“Distributed Denial of Service Attacks and their Remedies”

A

Seminar Report

Submitted in partial fulfillment for the award of the Degree of

Bachelor of Technology

Submitted by:

Mohit Jain

(CS-A)

Department of Computer Science & Engineering

Institute of Engineering & Technology, Alwar

December, 2010

Preface

This report discusses denial of service attacks, distributed denial of service attacks, detection of distributed denial of service attacks, and mitigation of distributed denial of service attacks. Suggested action is the encouragement of overall internet security, implementation of a detection mechanism and firewall, a rate limiting and resource multiplication policy, and an agreement with upstream networks concerning malicious traffic.

This document is a seminar report for the fulfillment of Seminar requirements for the award of the degree of Bachelor of Technology. This document describes the study conducted for the seminar presentation on “Distributed denial of service attacks and their remedies”.

Acknowledgment

I wish to express deep sense of gratitude to Dr. V. K. Agarwal (Chairman, IET Group of Institutions), Mr. S. P. Garg (Director, IET Group of Institutions), Dr. G. K. Joshi (Principal, Institute of Engineering & Technology – Alwar) for their support and resources.

I am greatly thankful to Mr. Mohit Khandelwal, Asst. Prof., Department of Computer Science & Engineering, who inspired and guided us in seminar work on "Distributed Denial of Service Attack and their remedies".

He helped and encouraged us in every possible way. The knowledge acquired during the work will defiantly help us in our future ventures.

I am also thankful towards Mr Anoop Vashishta (HOD – CS) for all the support and excellent guidance.

I also thank all the teachers of the department and my fellow students for their help in various aspects during the work.

Mohit Jain (CS - A)

Table of Contents

Table of Figures

1. Introduction

1.1 Overview

Denial of service attacks have become a growing problem over the last few years resulting in large losses for the victims [2]. One good example of this loss is the attacks of Yahoo, CNN, and Amazon in February of 2000 which had an estimated loss of several million to over a billion dollars [8]. This report will go over the fundamentals of denial of service attacks, how they can be detected, and some of the most common ways of mitigating the damage they can inflict upon their victims.

Distributed Denial of Service (DDoS) attacks are a virulent, relatively new type of attack on the availability of Internet services and resources. DDoS attackers infiltrate large numbers of computers by exploiting software vulnerabilities, to set up DDoS attack networks. These unwitting computers are then invoked to wage a coordinated, large-scale attack against one or more victim systems. As specific countermeasures are developed, attackers enhance existing DDoS attack tools, developing new and derivative DDoS techniques and attack tools. Rather than react to new attacks with specific countermeasures, it would be desirable to develop comprehensive DDoS solutions that defend against known and future DDoS attack variants. However, this requires a comprehensive understanding of the scope and techniques used in different DDoS attacks.

1.2 Denial of Service Attacks

Denial of service attacks come in an almost endless variety of forms but have the core similarity of their purpose. This purpose is to deny legitimate use of the services provided by their victim [1]. This is achieved by exhausting the systems resources such as bandwidth, and memory [8]. Unfortunately due to the limited nature of resources on the internet and the end to end focus of the networks design this is fairly easily achieved [1].

There are several different main kinds of methods that attackers use. The most straight forward method is sending a stream of packets to the victim to use all of the systems resources which is known as flooding [1]. Another common method is to send a smaller number of altered packets to confuse the protocol or application [1].

The most prevalent form of denial of service attack is the TCP/SYN Flooding method which makes up 90% of all denial of service attacks [8]. This attack takes advantage of the three way handshake procedure that the TCP protocol uses [2]. Normally the procedure goes something like the

following. The client sends a SYN message to let the server know the client wants to connect. Then the server sends a SYN/ACK message back letting the client know that it received the client's SYN message and is reserving resources for it. Finally the client sends the server an ACK message to complete the connection [2]. In a TCP/SYN flooding attack the misbehaving client or clients sends a flood of SYN messages to the server with spoofed IP's (fake IP info) but never respond to the SYN/ACK message the server responds with (to the spoofed IP's). This results in the server holding half open connections and reserving resources for each fraudulent SYN message eventually consuming them all [2]. Now that the basic nature of a denial of service extent has been explained we will go into distributed denial of service attacks.

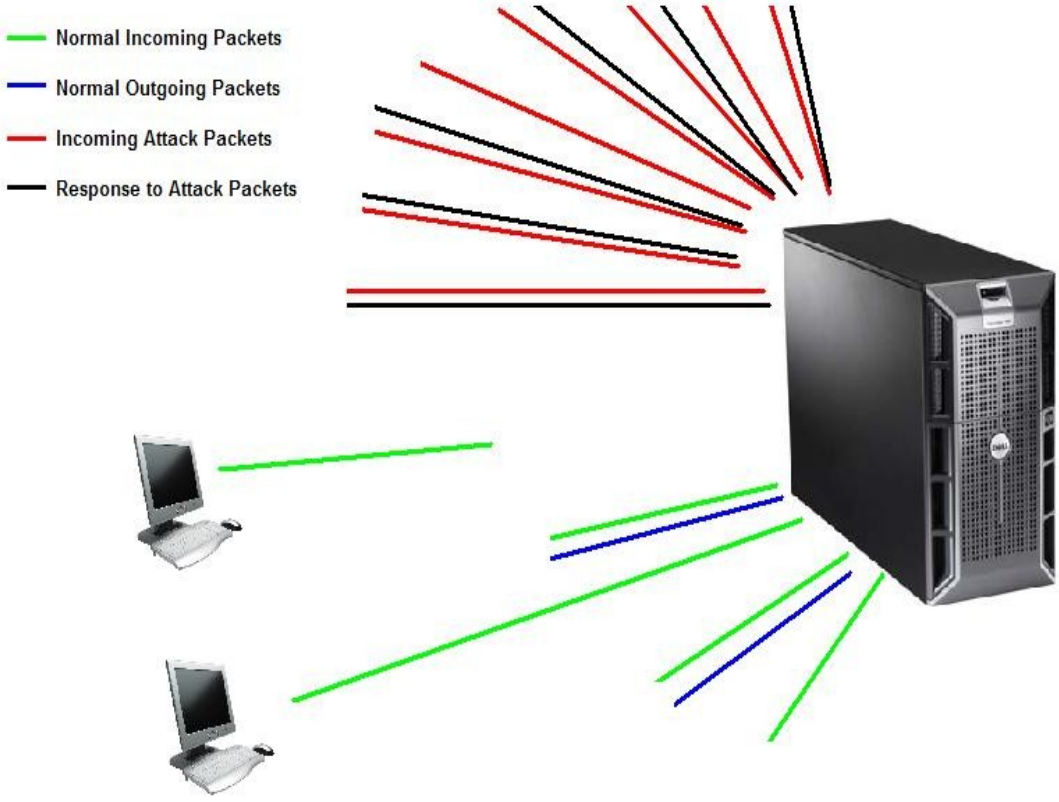

Figure 1: Denial of Service Attack

The red lines in the figure 1 indicates Incoming attack packets while black lines represent corresponding outgoing packets. Due to the network conjunction by this attack traffic some user's requests might never reach the server and some might never get the response. This illustrates denial of service to these users.

1.3 Distributed Denial of Service Attacks

Distributed denial of service attacks are basically denial of service attacks perpetrated by many systems at the same time on a single victim [1]. Such an attack occurs in two phases, the recruiting stage where the attacker recruits machines infecting them with an attack code and the actual attack phase when the recruited machines run the attack code [1]. See figure 1 from the Techguide.com Publication, "Stopping Attacks: The Importance of Denial of Service (DoS) Security Appliances," for a visual of how the attack works [7]. Some tools used by attackers in the past have included Trinoo (Trojan horse first discovered on December 30th 1999) [5], Tribe Flood Network (capable of UDP, ICMP, SYN Flood attacks as well as Smurf attacks) [3] and stacheldraht (based on Tribe Flood Network's Code) [4].

