CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The Internet owes much of its historic success and growth to its openness to new applications. New applications can be designed, implemented and come into widespread use much more quickly, if they do not need to wait for key features to be added to the underlying network. Perversely, this has happened as a rational response of network and system administrators needing to cope with the consequences of the Internet’s openness. The Internet architecture is vulnerable to Denial-of Service (DoS) attacks, where any collection of hosts with enough bandwidth can disrupt legitimate communication between any pair of other parties, simply by flooding one end or the other with unwanted traffic. These attacks are widespread, increasing, and have proven resistant to all attempts to stop them. Thus, in order to defend against these Distributed Denial-of-Service (DDoS) attacks that plague websites today; it is proposed to mitigate the DDoS attacks.

1.1.1 Distributed Denial of Service Attacks

Confidentiality, Integrity and Availability often referred as the CIA triad are the three main objectives of computer security. According to the code of Laws of the United States regarding the definition of information security, “availability means ensuring timely and reliable access to and use of
information. Denial of Service attacks makes the resources unavailable to the intended users making it a major potential threat to availability.

According to International Telecommunications Union’s (ITU-T) recommendation, Denial of Service is defined as the prevention of authorized access to resources or the delay of time-critical operations. The Committee on National Security Systems (CNSS) defines Denial of Service as: “Any action or series of actions that prevents any part of an information system from functioning.

DoS attack can be characterized as an attack with the purpose of preventing legitimate users from using a victim computing system or network resource. A Distributed Denial of Service (DDoS) attack is a large-scale, coordinated attack on the availability of services of a victim system or network resource, launched indirectly through many compromised computers on the Internet. The services under attack are those of the “primary victim”, while the compromised systems used to launch the attack are often called the “secondary victims. An Attacker controls the primary victims, which in turn control the secondary victims (Zombies). According to the United States Computer Emergency Response Team, the symptoms of denial-of-service attacks include unusual slow network performance, unavailability of a particular web site and the inability to access any web site.

On the Internet, a DDoS attack is one in which a multitude of compromised systems attack a single target, thereby causing denial of service for users of the targeted system. The flood of incoming messages to the target system essentially forces it to shut down, thereby denying service to the system to legitimate users. A hacker begins a DDoS attack by exploiting vulnerability in one computer system and making it the DDoS "master." It is from the master system that the intruder identifies and communicates with other systems that can be compromised. Figure 1.1 refers a typical DDoS
attack scenario. The masters control the zombies (compromised computers) and the attacker is in control of the masters. The below scenario is highly scalable as millions of zombies are in the attackers army.

![Diagram of a typical DDoS network](image)

**Figure 1.1 Typical DDoS network**

### 1.1.2 Threats of Distributed Denial of Service Attacks

Distributed Denial of Service attacks offers negative effects on the aspects of personal privacy, business activity and national security. What makes this dangerous attack possible are the enormous computing power on desktops today, the massive volume of electronic communications and the number of easily available tools that allow even a routine user to flood an organization's network. Some evidences reporting the seriousness of the DDoS attacks include:

- SANS DDoS report 2014 -- SANS announce results of its 2014 survey on DDoS, sponsored by Corero Network Security, in which 378 IT professionals answered questions about their experience with DDoS attacks and their ability to protect their assets.
- Enterprises experience 4.5 DDoS events per year that span a bandwidth of 1.7 GB, last 8.7 hours, and cause costly outages lasting 2.3 hours for enterprises.
- DDoS attacks are affecting every sector, they are no longer solely based on volume, they are also targeting applications and managing to deny service on those applications.
- Data Centers are increasingly coming under DDoS attack, according to a report by a security vendor Arbor networks (January 2014)
- According to “DDoS threat report 2013 released by NSFOCUS Information Technology Co., Ltd
  - 27.9 DDoS attacks occurred every hour
  - HTTP_FLOOD, DNS_FLOOD, TCP_FLOOD were the top three attack types contributing 87.32% of all attacks. Figure 1.2 graphically shows the major DDoS attack types occurred during 2013.

![Graph exhibiting major DDoS attack types during 2013](image)

**Figure 1.2** Graph exhibiting major DDoS attack types during 2013
○ Hacktivism remained the primary motive with 40.72% of all DDoS attacks targeting government entities and non-governmental organizations.

○ The longest single DDoS attack lasted 1,666 hours – 69 days and 10 hours.

