machines.Thousands of compromised machines act as an army under the command of one attacker. Such a collection of thousands of machines is called a "botnet", and often the actual owners of machines that are part of a botnet are unaware that their computers have been compromised and are being used to launch DDoS attacks [20]. The impact of DDoS attacks can vary from causing minor delay to users of a website, to serious financial losses for companies that rely on their online availability to do business.

It is necessary for an organization to protect itself against DoS and DDoS attacks. Without proper protection mechanisms, an organization targeted by a DoS or DDoS attack is likely to experience damage to its reputation, financial losses and legal expense, all of which are likely to affect its future. This study presents various types of DDoS attacks and techniques for defending against DoS and DDoS attacks.

2. EVOLUTION OF DDoS
The first DoS attack occurred in 1974 and was carried out by a 13 year old student named David Dennis at the University of Illinois Urbana Champaign. David had learned about a new command called "external" or "ext", which could be run on PLATO terminals. It allowed for interaction with external devices connected to the terminals. When run on a terminal with no external devices attached, however, it would cause the terminal to lock up and require a shutdown and power-on to restart its operation.

During the mid-to-late 1990s, when Internet Relay Chat (IRC) was becoming popular, some users fought for control of non-registered chat channels, where an administrative user would lose his or her powers if he or she logged off. This behavior led hackers to attempt to force all users within a channel to log out, so they could enter the channel alone and gain administrator privileges as the only user. One of the first large-scale DDoS attacks occurred in August 1999, when a hacker used a tool called "Trinoo" to disable the University of Minnesota's computer network for over two days. Trinoo was basic and without any anonymity features; it consisted of a network of compromised machines called "Masters" and "Daemons", allowing an attacker to send a DoS instruction to a few Masters, which then forwarded instructions to the hundreds of Daemons to commence a UDP flood against the target IP address.

During February 2000, DDoS attacks truly caught the public's attention. Several of the most well-known Internet sites at the time were targeted, including Yahoo, CNN, Amazon, Buy.com, E*Trade, and ZDNet. Even the Website of the FBI, the foremost prosecutor of cybercrime, was brought offline for three hours by a DDoS attack. Despite this, each targeted Website experienced some level of downtime as a result of the February 2000 DDoS attacks. Another notable DDoS attack that took place during the early 2000s targeted all 13 of the Internet's root domain name service (DNS) servers in 2002[21]. DNS is a hierarchical system, as smaller DNS servers rely on other larger DNS servers; on the highest level there are 13 root name servers, without which the world's DNS system would fail. The effect of a powerful DDoS attack on all 13 of the root name servers simultaneously would be catastrophic. Internet browsing would be slow or even unusable for everyone in the world. During the 2002 attack on the root name servers, all 13 servers experienced heavy load, and some of them were unreachable from parts of the global Internet. DDoS attacks are larger, stealthier, more targeted, and more sophisticated than ever. Given the extraordinary and rapid changes in the DDoS terrain, traditional DDoS mitigation tactics such as bandwidth over-provisioning, firewalls, and intrusion prevention system (IPS) devices are no longer sufficient to protect an organization's networks, applications, and services.

3. TYPES OF ATTACK

Classifying the different types of DoS and DDoS attacks is important in order to provide protection mechanisms to underlying systems. Each type of attack has different characteristics that may suggest it belongs to multiple categories. Generally speaking, types of attacks include those that target network resources, those that target server resources, and those that target application resources. The following is a list of some the most common attacks and their technical underpinnings.

3.1 Attacks Targeting Network Resources

Attacks that target network resources attempt to consume all of a victim's network bandwidth by using a large volume of illegitimate traffic to saturate the company's Internet pipe. Attacks of this manner, called network floods, are simple yet effective. In a typical flooding attack, the offence is distributed among an army of thousands of volunteered or compromised computers – a botnet – that simply sends a huge amount of traffic to the targeted site, overwhelming its network.

3.1.1 UDP Flood

User Datagram Protocol (UDP) is a connectionless protocol that uses datagrams embedded in IP packets for communication without needing to create a session between two devices. A UDP Flood attack does not exploit a specific vulnerability, but rather simply abuses normal behavior at a high enough level to cause network congestion for a targeted network. It consists of sending a large number of UDP datagrams from potentially spoofed IP addresses to different ports on a victim server; the server receiving this traffic is unable to process every request, and consumes all of its bandwidth attempting to send ICMP "destination unreachable" packet replies to confirm that there was no application listening on the targeted ports.