Distributed denial of service attacks can be deeply analyzed and broken into a variety of components such as degree of automation, exploited weakness, validity of the source address, attack rate dynamics, and impact on victim [1]. With all of these different characteristics, and the virtual arms race that is going on due to attackers altering their tools in response to security advances there is a need to be able to detect attacks so that action can be taken to mitigate the damage [1].

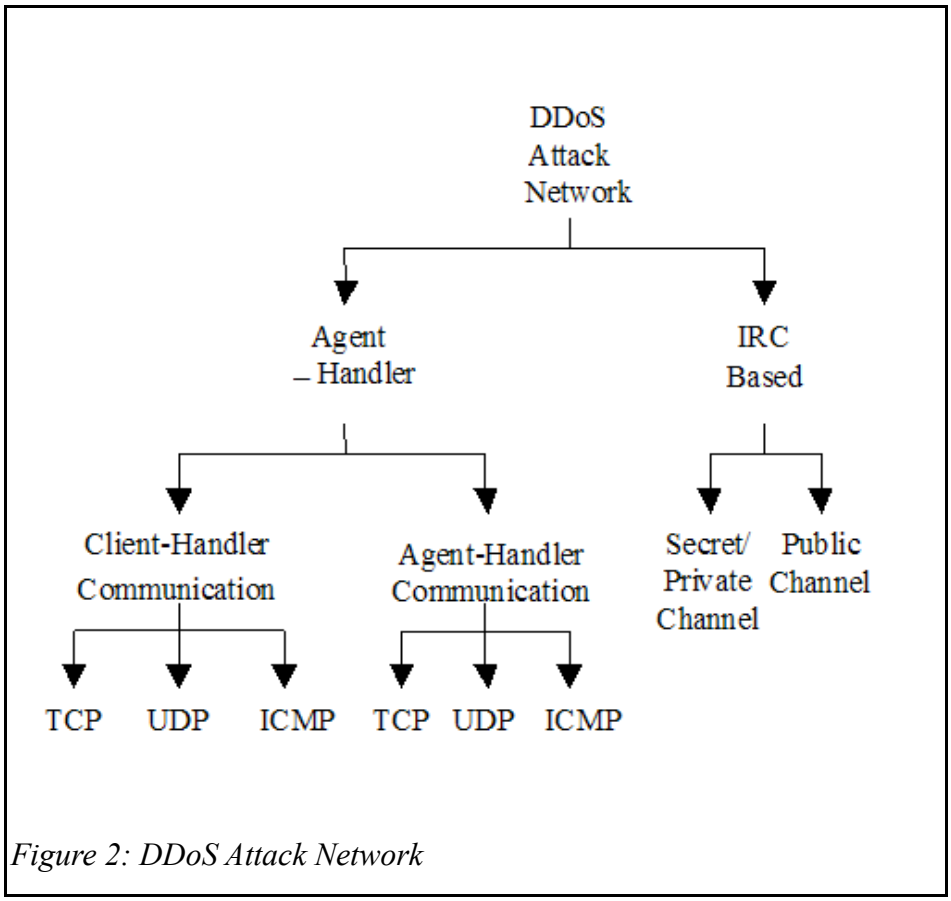

Figure 2: DDoS Attack Network

Figure 2 shows two main types of DDoS attack networks: the Agent-Handler model and the Internet Relay Chat (IRC-Based) model (See Figure 2).

1.3.1 Agent-Handler Model

An Agent-Handler DDoS attack network consists of clients, handlers, and agents (see Figure 3). The client platform is where the attacker communicates with the rest of the DDoS attack network. The handlers are software packages located on computing systems throughout the Internet that the attacker uses to communicate indirectly with the agents. The agent software exists in compromised systems that will eventually carry out the attack on the victim system. The attacker communicates with any number of handlers to identify which agents are up and running, when to schedule attacks, or when to upgrade agents. Depending on how the attacker configures the DDoS attack network, agents can be instructed to communicate with a single handler or multiple handlers. Usually, attackers will try and place the handler software on a compromised router or network server that handles large volumes of traffic. This makes it harder to identify messages between the client and handler and between the handler and agents. The communication between attacker and handler and between the handler and agents can be via TCP, UDP, or ICMP protocols. The owners and users of the agent systems typically have no knowledge that their system has been compromised and will be taking part in a DDoS attack.

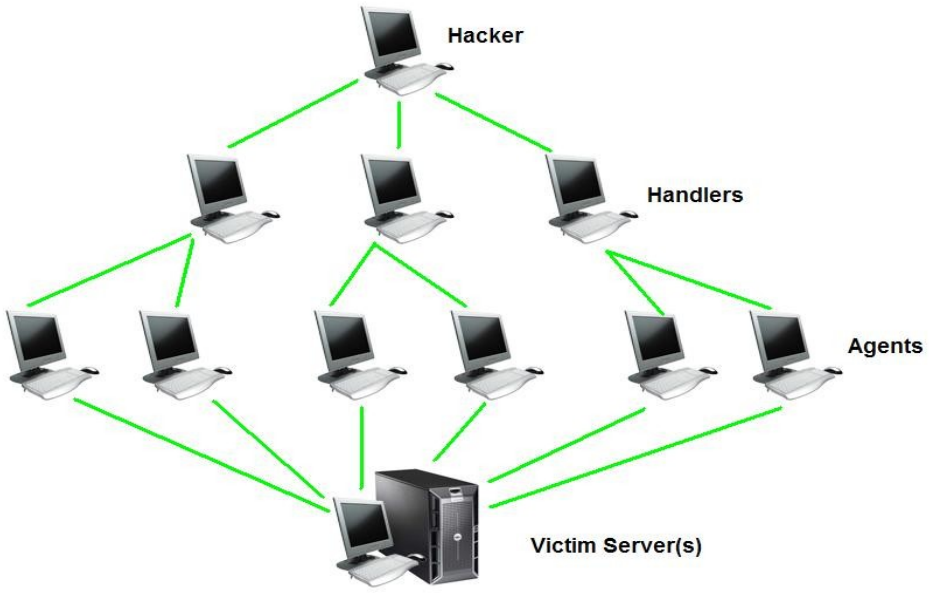

Figure 3: Agent-Handler Model

In descriptions of DDoS tools, the terms handler and agents are sometimes replaced with master and daemons respectively. Also, the systems that have been violated to run the agent software are referred to as the secondary victims, while the target of the DDoS attack is called the (primary) victim.

1.3.2 IRC Based Attack Model

Internet Relay Chat (IRC) is a multi-user, on-line chatting system. It allows computer users to create two-party or multi-party interconnections and type messages in real time to each other [9]. IRC network architectures consist of IRC servers that are located throughout the Internet with channels to communicate with each other across the Internet. IRC chat networks allow their users to create public, private and secret channels. Public channels are channels where multiple users can chat and share messages and files. Public channels allow users of the channel to see all the IRC names and messages of users in the channel [10]. Private and secret channels are set up by users to communicate with only other designated users. Both private and secret channels protect the names and messages of users that are logged on from users who do not have access to the channel [11]. Although the content of private channels is hidden, certain channel locator commands will allow users not on the channel to identify its existence, whereas secret channels are much harder to locate unless the user is a member of the channel.

An IRC-Based DDoS attack network is similar to the Agent-Handler DDoS attack model except that instead of using a handler program installed on a network server, an IRC communication channel is used to connect the client to the agents. By making use of an IRC channel, attackers using this type of DDoS attack architecture have additional benefits. For example, attackers can use "legitimate" IRC ports for sending commands to the agents [12]. This makes tracking the DDoS command packets much more difficult. Additionally, IRC servers tend to have large volumes of traffic making it easier for the attacker to hide his presence from a network administrator. A third advantage is that the attacker no longer needs to maintain a list of agents, since he can simply log on to the IRC server and see a list of all available agents [12]. The agent software installed in the IRC network usually communicates to the IRC channel and notifies the attacker when the agent is up and running. A fourth advantage is that IRC networks also provide the benefit of easy file sharing. File sharing is one of the passive methods of agent code distribution that we discuss in Section 4. This makes it easier for attackers to secure secondary victims to participate in their attacks.

