CHAPTER 2

A SURVEY OF NETWORK ATTACK DETECTION
TECHNIQUES

2.1 INTRODUCTION

The Internet was originally designed for openness and scalability. It is a global system of interconnected computer networks that use the standard Internet protocol suite called as Transmission Control Protocol/Internet Protocol (TCP/IP) to serve several billion users globally. Internet Protocol (IP) was designed to support ease of attachment of hosts to networks, and provides little support for verifying the contents of IP packet header fields described by David D. Clark (1988).

The design architecture of internet permits the user to access its resources anywhere at any time. The emergence of the computer networks brought us maximum benefits likes resource sharing, reliability, better performance ration and so on that are impossible provide by traditional centralized systems.

However, the prices of this benefits has been more security threats like unauthorized access to private information, malicious break in to other organizations systems, or intent to render a system to make it unreliable or unusable.
In computer networks attackers are generally adopts transmission media to put across attack. The network attacks happened around the TCP/IP protocol suite, which is the most widely used communication protocol and the de facto standard among the Internet society.

Several types of network attacks have been found up till now each of them utilizes one or more security vulnerabilities in the TCP/IP protocol specification or some well-known weakness in the implementations of TCP/IP. There includes, IP address spoofing, TCP sequence prediction, SYN flooding attack, DNS misuse, Ping of Death, or some newly emerged Java related attacks. However according to their basic techniques and impacts to the victim system, it is dividing into two basic categories as denial of service attacks and IP spoofing attacks described by Guang Yang (1997).

During data packet transmission in computer network it is easy to fake the source IP address of packets, and also it is very difficult to identify the source IP of each packet. Moreover, there is no inherent support in the TCP/IP layer to check whether a source is authorized to access a service. When a packet delivered to their destination, the respective server at the destination must decide whether to accept and service these packets.

While defenses from these attacks, defense technique such as firewalls can be added to protect servers, a key challenge for defense is how to discriminate legitimate requests for service from malicious access attempts. If it is easier for sources to generate service requests than it is for a server to check the validity of those requests, then it is difficult to protect the server from malicious requests that waste the resources of the server. This creates the opportunity for variety of IP based network attacks. The following section will discuss the detection and prevention of different type of network attacks like DoS attacks, bandwidth attacks, and protocol based attacks etc.
Because these attacks are more vulnerable to the performances of network operations.

2.2 DOS (DENIAL OF SERVICE) ATTACKS

The denial-of-service attacks attempt to disrupt the service provided by a network or a server. It can be launched in two forms illustrated by Hussain et al (2000). The first form aims to crash a system by sending one or more carefully crafted packets that exploit software vulnerability in the target system. For example, the “ping-of-death” attack sends a large numbers of Internet Control Message Protocol (ICMP) ping packet that is fragmented into multiple datagram to a target system, which can cause certain operating systems to crash, freeze, or reboot due to buffer overflow (CERT 1996).

The second form is to use massive volumes of useless traffic to occupy all the resources that could service legitimate traffic. While it is possible to prevent the first form of attack bypatching known vulnerabilities, the second form of attack cannot be so easily prevented. The targets can be attacked simply because they are connected to the public Internet.

When the traffic of a DoS attack comes from multiple sources, it is called as Distributed Denial of Service (DDoS) attack. By using multiple attack sources, the power of a DDoS attack is amplified and the problem of defense is made more complicated. The impact of DDoS attacks can vary from minor inconvenience to users of a Web site to serious financial losses for companies that rely on their online availability to do business.

On February 9, 2000, Yahoo, eBay, Amazon.com, E*Trade, ZDnet, Buy.com, the FBI, and several other Web sites fell victim to DDoS attacks resulting in substantial damage and inconvenience quoted by Garber, L (2000). From December 2005 to January 2006, 1,500 separate IP addresses...
were victims of DDoS attacks, with some attacks using traffic rates as high as 10 Gb/s quoted by Scalzo (2006), Vaughn and Evron (2006).

At present our day to day essential services, such as banking, transportation, power, health, and defense, are being progressively done through the efficient Internet-based applications. At the same time Internet based attacks can be launched anywhere in the world, and unfortunately any Internet based service is a potential target for these attacks. As emergency and essential services become reliant on the Internet as part of their communication infrastructure, the consequences of DDoS attacks could even become life threatening. Hence, it is crucial to deter, or otherwise minimize, the damage caused by DDoS attacks.

2.3 BANDWIDTH ATTACKS

The major impacts of bandwidth attacks are the consumption of the host’s resources. Generally, the victim could be a Web server or proxy server connected to the Internet. The victim has limited resources to process the incoming packets. When the traffic load becomes high, the victim will drop packets and inform senders, since the traffic consist of both legitimate users and attack sources to reduce their sending rates. Up on receiving the request the legitimate users will slow down their sending rates but the attack sources will maintain or increase their sending rates. Eventually, the victim’s resources, such as CPU and memory, will be exhausted and the victim will be unable to service legitimate traffic.