3.1.2 ICMP Flood

Internet Control Message Protocol (ICMP) is another connectionless protocol used for IP operations, diagnostics, and errors. Just as with a UDP flood, an ICMP flood (or Ping Flood) is a non-vulnerability based attack; that is, it does not rely on any specific vulnerability to achieve denial-of-service. An ICMP Flood can involve any type of ICMP message of echo request; once enough ICMP traffic is sent to a target server, it becomes overwhelmed from attempting to process every request, resulting in a denial-of-service condition.

3.1.3 IGMP Flood

Internet Group Management Protocol (IGMP) is yet another connectionless protocol, used by IP hosts to report or leave their multicast group memberships for adjacent routers. An IGMP Flood is non-vulnerability based, as IGMP allows multicast by design. Such floods involve a large number of IGMP message reports being sent to a network or router, significantly slowing down and eventually preventing legitimate traffic from being transmitted across the target network.

3.2 Attacks Targeting Server Resources

Attacks that target server resources attempt to exhaust a server's processing capabilities or memory, potentially causing a denial-of service condition. The idea is that an attacker can take advantage of an existing vulnerability on the target server (or a weakness in a communication protocol) in order to cause the target server to become busy handling illegitimate requests so that it no longer has the resources to handle legitimate ones.

3.2.1 TCP/IP Weaknesses

These types of attacks abuse the TCP/IP protocol by taking advantage of some of its design weaknesses. They typically misuse the six control bits (or flags) of the TCP/IP protocol – SYN, ACK, RST, PSH, FIN, and URG – thereby disrupt the normal mechanisms of TCP traffic. TCP/IP relies on a three-way handshake mechanism (SYN, SYN-ACK, and ACK) where every request creates a half-open connection (SYN), a request for a reply (SYN-ACK), and then an acknowledgement of the reply (ACK). Any attack that attempts to abuse the TCP/IP protocol will often involve sending TCP packets in the wrong order, causing the target server to run out of computing resources as it attempts to understand such abnormal traffic.

3.2.2 TCP SYN Flood

In the TCP handshake mechanism, there must be an agreement between each party for a connection to be established. If the TCP client does not exist or is a non-requesting client with a spoofed IP, such an agreement is not possible. In a TCP SYN, or simply SYN flood attack, the attacking clients lead the server to believe that they are asking for legitimate connections through a series of TCP requests with TCP flags set to SYN, coming from spoofed IP addresses. To handle each of these SYN requests, the target server opens threads and allocates corresponding buffers to prepare for a connection. It then tries to send a SYN-ACK reply back to the requesting clients to acknowledge their connection requests, but because the clients IP addresses are spoofed or the clients are unable to respond, an acknowledgement (ACK packet) is never sent back to the server. The server is still forced to maintain its open threads and buffers for each one of the original connection requests, attempting to resend its SYN-ACK request acknowledgement packets multiple times before resorting to a request time-out. Because server resources are limited and a SYN flood often involves a massive number of connection requests, a server is unable to time-out its open requests before even more new requests arrive, and this causes a denial-of-service condition.

3.2.3 TCP RST Attack

The TCP RST flag is intended to notify a server that it should immediately reset its corresponding TCP connection. In a TCP RST attack, the attacker interferes with an active TCP connection between two entities by guessing the current sequence number and spoofing a TCP RST packet to use the client's source IP (which is then sent to the server). A botnet is usually used to send thousands of such packets to the server with different sequence numbers, making it fairly easy to guess the correct one. Once this occurs, the server acknowledges the RST packet sent by the attacker, terminating its connection to the client located at the spoofed IP address.

3.2.4 TCP PSH+ACK Flood

When a TCP sender sends a packet with its PUSH flag set to 1, the result is that the TCP data is immediately sent or "pushed" to the TCP receiver. This action actually forces the receiving server to empty its TCP stack buffer and to send an acknowledgement when this action is complete. An attacker, usually using a botnet, can therefore flood a target server with many such requests. This overwhelms the TCP stack buffer on the target server, causing it to be unable to process the requests or even acknowledge them.