● “More than one-fifth of UK firms hit by DDoS attacks in 2012 – a survey by communications and analysis firm Neustar has revealed.

● Key industries reported the highest levels of attack, with 53% of telecom firms, 50% of e-commerce business, and 43% of retailers reporting DDoS attacks.

● 37% of DDoS attacks reported last for more than 24 hours, 24% lasted for more than three days, and 22% lasted for more than a week.

● According to recent market research, Botnet DDoS attacks are the most prevalent type of DDoS threat.

○ During the last quarter of 2011, one survey found 45% more DDoS attacks happened compared to the previous year

○ The average attack bandwidth observed during the last quarter of 2011 was 5.2 Gbps, which is 148% more than the previous quarter
More recent research found that ideologically motivated DDoS attacks are on the rise, supplanting financial motivation as the most frequent motivator of such DDoS attacks.

Many organizations protect their internal network resource and web applications by using sophisticated security measures such as firewalls, Intrusion Detection System etc., But, the hackers are far more intelligent in finding innovative ways to bypass the firewall and yet succeed in launching the DDoS attacks.

1.1.3 Common DDOS Attacks

CERT Coordination Center defines three basic types of Denial of Service attacks.

- Consumption of scarce, limited, or non renewable resources
- Destruction or alteration of configuration information
- Physical destruction or alteration of network components.

Some common DDoS attack types are discussed below.

1.1.3.1 SYN flood

In the SYN Flood attack, the attacker compromises the three-way-handshake for a TCP connection. In normal scenario between client and server communication, the TCP client sends a SYN packet to the TCP server. Upon receiving the SYN packet, the TCP server opens a session and sends back a SYN/ACK packet to the TCP client and waits for ACK packet for the three way handshake to be established. If the server does not receive the ACK, the server waits for a timeout and closes the session and releases the
resources. The attacker continually sends SYN packets to the server without sending the final ACK packet, thereby making the server to open multiple half open sessions, which in turn depletes the server resource. Figure 1.3 depicts a typical SYN Flooding scenario.

![SYN Flood scenario](image)

**Figure 1.3 SYN Flood scenario**

1.1.3.2 HTTP flood

In HTTP flood attacks, the attacker floods numerous HTTP request to access a web resource from the target web server. The requested resource is a large file, making the web server to spend its CPU resources to load the file. Mostly the attacker employs zombies, which mimics the normal web browsing behaviors.

1.1.3.3 ICMP flood (SMURF attack)

In ICMP Flood attack, the attacker floods ICMP echo request packets that have the victim server’s IP address, to a broadcast address. The ICMP echo reply comes from all the hosts in the network to the victim server IP address, thereby exhausting the bandwidth of the victim server network. DNS, ICMP protocols are commonly used for these kinds of attacks.
1.1.3.4 UDP flood

In UDP flood attack, the victim server network is overwhelmed by the large volume of UDP packets. These UDP packets are forwarded with random port numbers. If the application is not running on a specified port, the victim server will respond with an ICMP packet of “Destination unreachable” to the sender. As plenty of UDP packets with random ports are generated, massive UDP packets exhaust the resources on the victim server network.

1.1.3.5 DNS flood

DNS servers enable the clients to find the servers they are looking for. The client requests the IP address of a server by issuing the domain name. The DNS server resolves the domain name to IP address and vice versa. Attackers make use of this capability of the clients, by involving network of zombies to target a single DNS server with flood of valid request. It becomes very difficult for the DNS servers to distinguish from normally heavy traffic.

1.1.4 Botnet DDOS Attacks

A "bot" is a type of malware that allows an attacker to take control over an affected computer. Also known as “Web robots”, bots are usually part of a network of infected machines known as a “botnet”, which is typically made up of victim machines that stretch across the globe. A botnet is a collection of computers, connected to the internet, that interact to accomplish some distributed task.

Botnets have become the biggest threats on the Internet and are used for launching attacks and committing fraud. A study shows that, on a
typical day, about 40% of the 800 million computers connected to the Internet in a botnet. Those infected machines engage in many illegitimate activities, such as distributing spam, stealing sensitive information, launching denial-of-service attacks, and spreading new infections.