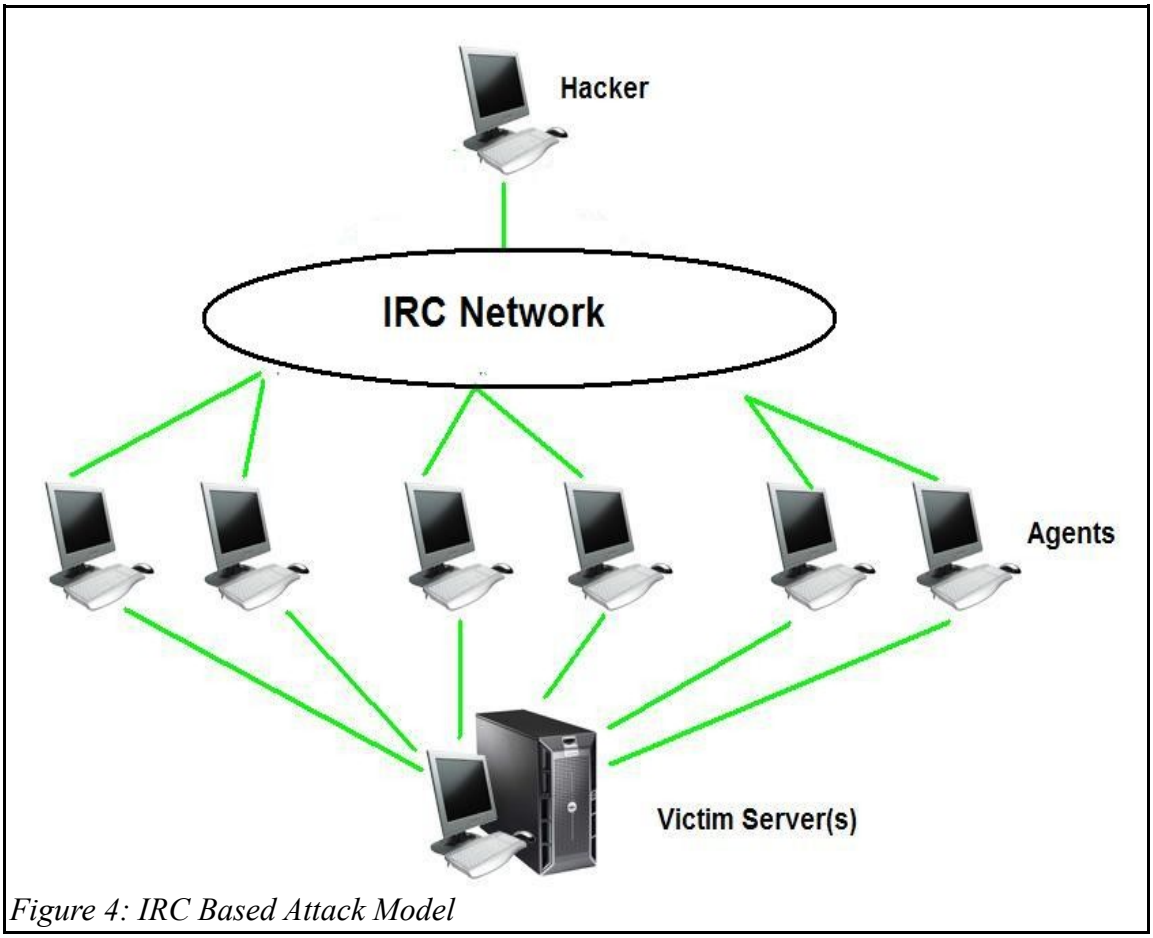

Figure 4: IRC Based Attack Model

In an IRC-based DDoS attack architecture, the agents are often referred to as "Zombie Bots" or "Bots". In both IRC-based and Agent-Handler DDoS attack models, we will refer to the agents as "secondary victims" or "zombies."

2. Case Study: Mininova

During March, 2009, the BitTorrent site Mininova was hit by a large-scale DDoS attack that caused a total of 14 hours of downtime [13, 14]. Regardless of what one think about torrent sites, this was an interesting example of how a website can be incapacitated by a DDoS attack.

Mininova shared some relevant information about the attack, especially a very telling traffic graph from their Internet connection. The below traffic graph shows the impact on one of Mininova's two Internet connections during the initial attack.

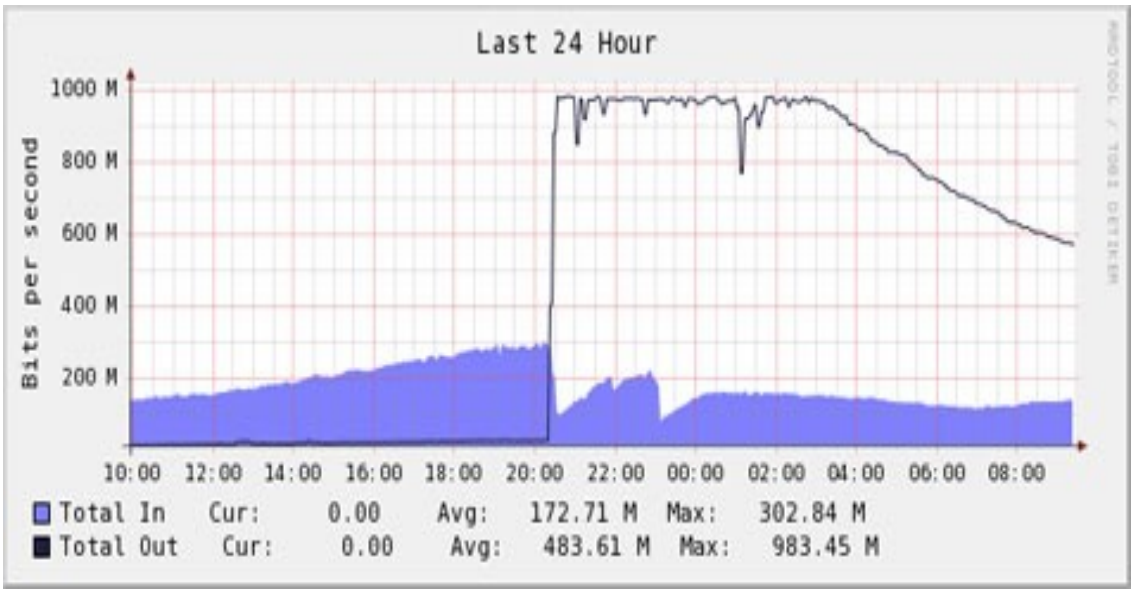

Figure 5: Mininova: Network Load

DDoS attacks are not an unusual event for BitTorrent sites, with smaller sites suffering the effects more often than they'd like. However, to take out one of the big players requires some serious power, and that is exactly what Mininova faced.

Mininova co-founder Niek confirmed around 10th march that "they have been suffering from a DDoS attack over the past few days. The site is currently being pounded by a botnet of hundreds of computers which is slowing the site down significantly and at times making it completely inaccessible." Niek said that "he has no idea who's behind the attack or why they chose to target Mininova. This is not the first time the site has had to deal with a Denial of Service attack, but they haven't witnessed one of this magnitude before."

It originating from three different continents, but seemed to wear off in the hours that followed. After some it was back in full force. Mininova is used to serving millions of visitors a day, but even they were not equipped to handle an attack like this. The attack originated from Germany and Argentina and was 2 Gbit strong. The DDoS attack maxed out the entire uplink and was hard to filter since it used UDP connections.

The site was attacked by a botnet (using hundreds of computers) using UDP connections, and judging by figure 5 it reached full effect almost immediately.

The attack generated 2 Gbit of traffic per second. Since the attack maxed out Mininova's Internet connection it made the site very slow and sometimes impossible to reach.