3.2.5 SSL based Attacks

With the rise of Secure Socket Layer (SSL), a method of encryption used by various other network communication protocols, attackers have begun to target it. SSL runs above TCP/IP conceptually, and provides security to users communicating over other protocols by encrypting their communications and authenticating communicating parties. SSL-based attacks could also simply mean that the DoS attack is launched over SSL-encrypted traffic, which makes it extremely difficult to identify; such attacks are often considered "asymmetric", as it takes significantly more server resources to deal with an SSL-based attack than it does to launch one.

3.3 Attacks Targeting Application Resources

Instances of DoS attacks that target application resources have grown recently and are widely used by attackers today. They target not only the well-known Hypertext Transfer Protocol (HTTP), but also HTTPS, DNS, SMTP, FTP, VOIP, and other application protocols that possess exploitable weaknesses allowing for DoS attacks. Just as attacks that target network resources, there are different types of attacks that target application resources, including both floods and "low and slow" attacks.

3.3.1 HTTP flood

An HTTP flood is the most common application-resource-targeting DDoS attack. It consists of what seem to be legitimate, session based sets of HTTP GET or POST requests sent to a victim's Web server, making it hard to detect. HTTP flood attacks are typically launched simultaneously from multiple computers (volunteered machines or bots), that continually and repeatedly request to download the target site's pages (HTTP GET flood), exhausting application resources and resulting in a denial-of-service condition.

3.3.2 DNS flood

A DNS flood is easy to launch, yet difficult to detect. Based on the same idea as other flooding attacks, a DNS flood targets the DNS application protocol by sending a high volume of DNS requests. The DNS server, overwhelmed and unable to process all of its incoming requests, eventually crashes.Domain Name System (DNS) is the protocol used to resolve domain names into IP addresses; its underlying protocol is UDP, taking advantage of fast request and response times without the overhead of having to establish connections (as TCP requires).

3.3.3 Hash Collisions Dos Attack

This kind of attack targets common security vulnerabilities in Web application frameworks. Collision resolutions are resource intensive, as they require an additional amount of CPU to process the requests. In a Hash Collision DoS attack scenario, the attacker sends a specially crafted POST message with a multitude of parameters. The parameters are built in a way that causes hash collisions on the server side, slowing down the response processing dramatically. Hash Collisions DoS attacks are very effective and could be launched from a single attacker computer, slowly exhausting the application server's resources.

4. ATTACK TOOLS

The previous chapters discussed various types of DDoS attacks occurring on both the network and application layers. While it is possible to execute many of these attacks manually, specialized attack tools have been developed for the purpose of executing attacks more easily and efficiently. The first DDoS tools – examples of which include Trinoo and Stacheldraht – were widely used around the turn of the century, but were somewhat complex and only run on the Linux and Solaris operating systems.

In more recent years, DDoS tools have become much more straightforward to use and cross-platform, rendering DDoS attacks much easier to carry out for attackers and more dangerous for targets. Some of these newer DDoS tools, such as Low Orbit Ion Cannon (LOIC), were originally developed as network stress testing tools and later modified and used for malicious purposes, while others such as Slowloris were developed by "gray hat" hackers – those aiming to draw the public's attention to a particular software weakness by releasing such tools publicly so the makers of the vulnerable software would be forced to patch it in order to avoid large-scale attacks

Low Orbit Ion Cannon (LOIC) "Hacktivist" group Anonymous' original tool of choice – Low Orbit Ion Cannon (LOIC) – is a simple flooding tool, able to generate massive amounts of TCP, UDP,

or HTTP traffic in order to subject a server to a heavy network load.

4.1 High Orbit Ion Cannon (HOIC)

After Anonymous "officially" dropped LOIC as its tool of choice, LOIC's "successor", "High Orbit Ion Cannon (HOIC), quickly took the spotlight when it was used to target the United States Department of Justice in response to its decision to take down Megaupload.com. While HOIC is also a simple application at its core – a cross-platform Basic script for sending HTTP POST and GET requests wrapped in an easy-to-use GUI – its effectiveness stems from its add-on "booster" scripts, or additional text files that contain additional Basic code interpreted by the main application upon a user's launch of an attack.