A botnet is a collection of software agents, or robots, that run autonomously and automatically. The term is most commonly associated with malicious software, but it can also refer to a network of computers using distributed computing software. While botnets are often named after their malicious software name, there are typically multiple botnets in operation using the same malicious software families, but operated by different criminal entities. While the term "botnet" can be used to refer to any group of bots, such as IRC bots, this word is generally used to refer to a collection of compromised computers (called zombie computers) running software, usually installed via drive-by downloads exploiting web browser vulnerabilities, worms, Trojan horses, or backdoors, under a common command-and-control infrastructure.

1.1.5 DDOS Attack Tools

Distributed Denial of Service attacks are under existence since mid 1980’s and are still the topmost web security threat. The vital reason behind this attack is the availability and sophistication of the attack tools. Examples of attack tools are: Trinoo, TFN2K, Shaft etc., The attack tools generate UDP Flooding, ICMP Flooding, TCP Flooding, Smurf attack etc.,

The below table gives details of DDoS attack tool and the type of attacks generated.
Table 1.1 Attack tools vs attack type generated

<table>
<thead>
<tr>
<th>DDoS attack tool</th>
<th>Attack type generated by the tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trinoo</td>
<td>UDP Flooding</td>
</tr>
<tr>
<td>TFN2K</td>
<td>UDP Flooding, TCP Flooding, Smurf, ICMP Flooding</td>
</tr>
<tr>
<td>Shaft</td>
<td>UDP Flooding, TCP Flooding, ICMP Flooding.</td>
</tr>
<tr>
<td>Stacheldraht</td>
<td>UDP Flooding, TCP Flooding, ICMP Flooding</td>
</tr>
<tr>
<td>Knight</td>
<td>UDP Flooding, TCP Flooding</td>
</tr>
<tr>
<td>Mstream</td>
<td>TCP Flooding</td>
</tr>
<tr>
<td>Trinity</td>
<td>UDP Flooding, TCP Flooding</td>
</tr>
</tbody>
</table>

1.2 DDOS DEFENSE TECHNIQUES

Several solutions are proposed by various researchers to overcome DDoS attacks in order to secure the networking environment from malicious attackers. The categories of defense mechanisms are:

- Firewall based protection
- Active Monitoring
- Overlay Networks
- Filtering Mechanism
- Capability based approaches
- Trace back and Pushback mechanism
- Filtering and Capability based Mechanisms
- Identification and Classification of Botnets
- CAPTCHA mechanism
1.2.1 Firewall Based Protection

Until the year 1996, the firewall was the basic means of protection for all sorts of network based attacks. Firewalls have simple rules, such as to allow or deny protocols, ports or IP addresses. Firewalls were also used to mitigate DDoS threats. Bailey et al (1996) proposed a methodology SYN Defender, which protects against the TCP SYN flood attacks by intercepting all SYN packets and mediating the connection attempts before they reach the operating system.

1.2.2 Active Monitoring

This category of solutions consists of using software agents to continuously monitor TCP/IP traffic in a network at a given place. It can watch for certain conditions to arise and react appropriately. Schuba et al (1997) proposed an active anomaly detection tool that can detect the conditions of a SYN flooding attack and react appropriately to defeat, or at least lessen the impact of, an attack. The researcher introduces, “synkill” that offers protection against SYN flooding for all hosts connected to the same local area network, independent of their operating system or networking stack implementation.

1.2.3 Overlay Networks

Here an overlay network is used to mitigate DDoS threat, where an overlay is a computer network which is built on top of another network. Nodes in the overlay can be thought of as being connected by virtual or logical links, each of which corresponds to a path, perhaps through many physical links, in the underlying network. Stone (2000) proposed an IP Overlay Network for tracking DoS floods called CenterTrack. It consists of IP tunnels or other connections that is used to selectively reroute interesting
datagrams directly from edge routers to special tracking routers. The tracking routers, or associated snipers, can easily determine the ingress edge router by observing from which tunnel the datagrams arrive. The datagrams can be examined, then dropped or forwarded to the appropriate egress points. CenterTrack basically re-routes all traffic aimed at a destination under attack, through a special-purpose intermediate node; because it is out of the normal path, the intermediate node can do sophisticated analysis and filtering.

Keromytis et al (2002) proposed a SOS: Secure Overlay Services architecture that proactively prevents DoS attacks, geared toward supporting Emergency Services or similar types of communication. The architecture is constructed using a combination of secure overlay tunneling, routing via consistent hashing, and filtering. The probability of successful attacks is reduced by

- Performing intensive filtering near protected network edges
- Introducing randomness and anonymity into the architecture.