This is a typical example of a DDoS attack. Its objective was to in one way or another overload a site or service until it can't function properly.

2.1 Effect on Load-time and Up-time

The above network graph in figure 5 is interesting, but what was the actual effect on the website's load-time, and how much down-time did it result in? Figure 7 shows Up-time monitoring data for the site (from Pingdom.com) which clearly shows the effect of the DDoS attack.

As can be seen by the load time graph in Figure 6, there were actually two separate attacks in two days. The time stamps below are in GMT+1.

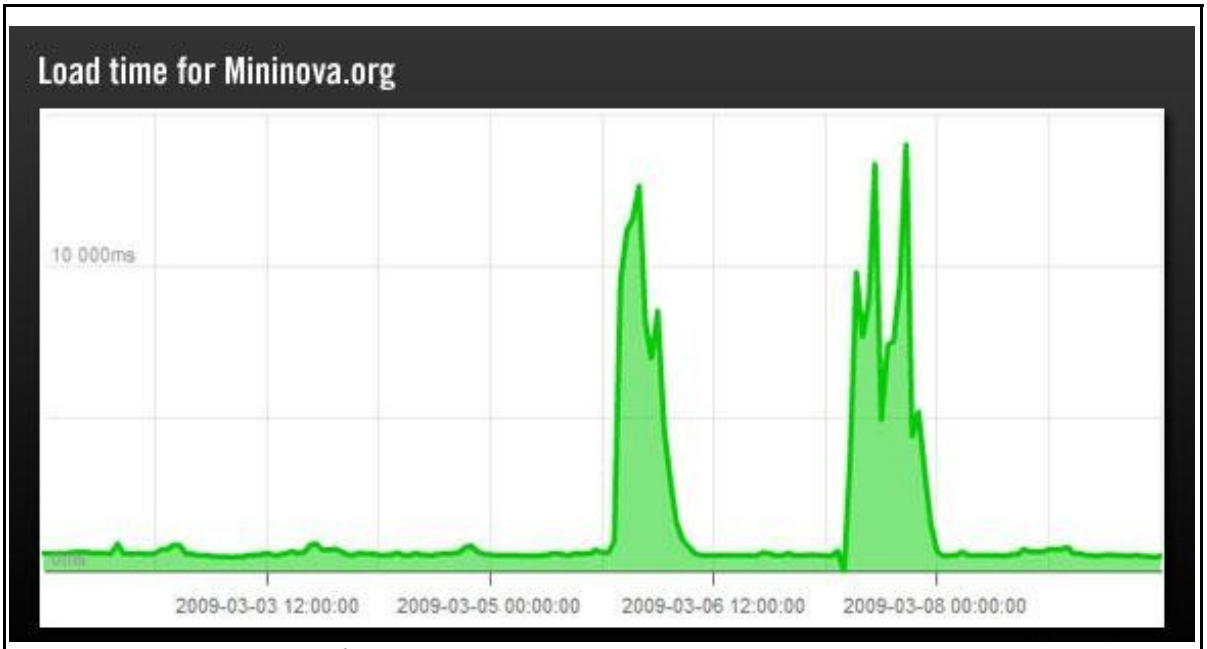

Figure 6: Mininova: Load time

Note that the load time in the graph above only includes the loading of the HTML, not images, etc.

The Figure 6 only shows the load time for when the website could be loaded at all. In many cases the load attempt simply timed out. So the effect was double. Slowdown AND downtime. Note how the reduced uptime in the graph of Figure 7 matches the periods of increased load time.

Counted over the two attacks, this DDoS attack cost Mininova 14 hours of downtime and some extreme slowdown. Remember that people tend to leave a website if it is too slow, so even when the website wasn't technically down many visitors would still have been turned away.

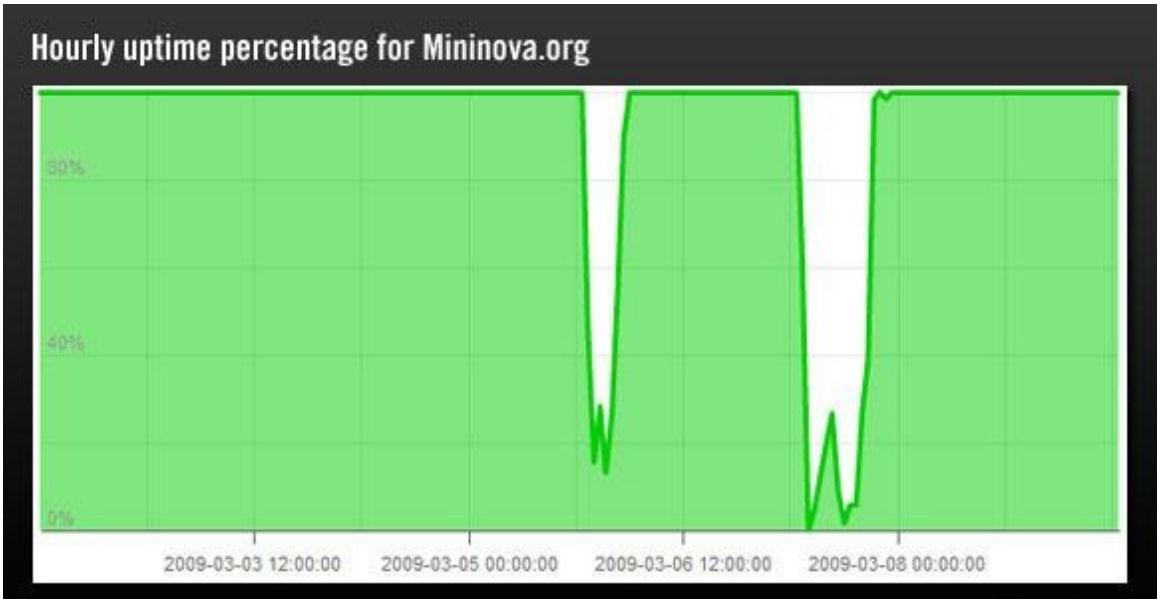

Figure 7: Mininova: Up-time

This practical example gives a decent picture of how devastating a DDoS attack can be to a website.

It's worth pointing out that what is described in this case study can happen to any type of site. A similar attack could have happened to a blog, an e-commerce site, a social network, a web host, etc.

Another thing to note is that there are a very wide range of different attacks that can happen. To name a different example than the one above (Explained in section 4), the domain registrar Network Solutions suffered from a large-scale attack on their DNS servers that indirectly affected hundreds of thousands of websites that used those DNS servers.

3. Case Study: Wordpress

Wordpress.com is one of the largest website on the internet with over 5 million users. On Oct, 27, 2008 Wordpress.com faced a large Distributed Denial of Service Attack [15].

Wordpress posted time line and description of the event (sub section 3.1) [15].