4.2 Hping

In addition to LOIC and HOIC, Anonymous and other hacking groups and individuals have employed various other tools to launch DDoS attacks, especially due to the Ion Cannons' lack of anonymity. One such tool, hping, is a fairly basic command line utility similar to the ping utility; however, it has more functionality than the sending of a simple ICMP echo request that is the traditional use of ping. hping can be used to send large volumes of TCP traffic at a target while

spoofing the source IP address, making it appear random or even originating from a specific user-defined source.

4.3 Slowloris

Besides straightforward brute-force flood attacks, many of the more intricate "low and slow" attack types have been wrapped up into easy- to-use tools, making for denial-of-service attacks that are much harder to detect. Slowloris, a tool developed by a gray hat hacker who goes by the handle "RSnake", is able to create a denial-of-service condition for a server by using a very slow HTTP request. By sending HTTP headers to the target site in tiny chunks as slow as possible the server is forced to continue to wait for the headers to arrive. If enough connections are opened to the server in this fashion, it is quickly unable to handle legitimate requests.

4.4 R U Dead Yet? (R.U.D.Y.)

Another slow-rate denial-of-service tool similar to Slowloris is R U Dead Yet? (R.U.D.Y.). R.U.D.Y. achieves denial of service by using long form field HTTP POST submissions rather than HTTP headers, as Slowloris does. By injecting one byte of information into an application POST field at a time and then waiting, R.U.D.Y. causes application threads to await the end of never-ending posts in order to perform processing. Since R.U.D.Y. causes the target Webserver to hang while waiting for the rest of an HTTP POST request, a user is able to create many simultaneous connections to the server with R.U.D.Y., ultimately exhausting the server's connection table and causing a denial-of-service condition.

4.5 The Botnet as a DDoS Tool

Regardless of the attack tool used the ability to launch an attack from multiple computers – whether it is hundreds, thousands, or millions – significantly amplifies the potential of an attack to cause denial-of-service. Attackers often have 'botnets' at their disposal – large collection of compromised computers, often referred to as 'zombies', infected with malware that allows an attacker to control them. Botnet owners, or 'herders', are able to control the machines in their botnet by means of a covert channel such as IRC (Internet Relay Chat), issuing commands to perform malicious activities such as distributed denial-of-service (DDoS) attacks, sending spam mail, and information theft.

5. EXISTING DDOS ATTACK DEFENCE PROPOSALS

Generally, there are four broad categories of defence against DDoS attacks: (1) attack prevention, (2) attack detection, (3) attack source identification, and (4) attack reaction. Attack prevention aims to stop attacks before they can reach their target. In the context of this study, it refers to filtering spoofed packets close to or at the attack sources, which is one of the most effective defence approaches for DDoS attacks that use spoofed traffic. Attack detection aims to detect DDoS attacks when they occur. Attack source identification aims to locate the attack sources regardless of whether the source address field in each packet contains erroneous information. It is a crucial step to minimize the attack damage and provide deterrence to potential attackers. Attack reaction aims to eliminate or curtail the effects of an attack. It is the final step in defending against DDoS attacks, and therefore determines the overall performance of the defense mechanism. The challenge for attack reaction is how to filter the attack traffic without disturbing legitimate traffic.

5.1 Attack Prevention

Attack Prevention aims to stop attacks before they actually cause damage. This approach assumes the source address of attack traffic is spoofed, which is true in many situations since attackers need spoofed traffic to hide the real source of the attack traffic and exploit protocol vulnerabilities. This approach normally comprises a variety of packet filtering schemes, which are deployed in routers. The packet filters are used to make sure that only valid (non- spoofed) traffic can pass through. This greatly reduces the chance of DDoS attacks occurring. However, it is not easy to specify a filtering rule that can differentiate spoofed traffic from legitimate traffic accurately.

5.1.1 Ingress/Egress Filtering

Ingress filtering means filtering the traffic coming into your local network, and egress filtering means filtering the traffic leaving your local network. [1].

5.1.2 Router-based Packet Filtering

Router-based Packet Filtering (RPF) by Park and Lee [2] extends ingress filtering to the core of the Internet. It is based on the principle that for each link in the core of the Internet, there is only a limited set of source addresses from which traffic on the link could have originated. If an unexpected source address appears in an IP packet on a link, then it is assumed that the source address has been spoofed, and hence the packet can be filtered.