Stavrou et al (2005) proposed a stateless multipath overlay network for countering DDoS attacks. Indirection-based overlay networks (IONs) are mechanisms based on the assumption that attackers will attack a fixed and bounded set of overlay nodes causing service disruption to a small fraction of the users. Andersen et al (2006) proposed an overlay network offering generic DDoS protection for targeted sites. OverDoSe clients and servers are isolated at the IP level. Overlay nodes route packets between a client and a server, and regulate traffic according to the server’s instructions. Through the use of light-weight security primitives, OverDoSe achieves resilience against compromised overlay nodes with a minimal performance overhead.
1.2.4 Filtering Mechanism

These methods employ a method of filtering of the packets based on some filtering rules. If an ISP is aggregating routing announcements for multiple downstream networks, strict traffic filtering should be used to prohibit traffic, which claims to have originated from outside these aggregated announcements. Some of such filtering mechanisms are given below.

Lee & Shieh (2005) proposed a new scheme, called ANTID, for detecting and filtering DDoS attacks which use spoofed packets to circumvent the conventional intrusion detection schemes by embedding in each IP packet, a unique path fingerprint, that represents the route an IP packet has traversed. ANTID is able to distinguish IP packets that traverse different Internet paths. Parno et al (2009) proposed an Assayer architecture that leverages hardware based attestation mechanisms to enable legitimate end hosts to embed secure proofs of code identity in packets. Receivers can specify traffic policies, which are enforced by on-path prioritizers. The primary contribution is the design and implementation of Assayer, a general architecture that allows a traffic recipient to filter traffic within the network based on properties of the sender. Liu et al (2009) highlighted on a novel closed-control, open-service architecture central to the StopIt design. Any receiver can use StopIt to block the undesired traffic it receives, yet the design is robust to various strategic attacks from millions of bots, including filter exhaustion attacks and bandwidth flooding attacks that aim to disrupt the timely installation of filters. The StopIt design uses a secure source authentication system Passport to prevent source address spoofing. Thus by employing the scheme stopit efficiently helps in controlling DoS.
1.2.5 Capability Based Approaches

Capabilities or tokens are used in these mechanisms for authentication purposes and also to classify between a legitimate and an attacker. Anderson et al (2003) proposed a capability based approach which poses a restriction on the exchange of information without prior permission from the target. Instead of being able to send anything to anyone at any time, the architecture demands, that the nodes must first obtain “permission to send” from the destination; a receiver provides tokens, or capabilities, to those senders whose traffic it agrees to accept. The senders then include these tokens in packets. Thus, only those packets with the tokens are allowed to pass the network. Parno et al (2007) proposed an approach to mitigate the Denial-of-Capability (DoC) attack, which prevents new capability-setup packets from reaching the destination, limits the value of these systems. Portcullis, the proposed system mitigates DoC attacks by allocating scarce link bandwidth for connection establishment packets based on per-computation fairness. This approach proves that a legitimate sender can establish a capability with high probability regardless of an attacker's resources or strategy and that no system can improve on our guarantee. Wang et al (2010) proposed web referral architecture to mitigate Denial of Service threats. In WRAPS (web referral architecture for privileged service), a website grants a client greater privilege to access its service by assigning to it a secret fictitious URL called privilege URL with a capability token embedded in part of the IP and port number fields. Through that URL, the client can establish a privileged channel with that website (referred to as target website) even in the presence of flooding attacks. A client may obtain a privilege URL either directly from the target website or indirectly from the website’s trusted neighbors. A website offers a client a privilege URL if the client is referred by one of the site’s trusted
neighbors, or is otherwise qualified by the site’s policies that are used to identify valued clients, for example, those who have paid or who are regular visitors. A qualified client will be redirected to the privilege URL generated automatically using that client’s identity, service information, and a server secret.

1.2.6 Traceback and Pushback Mechanism

Traceback mechanisms concentrates on identifying the hosts responsible for an attack and, like source filtering, does little to prevent sources from sending Pushback on the other hand employs dynamic traffic filters. Dynamic pushback is used to prevent resource exhaustion. With pushback, a node or link characterizes the types of packets causing the flood, and sends requests upstream to rate limit them closer to the source.