3.1 Wordpress: Event Time Line (Oct, 27, 2008)

*** The following text in this subsection is as posted by wordpress.com

9:40 AM EST — Our internal monitoring systems alerted us to unusual activity in one of the four geographically diverse datacenters which serve WordPress.com traffic. Here is what that anomaly looks like in graphical terms:

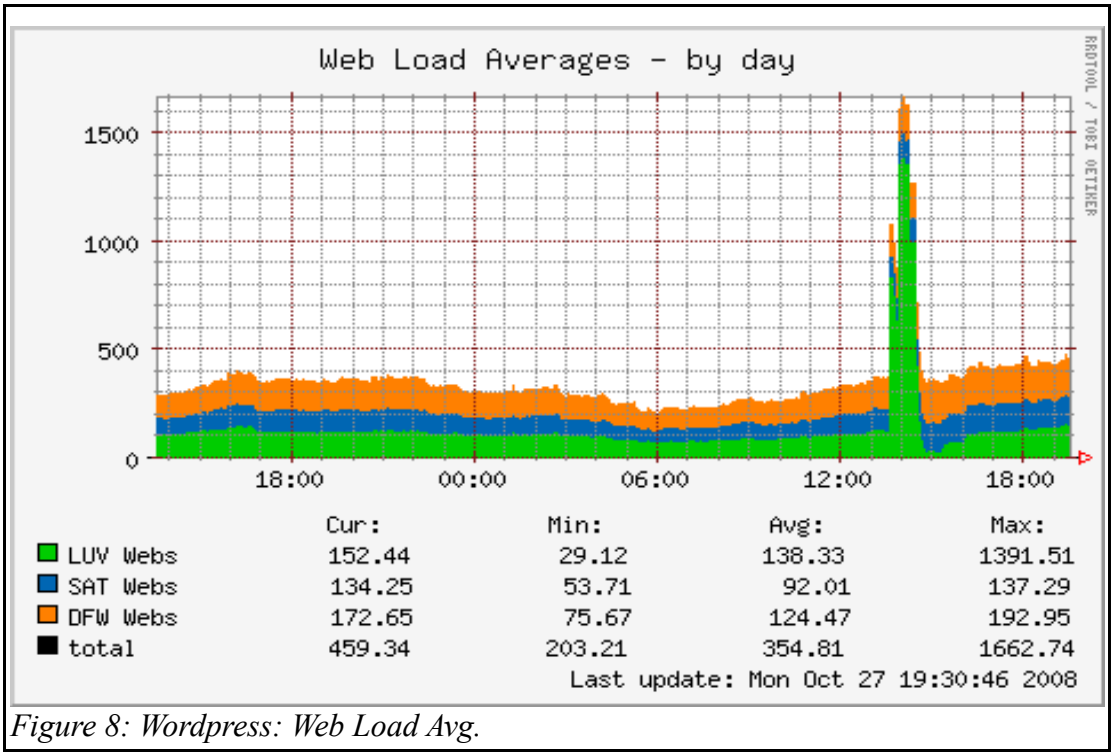

Figure 8: Wordpress: Web Load Avg.

10:00 AM EST — The target of the attack was identified and removed from our network. The attack, however continued. This is because the attacker had hijacked tens of thousands of computers (probably by installing a virus which was spread via email) and these computers had no idea the site was no longer there. A small log sample shows over 8 million requests for this one site from over 10,000 unique IP addresses.

10:20 AM EST — Since we have servers in multiple data centers throughout the United States which serve traffic for WordPress.com all the time, we were able to route all legitimate traffic out of the affected data center, and let the single affected data center deal with the attack.

11:30 AM EST — The IPs targeted in the attack were null routed at this point which allowed us to bring all datacenters back online to serve normal traffic.

We keep hourly traffic metrics and based on those numbers, it looks like during the attack there was about a 5% decrease in overall pageviews during the 40 minutes before traffic was re-routed. All things considered, not a bad outcome for an attack this size. Looking at bandwidth graphs, this attack was in the 500Mbit – 750Mbit/sec range.

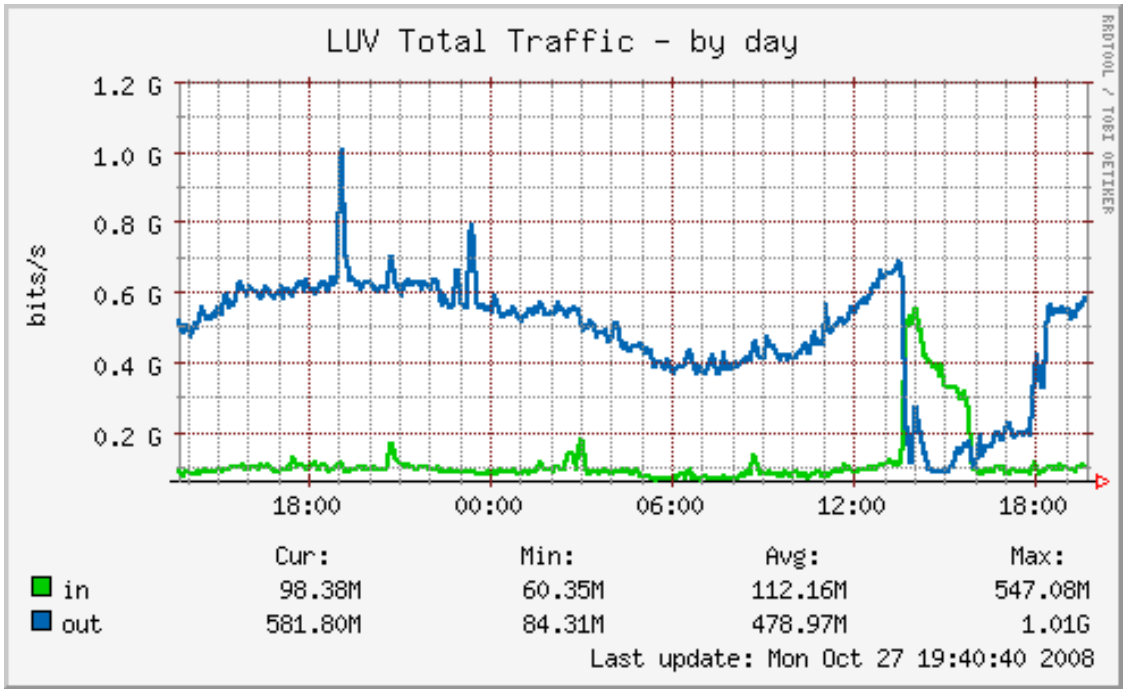

Figure 9: Wordpress: Traffic Data

4. DDoS Attack on Root Name Servers

4.1 Root Name Server

A root name server is a name server for the Domain Name System's root zone. It directly answers requests for records in the root zone and answers other requests returning a list of the designated authoritative name servers for the appropriate top-level domain (TLD). The root name servers are a critical part of the Internet because they are the first step in translating (resolving) human readable host names into IP addresses that are used in communication between Internet hosts.

The Domain Name System is a hierarchical naming system for computers, services, or any resource participating in the Internet. The top of that hierarchy is the root domain. The root domain does not have a formal name and its label in the DNS hierarchy is an empty string. All fully qualified domain

names (FQDNs) on the Internet can be regarded as ending with this empty string for the root domain, and therefore ending in a full stop character (the label delimiter), e.g., www.example.com.. This is generally implied rather than explicit, as modern DNS software does not actually require that the terminating dot be included when attempting to translate a domain name to an IP address.

The root domain contains all top-level domains of the Internet. As of June 2009, there are 20 generic top-level domains (gTLDs) and 248 country code top-level domains (ccTLDs) in the root domain. In addition, the ARPA domain is used for technical name spaces in the management of Internet addressing and other resources. A TEST domain is used for testing internationalized domain names.

The choice of 13 nameservers was made because of limitations in the original DNS specification, which specifies a maximum packet size of 512 bytes when using the User Datagram Protocol (UDP).[7] The addition of IPv6 addresses for the root nameservers requires more than 512 bytes, which is facilitated by the EDNS0 extension to the DNS standard.[8] While only 13 names are used for the root nameservers, there are many more physical servers; C, F, I, J, K, L and M servers now exist in multiple locations on different continents, using anycast address announcements to provide decentralized service. As a result most of the physical root servers are now outside the United States, allowing for high performance worldwide.