5.1.3 Source Address Validity Enforcement (SAVE) Protocol.

The router-based packet filter is vulnerable to asymmetrical and dynamic Internet routing as it does not provide a scheme to update the routing information. To overcome this disadvantage, Li et al. have proposed a new protocol called the Source Address Validity Enforcement (SAVE) protocol [3], which enables routers to update the information of expected source IP addresses on each link and block any IP packet with an unexpected source IP address.

5.2 Attack Detection

After attack prevention, the next step in defending against DoS attacks is attack detection. A critical measure of performance for any detection scheme is its coverage, i.e., what proportion of actual attacks can be detected. Generally there are two groups of DoS attack detection techniques. The first group is called DoS-attack-specific detection, which is based on the special features of DoS attacks. The second group is called anomaly-based detection, which models the behavior of normal traffic, and then reports any anomalies.

5.2.1 DoS-Attack-Specific Detection.

Generally, DoS attack traffic is created at an attacker's will. First, attackers want to send as much traffic as possible to make an attack powerful. Hence, attack traffic does not observe any traffic control protocols, such as TCP flow control. Second, attack traffic is created in a random pattern to make an attack anonymous. Third, for each known attack, attack traffic at the target is highly correlated with abnormal traffic behavior at the attack sources. Gil and Poletto propose a scheme called MULTOPS [4] to detect denial of service attacks by monitoring the packet rate in both the up and down links. Wang et al. [5] proposed SYN detection to detect SYN floods, and Blazek et al. [6] proposed batch detection to detect DoS attacks. Both methods detect DoS attacks by monitoring statistical changes.

Generally, DoS attack flows are not regulated by TCP flow control protocols as normal flows do. Based on this assumption, Cheng et al. propose to use spectral analysis [7] to identify DoS attack flows. In this approach, the number of packet arrivals in a fixed interval is used as the signal. Based on the strong correlation between traffic behavior at the target and the attack source, Cabrera et al. [8] have proposed a

scheme to proactively detect DDoS attacks using time series analysis.

A new DoS attack detection scheme using source IP address monitoring is presented in [9]. Generally, the set of source IP addresses that is seen during normal operation tends to remain stable. In contrast, during DoS attacks, most of the source IP addresses have not been seen before. By using a carefully pre-built IP Address Database, it is possible to sequentially monitor the proportion of new source IP addresses seen by the target, and detect any abrupt change using a statistical test called Cumulative Sum (CUSUM) [10]. An abrupt change of the proportion of new source IP addresses is a strong indication of a DoS attack.

5.2.2 Anomaly-based Detection.

Signature-based detection and anomaly-based detection are two different approaches for network-based intrusion detection systems (IDS). Signature-based detection can identify an attack if the monitored traffic matches known characteristics of malicious activity. It is relatively easy for attackers to vary the type and content of attack traffic, which makes it difficult to design accurate signatures for DoS attacks [11]. While signature-based detection can be used to detect communication between attackers and their 'zombie' computers for known attack tools [12], in many cases this communication is encrypted, rendering signature-based detection ineffective. In contrast, anomaly-based detection can identify an attack if the monitored traffic behavior does not match the normal traffic profile that is built using training data. In 1987, Denning [13] first proposed a real-time intrusion detection model to detect attacks by monitoring a system's audit records for abnormal patterns of system usage.

5.3 Attack Source Identification

Once an attack has been detected, an ideal response would be to block the attack traffic at its source. Unfortunately, there is no easy way to track IP traffic to its source. This is due to two aspects of the IP protocol. The first is the ease with which IP source addresses can be forged. The second is the stateless nature of IP routing, where routers normally know only the next hop for forwarding a packet, rather than the complete end-to-end route taken by each packet. In order to address this limitation, many schemes based on enhanced router functions or modification of the current protocols have been proposed to support IP traceability.

5.3.1 IP Traceback by Active Interaction.

The main feature for IP traceback schemes in this category is that routers actively interfere with the attack traffic and trace the attack sources based on the reaction of attack traffic. Backscatter traceback [14] is a traceback scheme based on the observation that DoS attacks generally use invalid spoofed source IP addresses. Burch and Cheswick [15] proposed a link-testing traceback technique. This scheme requires considerable knowledge of network topology and the ability to generate huge traffic in any network link. Generally, high-speed routers lack tracking ability, such as the ability to tell from which link a packet comes.