Savage et al (2000) describes a technique for tracing anonymous packet flooding attacks on the Internet back towards their source. It describes a general purpose traceback mechanism based on probabilistic packet marking in the network. This approach allows a victim to identify the network path(s) traversed by attack traffic without requiring interactive operational support from Internet Service Providers (ISPs). This traceback can be performed “post-mortem after an attack has completed. An algebraic approach to IP traceback is proposed by Matt et al (2001) for providing the traceback data by having routers embed information randomly into packets. It also reframes the traceback problem as a polynomial reconstruction problem and uses techniques from algebraic coding theory to provide robust methods of transmission and reconstruction. Ioannidis et al (2002) treats DDoS attacks as a congestion-control problem, but because most such congestion is caused by malicious hosts not obeying traditional end-to-end congestion control, the
problem is handled by the routers. Functionality is added to each router to
detect and preferentially drop packets that probably belong to an attack.
Upstream routers are also notified to drop such packets (hence the term
Pushback) in order that the router’s resources be used to route legitimate
traffic.

1.2.7 Filtering and Capability Approaches

These are those approaches that employ both the concepts of
filtering and capability based mechanisms to defend DoS attacks. Yaar et al
(2004) proposed a Stateless Internet Flow Filter (SIFF), which allows an end-
host to selectively stop individual flows from reaching its network, without
any of the common assumptions is proposed. The network traffic is divided
into two classes, privileged (prioritized packets subject to recipient control)
and unprivileged (legacy traffic). Privileged channels are established through
a capability exchange handshake. Capabilities are dynamic and verified by the
routers in the network, and can be revoked by quenching update messages to
an offending host. SIFF is transparent to legacy clients and servers, but only
updated host will enjoy the benefits of it. Thus architecture involves both the
filtering and the capability mechanism to mitigate DoS attacks.

1.2.8 Identification and Classification of Botnets

Seewald & Gansterer (2009) proposed a passive approach to detect
and identify the botnets. A passive botnet defense approach is proposed at
three hierarchical levels, namely the level of a single packet, network access
and TCP conversations. At the level of a single packet, the payload is
distinguished as malicious or spam. At the network access level, the traffic is
analyzed to a given darknet and at the level of TCP conversations, the legitimate TCP conversations are distinguished from the illegitimate ones.

1.2.9 Captcha Based Mechanism

Several works are in the literature that provides an application level defense mechanism. Google, Yahoo and Hotmail uses text based CAPTCHA (Computer Aided Public Turing Test to tell Computer Human Apart) as an application level defense mechanism.

Table 1.2 CAPTCHA images from Google, Yahoo and Hotmail

<table>
<thead>
<tr>
<th>Google CAPTCHA</th>
<th>Yahoo CAPTCHA</th>
<th>Hotmail CAPTCHA</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Google CAPTCHA" /></td>
<td><img src="image2" alt="Yahoo CAPTCHA" /></td>
<td><img src="image3" alt="Hotmail CAPTCHA" /></td>
</tr>
</tbody>
</table>

Among them, Basso et al (2009) proposed a scheme, for generating image CAPTCHA which guarantees better security over text-based CAPTCHA mechanisms and ensures user-friendliness of the system. The scheme exploits the current attack tool’s difficulty in performing image segmentation in the presence of complex background and recognition of specific concepts from background clutter. Datta et al (2009) proposed a design of picture based CAPTCHA methodology wherein they introduced controlled distortions of images which can potentially make automated tools recognition harder without affecting human recognition. The systematic
distortions make the automated image making and recognition, a very hard AI problem. Banday et al (2009) proposed an Image Flip CAPTCHA methodology, which occupies less screen area, making it compatible for web pages. The CAPTCHA image is formed with subset of flipped and non-flipped images. Security analysis is based on farming out attacks, attacks that involve use of segmentation, shape matching and random guessing. However, the methodology expects the user to click all the images that are not flipped, which in turn increases the access time to web pages.

Some important types of CAPTCHA test are:

- Text-based test
- Image-based test
- Audio-based test

A good CAPTCHA possesses also the following properties:

- It should be taken quickly and easily by human users.
- It should accept all human users (without any discrimination) with high reliability while rejecting very few.
- Virtually no machine should be able to solve it.
- It should resist attacks even if the algorithm and its data are known.

1.2.10 Core Challenges in DDoS Defense

Although many DDoS mitigation methods are proposed by various researchers since 1990, protecting the organization's network from DDoS traffic still remains a major concern. While the number of freeware packages
available for generating DDoS traffic is increasing each year, the detection of most of these methods is neither satisfactory nor fully automated.