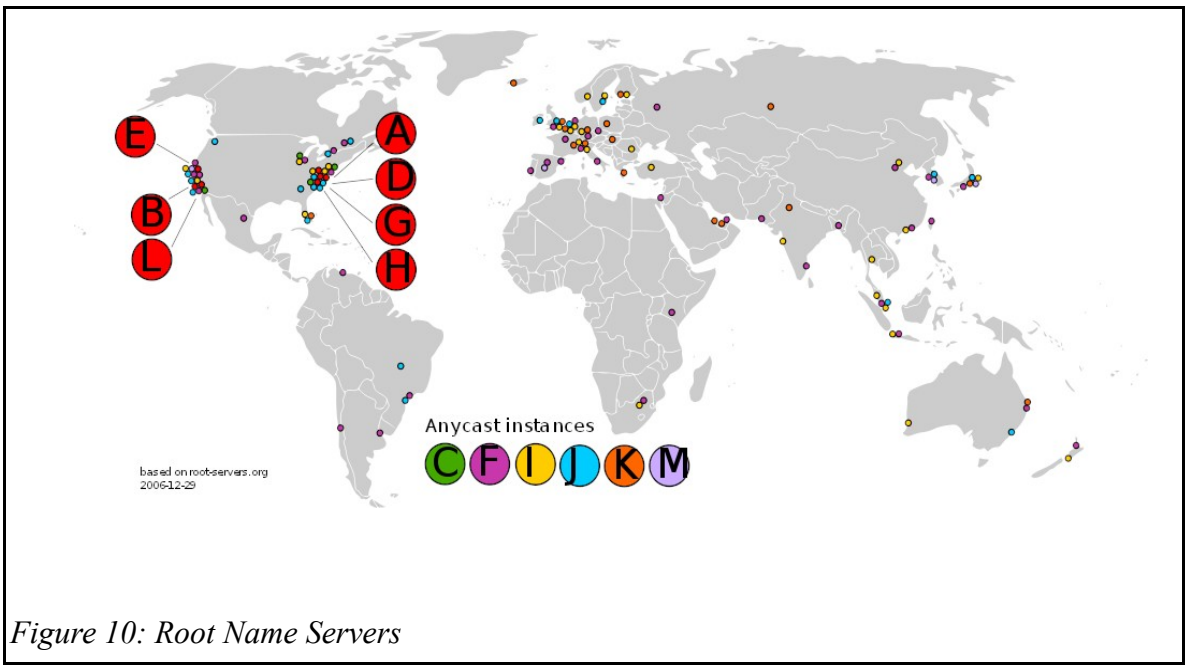

Figure 10: Root Name Servers

4.2 Attacks

Distributed denial of service attacks on root nameservers are Internet events in which distributed denial-of-service attacks target one or more of the thirteen Domain Name System root nameservers. The root nameservers are critical infrastructure components of the Internet, mapping domain names to Internet Protocol (IP) addresses and other information. Attacks against the root nameservers can impact operation of the entire Internet, rather than specific websites

4.2.1 October 21, 2002

On October 21, 2002 an attack lasting for approximately one hour was targeted at all 13 DNS root name servers.[16]

This event was the first significant attack directed at disabling the Internet itself instead of specific websites.[citation needed] This was the second significant failure of the root nameservers. The first caused the failure of seven machines in April 1997 due to a technical problem.[17]

4.2.2 February 6, 2007

On February 6, 2007 an attack began at 1000 UTC and lasted twenty-four hours. At least two of the root servers (G-ROOT and L-ROOT) reportedly suffered badly while two others (F-ROOT and M-ROOT) experienced heavy traffic [16]. The latter largely contained the damage by distributing requests to other root server instances with anycast addressing. ICANN published a formal analysis shortly after the event.[18]

Due to a lack of detail, speculation about the incident proliferated in the press until details were released. On February 8, 2007 it was announced by Network World that: "If the United States found itself under a major cyber attack aimed at undermining the nation's critical information infrastructure, the Department of Defense is prepared, based on the authority of the President, to launch an actual bombing of an attack source or a cyber counterattack."[19]

5. Defense Mechanisms

The seriousness of the DDoS problem and the increased frequency of DDoS attacks have led to the advent of numerous DDoS defense mechanisms. Some of these mechanisms address a

specific kind of DDoS attack such as attacks on Web servers or authentication servers. Other approaches attempt to solve the entire generic DDoS problem. Most of the proposed approaches require certain features to achieve their peak performance, and will perform quite differently if deployed in an environment where these requirements are not met. As is frequently pointed out, there is no "silver bullet" against DDoS attacks. Therefore we need to understand not only each existing DDoS defense approach, but also how those approaches might be combined together to effectively and completely solve the problem.

Based on the activity level of DDoS defense mechanisms, we differentiate between preventive and reactive mechanisms.

5.1 Preventive

The goal of preventive mechanisms is either to eliminate the possibility of DDoS attacks altogether or to enable potential victims to endure the attack without denying services to legitimate clients. According to these goals we further divide preventive mechanisms into attack prevention and denial-of-service prevention mechanisms.

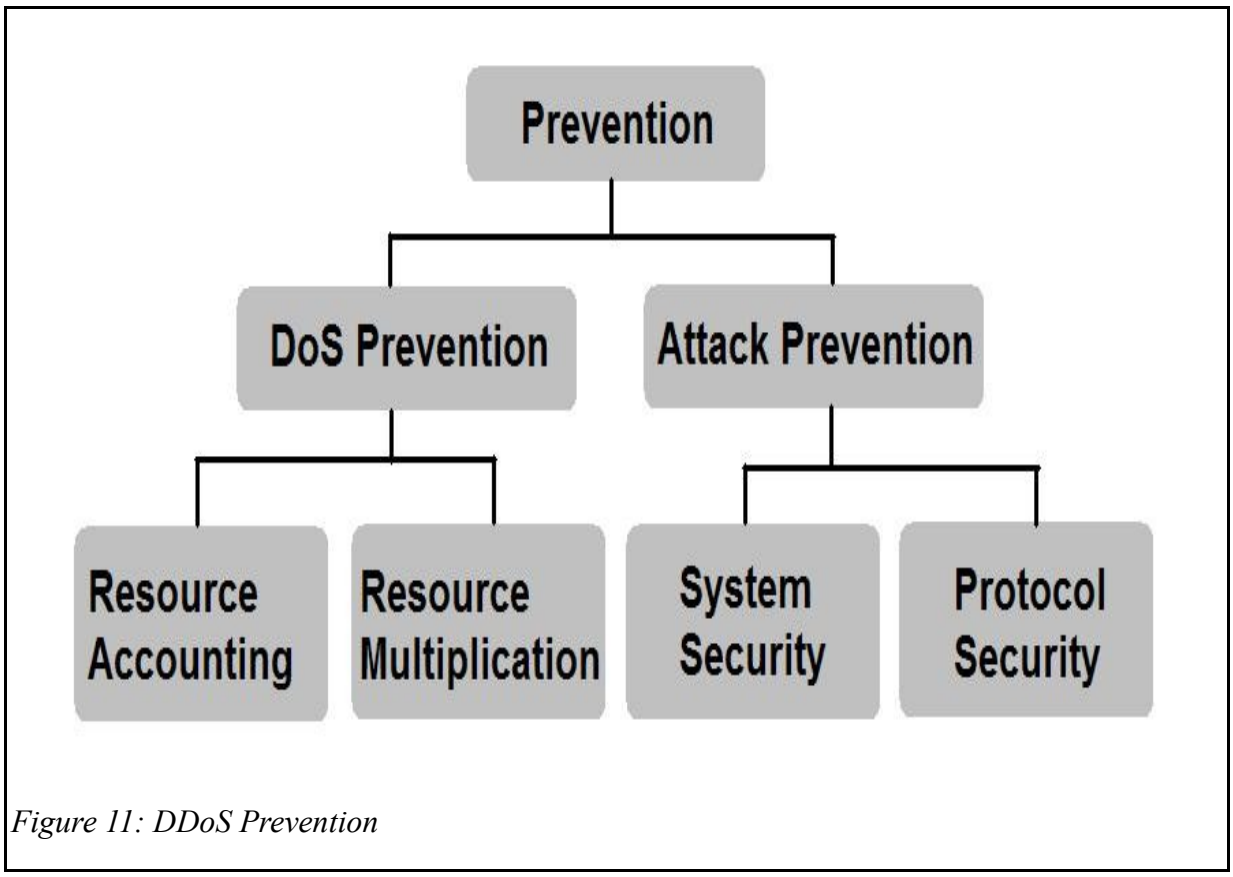

Figure 11: DDoS Prevention

5.1.1 Attack Prevention

Attack prevention mechanisms modify the system configuration to eliminate the possibility of a DDoS attack. Based on the target they secure, we further divide them into system security and protocol security mechanisms.