5.3.2 Probabilistic IP Traceback Schemes.

The general idea of all probabilistic IP traceback schemes is that routers probabilistically insert partial path information into the incoming traffic, and the target reconstructs the packet path using the partial path information. Savage et al. proposed to traceback the IP source by probabilistic packet marking (PPM) [16]. The main idea of PPM is that each router embeds its IP address (partial path information) into the incoming packets probabilistically while they travel between the source and the destination. Based on the embedded path information, a target can reconstruct the packet transmission path.

Song et al. have improved the efficiency and security of the PPM scheme by introducing a new hashing scheme to encode the path information, and an authentication scheme to ensure the integrity of the marking information [17]. More details about PPM can be found in [16]. To prevent attackers from spoofing the ICMP packets, an authentication field is used in the iTrace packet. This scheme was later improved by Wu et al. [18].

5.3.3 Hash-based IP Traceback.

As discussed before, all the probabilistic approaches fail to identify attack paths when attack traffic is very scarce on each independent link during a highly distributed denial of service attack. Similarly, probabilistic approaches also fail to trace back the attack source, where the attack only contains a small number of packets. For example, the "ping-of-death" attack only needs one sufficiently long ICMP packet that is fragmented into multiple datagrams in order to attack a vulnerable target. Consequently, a better traceback approach is needed, such that it is not affected by traffic volume and is able to trace back even one single packet.

Snoeren et al. [19] proposed a scheme, called hash-based IP traceback, to trace individual packets. In this proposal, routers keep a record of every packet passing through the router.

5.4 Attack Reaction

Unlike more subtle attacks, such as remote-to-local attacks, DoS attacks try to damage the target as much as possible and attackers do not attempt to disguise the attack since the target will be aware of the attack damage eventually. In order to minimize the loss caused by DoS attacks, a reaction scheme must be employed when an attack is underway.

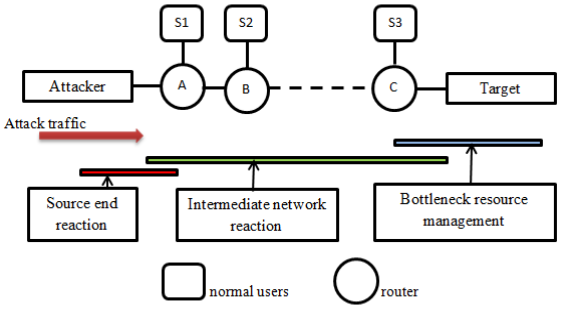

Fig 1: A model of DoS attacks reaction schemes

Consider a DoS attack whose aim is to congest the target's communication channel, which includes the target and the network links to which the target is connected. Figure 1 shows a simple model of a DoS attack, where thick lines represent high-bandwidth links and thin lines represent low-bandwidth links. The bottleneck of a target's communication channel can be caused by low-bandwidth network links as well as poorly-provisioned hosts. DoS attacks take effect once the resource limit of a bottleneck is reached. Hence, to minimize attack damage, the initial attack reaction is to protect the bottleneck's resources, which is called bottleneck resource management. Once the bottleneck resource is protected, the target is able to restore partial service instead of being completely paralyzed by the attack. If the attack volume is large enough, new bottlenecks will appear, even though the original bottleneck has been protected. As shown in Figure 1, the link between router C and the target is the bottleneck. Attack damage can be alleviated if bottleneck resource management schemes are used to protect this link. However, when the attack traffic volume is excessively high, the bandwidth limit of link A-B will be reached, and normal users S1 and S2 will fail to access the target. To protect S1 and S2, attack reaction should be applied at router A. We define

intermediate network reaction as the attack reaction taken at the routers between the attacker and the victim. In an ideal situation, attack traffic should be filtered at the source (Router A), which is called source end reaction.

6. CONCLUSION

DDoS attackers exploit flaws in protocols and systems to deny access of target services. Attackers also control a large number of compromised hosts to launch DDoS attacks. DDoS attacks are a complex and serious problem, and consequently, numerous approaches have been proposed to counter them. The attacks described here are intended to help the community think about the threats we face and the measures we can use to counter those threats. One positive benefit we foresee from this paper is to foster easier cooperation among researchers on DDoS defence mechanisms. We do not claim that these attack types are complete and all-encompassing. We must not be deceived by the simplicity of the current attacks; for the attackers this simplicity arises more from convenience than necessity. As defence mechanisms are deployed to counter simple attacks, we are likely to see more complex attack scenarios. This article reviews current DDoS defence solutions in deployment and research. With these innovative ideas the article provides a fundamental understanding for developing new solutions.