- Attackers are widely distributed and so the attack agents. Wang et al (2010) proposed a design for hybrid peer to peer botnet architecture, which is, widely distributed, highly scalable, difficult to be shut down and is harder for defenders to identify these bots. Although the defenders were able to remove considerable botnet population, this robust architecture has the capability to sustain control over the rest of the bots and prevents significant exposure of the botnet network topology. The attacker’s army is more powerful as they have millions of zombies into their network and they manage their network better in order to target the servers.

- Text based CAPTCHAs (Computer Aided Public Turing Test to tell Computer and Human Apart), used by a majority of the websites today, are susceptible to application level DDoS attacks. There are several automated programs that can read text CAPTCHA and thereby send illegitimate application level traffic to the web server.

- The difficulty of security devices, such as firewall, in distinguishing between legitimate and illegitimate network traffic. Botmasters generate illegitimate traffic as per the protocol specification. The illegitimate traffic often looks genuine protocol traffic.
1.3 RATIONALE FOR THE PROPOSED SYSTEM

There are four prime motives behind the design of the proposed DDoS Mitigation architecture for web servers. They are

- Proposal of Comprehensive Defense Mechanism
- Identification of image distortion features for generation of Picture based CAPTCHA.
- Identification of key features in botnet identification and classification
- Identification of components involved in DDoS resilient server side architecture.

1.4 OBJECTIVES OF THE THESIS

The objective of the thesis work is to design and develop a defense in depth architecture for mitigation of DDoS attacks on web servers with the following aims:

- To mitigate DDoS attacks on the application layer and the Network Layer of the OSI model.
- To design and develop a distributed referral architecture in order to divide and conquer the DDoS attacks and to provide secure access to the web server resources to the legitimate clients.
- To propose a novel architecture for picture-based CAPTCHA (Computer Aided Public Turing Test to Tell Computers and Human Apart) technique as an application level defense
mechanism, which provides improved security over text based CAPTCHA and other picture-based CAPTCHA techniques by ensuring user friendliness of the system.

- To devise a novel architecture for identifying and classifying the botnets, as a network level defense mechanism and providing significant inputs to the firewall to filter the attack packets.

- To propose a DDoS tolerant server side architecture in order to enhance the system resilience to DDoS attacks.

1.5 THESIS CONTRIBUTION

Comparing with all the defense mechanisms proposed in the literature, the proposed work is different in many ways. Firstly, the referral architecture proposed defends the DDoS requests using multiple Trusted Referral Servers at the application layer. Secondly, A novel architecture for generation of picture based CAPTCHA is proposed which provides improved security over existing CAPTCHA techniques. Thirdly, Linear Support Vector Machines, a machine learning algorithm, are used for the purpose of identifying and classifying the botnets at the network level. Finally, the new server side architecture is proposed which is resilient to DDoS attacks. The combination of application level defense mechanism, network level defense mechanism and DDoS tolerant Server architecture makes our approach different from many other existing techniques.

1.5.1 Defense Mechanism at the Application Layer

Defense mechanism at the Application Level is to prevent automated tools access to web resources are based on Referral Architecture
and Picture based CAPTCHA Application. The objective is to filter numerous illegitimate accesses to web resources and provide access to the legitimate clients

1.5.2 Defense Mechanism at the Network Layer

The objective of this phase is to analyze the IP traffic for identifying and classifying the botnets, identifying and classifying the virus signatures in the payload, identifying the existence of co-relation between the spambot likeliness and viral payload.

1.5.3 DDOS Resilient Server Side Architecture

The objective of this module is to design a DDoS tolerant generic architecture for the web server. This architecture has the following characteristics: redundant and heterogeneous web servers, virtualization, IPSec protocol, Intermediary Proxy server with Load Balancer, Network Intrusion Detection System, and IPS manager.

1.6 THESIS ORGANIZATION

The organization of the thesis is provided as follows:

Chapter two summarizes the literature review of the state-of-the-art Distributed Denial of Service defense research works and findings.

Chapter three elucidates the functional framework of the system proposed in this research.

Chapter four illustrates the application level solution proposed in this research work.
Chapter five explains the network level solution and the DDoS resilient server side architecture proposed in this research work.

Chapter six concludes this research and discusses some promising future research lines in mitigation of Distributed Denial of Service attacks.