5.1.1.1 System Security

System security mechanisms increase the overall security of the system, guarding against illegitimate accesses to the machine, removing application bugs and updating protocol installations to prevent intrusions and misuse of the system. DDoS attacks owe their power to large numbers of subverted machines that cooperatively generate the attack streams. If these machines were secured, the attackers would lose their army and the DDoS threat would then disappear. On the other hand, systems vulnerable to intrusions can themselves become victims of DDoS attacks in which the attacker, having gained unlimited access to the machine, deletes or alters its contents. Potential victims of DDoS attacks can be easily overwhelmed if they deploy vulnerable protocols. Examples of system security mechanisms include monitored access to the machine, applications that download and install security patches, firewall systems, virus scanners, intrusion detection systems, access lists for critical resources, capability-based systems and client-legitimacy-based systems. The history of computer security suggests that this approach can never be 100% effective, but doing a good job here will certainly decrease the frequency and strength of DDoS attacks

5.1.1.2 Protocol Security

Protocol security mechanisms address the problem of bad protocol design. Many protocols contain operations that are cheap for the client but expensive for the server. Such protocols can be misused to exhaust the resources of a server by initiating large numbers of simultaneous transactions. Classic misuse examples are the TCP SYN attack, the authentication server attack, and the fragmented packet attack, in which the attacker bombards the victim with malformed packet fragments forcing it to waste its resources on reassembling attempts. Examples of protocol security mechanisms include guidelines for a safe protocol design in which resources are committed to the client only after sufficient authentication is done, or the client has paid a sufficient price, deployment of powerful proxy server that completes TCP connections, etc. Deploying comprehensive protocol and system security mechanisms can make the victim

completely resilient to protocol attacks. Also, these approaches are inherently compatible with and complementary to all other approaches.

5.1.2 DoS Prevention

Denial-of-service prevention mechanisms enable the victim to endure attack attempts without denying service to legitimate clients. This is done either by enforcing policies for resource consumption or by ensuring that abundant resources exist so that legitimate clients will not be affected by the attack. Consequently, based on the prevention method, we differentiate between resource accounting and resource multiplication mechanisms.

5.1.2.1 Resource Accounting

Resource accounting mechanisms police the access of each user to resources based on the privileges of the user and his behavior. Such mechanisms guarantee fair service to legitimate well-behaving users. In order to avoid user identity theft, they are usually coupled with legitimacy-based access mechanisms that verify the user's identity.

5.1.2.2 Resource Multiplication

Resource multiplication mechanisms provide an abundance of resources to counter DDoS threats. The straightforward example is a system that deploys a pool of servers with a load balancer and installs high bandwidth links between itself and upstream routers. This approach essentially raises the bar on how many machines must participate in an attack to be effective. While not providing perfect protection, for those who can afford the costs, this approach has often proven sufficient. For example, Microsoft has used it to weather large DDoS attacks.

5.2 Reactive

Reactive mechanisms strive to alleviate the impact of an attack on the victim. In order to attain this goal they need to detect the attack and respond to it. The goal of attack detection is to detect every attempted DDoS attack as early as possible and to have a low degree of false positives. Upon attack detection, steps can be taken to characterize the packets belonging to the attack stream and provide this characterization to the response mechanism.

5.2.1 Detection Strategy

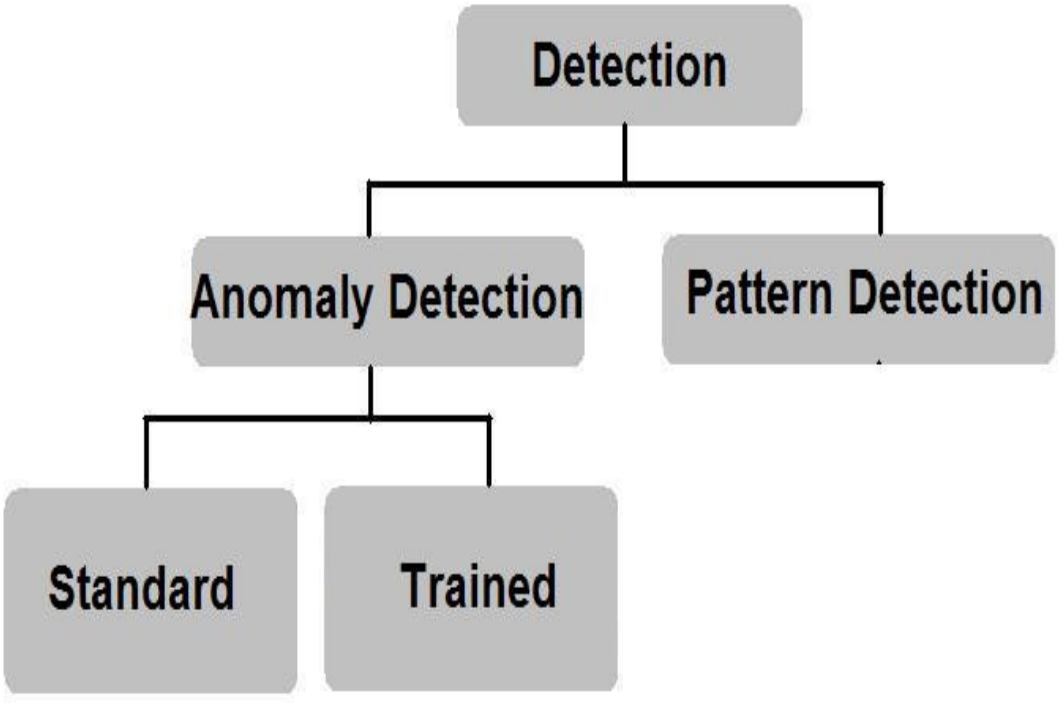

Figure 12: DDoS Detection

We classify reactive mechanisms based on the attack detection strategy into mechanisms that deploy pattern detection, anomaly detection, hybrid detection, and third-party detection.

5.2.1.1 Pattern Attack Detection

Mechanisms that deploy pattern detection store the signatures of known attacks in a database. Each communication is monitored and compared with database entries to discover occurrences of DDoS attacks. Occasionally, the database is updated with new attack signatures. The obvious drawback of this detection mechanism is that it can only detect known attacks, and it is usually helpless against new attacks or even slight variations of old attacks that cannot be matched to the stored signature. On the other hand, known attacks are easily and reliably detected, and no false positives are encountered.

5.2.1.2 Anomaly Attack Detection

Mechanisms that deploy anomaly detection have a model of normal system behavior, such as a model of normal traffic dynamics or expected system performance. The current state of the system is periodically compared with the models to detect anomalies. The advantage of anomaly detection over pattern detection is that unknown attacks can be discovered. However, anomaly based detection has to address two issues:

a) Threshold setting

Anomalies are detected when the current system state differs from the model by a certain threshold. The setting of a low threshold leads to many false positives, while a high threshold reduces the sensitivity of the detection mechanism.

b) Model update

Systems and communication patterns evolve with time, and models need to be updated to reflect this change. Anomaly based systems usually perform automatic model update using statistics gathered at a time when no attack was detected. This approach makes the detection mechanism vulnerable to increasing rate attacks that can wrongly train models and delay or even avoid attack detection.