7. REFERENCES

[1] Ferguson, P. and Senie, D. 2000. Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing.

[2] Park, K. and Lee, H. 2001b. On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law Internets. In Proceedings of the 2001 ACM SIGCOMM Conference. San Diego, California, USA, 15–26.

[3] Li, J., Mirkovic, J., Wang, M., Reither, P., and Zhang, L. 2002. Save: Source address validity enforcement protocol. In Proceedings of IEEE INFOCOM 2002. 1557–1566.

[4] Gil, T. M. and Poletto, M. 2001. Multops: a data-structure for bandwidth attack detection. In Proceedings of the 10th USENIX Security Symposium.

[5] Wang, H., Zhang, D., and Shin, K. G. 2002. Detecting SYN flooding attacks. In Proceedings of IEEE INFOCOM 2002. 1530–1539.

[6] Blazek, R. B., Kim, H., Rozovskii, B., and Tartakovsky, A. 2001. A novel approach to detection of "denial-of-service" attacks via adaptive sequential and batch-sequential change-point detection methods.

[7] C. Cheng , H. T. Kung , Koan-sin Tan. Use of spectral analysis in defense against DoS attacks. In Proceedings of IEEE GLOBECOM 2002. 2143–2148.

[8] Cabrera, J. B. D., Lewis, L., Qin, X., Lee, W., Prasanth, R. K. Proactive detection of distributed denial of service attacks using MIB traffic variables - a feasibility study. In Proceedings of the 7th IFIP/IEEE International Symposium on Integrated Network Management. Seattle, WA, 609–622.

[9] Peng, T., Leckie, C., and Kotagiri, R. 2004. Proactively detecting distributed denial of service attacks using source ip address monitoring. In Proceedings of the Third International IFIP-TC6 Networking Conference (Networking 2004).

[10] Brodsky, B. E. and Darkhovsky, B. S. 1993. Nonparametric Methods in Change-point Prob- lems. Kluwer Academic Publishers.

[11] Kompella, R. R., Singh, S., and Varghese, G. 2004. On scalable attack detection in the network. In IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement. ACM Press, New York, NY, USA, 187–200.

[12] Cheng, G. 2006. Malware FAQ: Analysis on DDOS tool Stacheldrahtv1.666,http://www.sans.org/resources/malwarefaq/stacheldraht.php

[13] Denning, D. E. 1987. An intrusion-detection model. IEEE Trans. Softw. Eng. 13, 2, 222–232.

[14] Gemberling, B., Morrow, C., and Greene, B. 2001. ISP security-real world techniques. Pre- sentation, NANOG. C 2827, the Internet Engineering Task Force (IETF).

[15] Burch, H. and Cheswick, B. 2000. Tracing anonymous packets to their approximate source. In Proceedings of the 14th Systems Administration Conference. New Orleans, Louisiana, USA.

[16] Savage, S., Wetherall, D., Karlin, A., and Anderson, T. 2000. Practical network support for IP traceback. In Proceedings of the 2000 ACM SIGCOMM Conference.

[17] Song, D. X. and Perrig, A. 2001. Advanced and authenticated marking schemes for IP trace- back. In Proceedings of IEEE INFOCOM 2001. 878–886.

[18] Wu, S. F., Zhang, L., Massey, D., and Mankin, A. 2001. Intension-Driven ICMP Trace-Back. IETF Internet Draft.

[19] Snoeren, A. C., Partridge, C., Sanchez, L. A., Jones, C. E., Tchakountio, F., Kent, S. T. and Strayer, W. T. 2001. Hash-based IP traceback. In Proceedings of the 2001 ACM SIG-COMM Conference. San Diego, California, USA.

[20] Esraa Alomari, Selvakumar Manickam, " Botnet-based Distributed Denial of Service Attacks on Web Servers:Classification and Art", International Journal of Computer Applications (0975 – 8887) Volume 49– No.7, July 2012.