5.2.2 Reaction Strategy

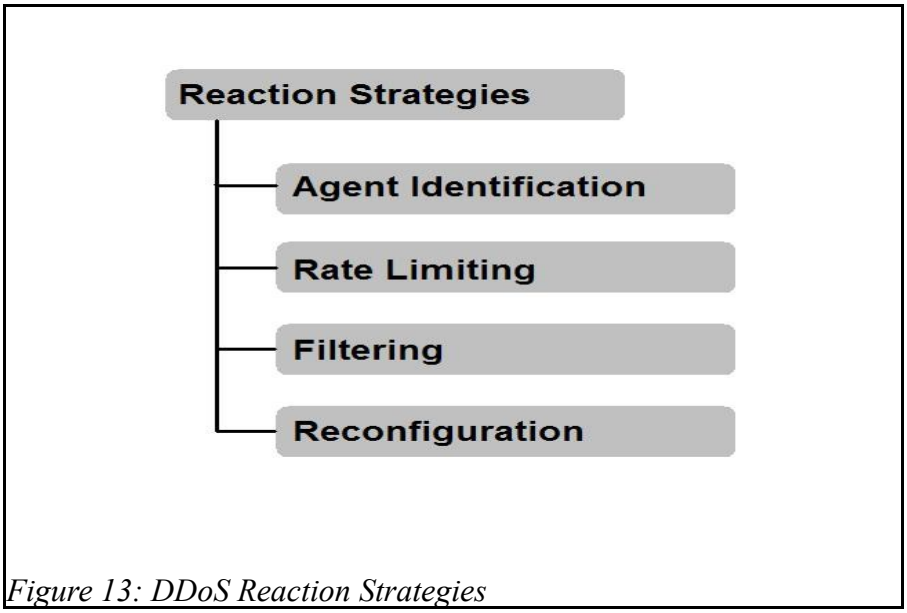

Figure 13: DDoS Reaction Strategies

The goal of the attack response is to relieve the impact of the attack on the victim, while imposing minimal collateral damage to legitimate clients of the victim. We classify reactive mechanisms based on the response strategy into mechanisms that deploy agent identification, rate-limiting, filtering and reconfiguration approaches.

5.2.2.1 Agent Identification

Agent identification mechanisms provide the victim with information about the identity of the machines that are performing the attack. This information can then be combined with other response approaches to alleviate the impact of the attack. Agent identification examples include numerous trace back techniques and approaches that eliminate spoofing, thus enabling use of the source address field for agent identification.

5.2.2.2 Rate-limiting

Rate-limiting mechanisms impose a rate limit on a stream that has been characterized as malicious by the detection mechanism. Rate limiting is a lenient response technique that is usually deployed when the detection mechanism has a high level of false positives or cannot precisely characterize the attack stream. The disadvantage is that they allow some attack traffic through, so extremely high scale attacks might still be effective even if all traffic streams are rate-limited.

5.2.2.3 Filtering

Filtering mechanisms use the characterization provided by a detection mechanism to filter out the attack stream completely. Examples include dynamically deployed firewalls, and also a commercial system Traffic Master. Unless detection strategy is very reliable, filtering mechanisms run the risk of accidentally denying service to legitimate traffic. Worse, clever attackers might leverage them as denial of service tools.

5.2.2.4 Reconfiguration

Reconfiguration mechanisms change the topology of the victim or the intermediate network

to either add more resources to the victim or to isolate the attack machines. Examples include reconfigurable overlay networks, resource replication services, attack isolation strategies, etc.

6. Related Work

Although distributed denial-of-service attacks have been recognized as a serious problem, we are not aware of much attempts to introduce formal classification into the DDoS attack mechanisms. The reason might lay in the use of fairly simple attack tools that have dominated most DDoS incidents. Those tools performed full-force flooding attacks using several types of packets. As defense mechanisms are deployed to counter these simple attacks, we expect to be faced with more complex strategies.

Few authors present classification of denial of service attacks according to the type of the target (firewall, Web server, router), a resource that the attack consumes (network bandwidth, TCP/IP stack) and the exploited vulnerability (bug or overload). This classification focuses more on the actual attack phase, while we are interested in looking at the complete attack mechanism in order to highlight features that are specific to distributed attacks.

CERT is currently undertaking the initiative to devise a comprehensive taxonomy of computer incidents as part of the design of common incident data format and exchange procedures, but unfortunately their results are not yet available. We are not aware of any attempt to formally classify DDoS defense systems, although similar works exist in field of intrusion detection systems and offer informative reading for researchers in the DDoS defense field.

7. Conclusion

Denial of service attacks are a huge threat to the internet as a whole. In order to thwart these attacks over all internet security must be promoted and potential targets must be prepared for the potential attacks. It is critical that security methods evolve with the evolving denial of service attacks to be truly secure. Formal Classification by some Community related organization is necessary in the field of Distributed Denial of Service.

Appendix A: Glossary

DoS – Denial of Service

DDoS – Distributed Denial of Service

IRC – Internet Relay Chat

TCP – Transmission Control Protocol

IP – Internet Protocol

SYN - Synchronize

ACK - Acknowledgment

UDP – User Datagram Protocol

ICMP – Internet Message Control Protocol

Appendix B: References

[1]. J. Mirkovic and P. Reiher, “A Taxonomy of DDoS Attack and DDoS Defense Mechanisms,” ACM SIGCOMM Computer Communications Review, Volume 34, Number 2, April 2004, pp. 39-53.

[2]. V. A. Siris and F. Papagalou, Application of Anomaly Detection Algorithms for Detecting SYN Flooding Attacks, Institute of Computer Science, Foundation for Research and Technology, Globecom, 2004.

[3]. D. Dittrich, Tribe Flood Network, http://staff.washington.edu/dittrich/talks/cert/tfn.html, Accessed 11/13/2007, 5:51pm.

[4]. D. Dittrich, The "stacheldraht" distributed denial of service attack tool, http://staff.washington.edu/dittrich/misc/stacheldraht.analysis, Accessed 11/13/2007, 5:51pm, Copyright 1999.

[5]. Symantic, W32.DoS.Trinoo, http://www.symantec.com/security_response/writeup.jsp?docid=2000-122009-3804-99&tabid=2, Accessed 11/13/2007, 5:51 pm.

[6]. M. Tanase, Closing the Floodgates:DDoS Mitigation Techniques, http://www.securityfocus.com/infocus/1655, Accessed 11/13/2007, Published: 1/07/2003.

[7]. Techguide.com, “Stopping Attacks: The Importance of Denial of Service (DoS) Security Appliances” http://www.opsec.com/solutions/partners/downloads/stopping_attacks.pdf, , Visted 11/13/2007, 5:51 pm.

[8]. V. A. Siris, “Denial of Service and Anomaly Detection” http://www.ics.forth.gr/netlab/presentations/dos_detection_zagreb.pdf, Accesed 11/13/2007, 5:51 pm.

[9]. Joseph Lo and Others. “An IRC Tutorial”, irchelp.com. 1997. http://www.irchelp.org/irchelp/irctutorial.html#part1. (8 April 2003).

[10]. Nicolas Pioch. “A Short IRC Primer”. Edition 1.2, January 1997. http://www.irchelp.org/irchelp/ircprimer.html#DDC. (21 April 2003).

[11]. Kleinpaste, Karl, Mauri Haikola, and Carlo Kid. “The Original IRC Manual”. March 18, 1997. http://www.user-com.undernet.org/documents/irc-manual.html#seen (21 April 2003).

[12]. Kevin J. Houle. “Trends in Denial of Service Attack Technology”. CERT Coordination Center, Carnegie Mellon Software Engineering Institute. October 2001. www.nanog.org/mtg-0110/ppt/houle.ppt. (14 March 2003).

[13]. http://torrentfreak.com/mininova-hit-by-massive-ddos-attack-090307/

[14]. http://royal.pingdom.com/2009/03/10/the-anatomy-of-a-ddos-attack/

[15]. http://barry.wordpress.com/category/wordpresscom/

[16]. http://en.wikipedia.org/wiki/Distributed_denial_of_service_attacks_on_root_nameservers

[17]. "Router glitch cuts Net access". CNET News.com. 1997-04-25. Retrieved 2008-07-11.

[18]. "Factsheet - Root server attack on 6 February 2007". ICANN. 2007-03-01. Retrieved 2008-07-11.

[19]. Messmer, Ellen (2007-02-08). "U.S. cyber counterattack: Bomb 'em one way or the other". Network World, Inc. Retrieved 2008-07-11.