The other impact of this attack is the consumption of network bandwidth, which is more disruptive than the first. If the malicious flows are able to dominate the communication links that lead to the victim, then the legitimate flows will be blocked. Therefore, not only the intended victim of the attack is disabled, but also any system that relies on the communication
links of the attack path. Although a congested router can control the traffic flow by dropping packets, legitimate traffic will also be discarded if there is no clear mechanism to differentiate legitimate traffic from attack traffic.

The term attack power is referred by Tao Peng et al (2007) indicates the level of resources consumed at the victim by the attack described by. Generally, the attack power consists of two parameters. The first parameter is the traffic volume, which can be represented by the number of packets in a given period. The second parameter is the level of resources consumed per packet, which can be represented by CPU time or memory needed to process the packet.

2.4 PROTOCOL-BASED ATTACKS

A protocol-based attack can normally be launched effectively from a single attack source. The depth of this type of attack power is based on specific weaknesses of the Internet protocols. There are two types of protocol-based attack namely SYN floods and ICMP floods attacks.

2.4.1 SYN Flood Attack

During the normal TCP connection establishment, the server expects to receive a connection request. In the mean time the client will negotiate with the server to set up a connection, which is called a three-way handshake. Figure 2.1 shows the normal TCP connection establishment. Initially the client will send a SYN packet to the server and requests for a TCP connection. Then the server will respond to the connection request using a SYN-ACK packet, and store the request information in the memory stack described by Wright and Stevens (1995). Under BSD-style network software three memory structures are allocated once a SYN packet is received, that is, socket (A network socket is an endpoint of an inter-process communication
flow across a computer network), inpcb(Internet protocol control block), and tcpcb(TCP control block). These data structures are used to store the details of the requested TCP connection, and their combined size for a single TCP connection may typically exceed 280 B described by Schuba et al (1997). At this point, a connection is in a half-open state, called the SYN RECV state RFC 793(1981).

![Figure 2.1 Three Way Handshake](image)

To prevent the system from depleting its memory, each operating system will limit the number of concurrent TCP connections in the SYN RECV state. After receiving the SYN-ACK packet, the client will confirm the request using an ACK packet. When the server receives the ACK packet,
it checks the memory stack to see whether this packet is used to confirm an existing request. If it is, then the TCP connection is moved from the SYN RECV state to the ESTABLISHED state. After this, the client and server have finished the three-way handshake and can start data transfer. Another way to remove a connection in the SYN RECV state is to either send a RST packet or wait until its timer expires.

The SYN flood attack exploits a vulnerability of the TCP three-way handshake, namely, that a server needs to allocate a large data structure for any incoming SYN packet regardless of its authenticity. During SYN flood attacks, the attacker sends SYN packets with source IP addresses that do not exist or are not in use. During the three way handshake, when the server puts the request information into the memory stack, it will wait for the confirmation from the client that sends the request. While the request is waiting to be confirmed, it will remain in the memory stack. Since the source IP addresses used in SYN flood attacks can be nonexistent, the server will not receive confirmation packets for requests created by the SYN flood attack. Each half-open connection will remain on the memory stack until it times out. That is a 2 MSL (2 maximum segment lifetimes) timer measures the time, a connection have been in the time wait state. More and more requests will accumulate and fill up the memory stack. Therefore, no new request, including legitimate requests, can be processed and the services of the system are disabled.

Generally, the space for the memory stack allocated by the operating system is small, and even a small scale SYN flood attack can be disruptive. On the other hand, SYN floods can be also launched from compromised machines using legitimate source IP addresses given these compromised machines are configured to ignore the SYN/ACK packets from
the target. SYN floods remain one of the most powerful flooding attack methods.

### 2.4.2 ICMP Flood Attack

The Internet Control Message Protocol (ICMP) is based on the IP protocol and is used to diagnose network status. An ICMP flood is a type of bandwidth attack that uses ICMP packets. On IP networks, a packet can be directed to an individual machine or broadcast to an entire network. When a packet is sent to an IP broadcast address from a machine on the local network, that packet is delivered to all machines on that network. When a packet is sent to that IP broadcast address from a machine outside the local network, it is broadcast to all machines on the target network as long as routers are configured to pass along that traffic.

IP broadcast addresses are usually network addresses with the host portion of the address having all one bits. For example, the IP broadcast address for the network 10.*.*.* is 10.255.255.255, and for the network 10.50.*.* is 10.50.255.255. Network addresses with all zeros in the host portion, such as 10.50.0.0, can also produce a broadcast response.

The “smurf” attack is a type of ICMP flood, where attackers use ICMP echo request packets directed to IP broadcast addresses from remote locations to generate denial of service attacks. There are three parties in these attacks they are the attacker, the intermediary and the victim it is understood that the intermediary can also be a victim. During this attack first the attacker sends one ICMP echo request packet to the network broadcast address and the request is forwarded to all the hosts within the intermediary network. Second, all of the hosts within the intermediary network send the ICMP echo replies to flood the victim. Solutions to the smurf attack are discussed in CERT (1998),
which includes disabling the IP-directed broadcast service at the intermediary network. Nowadays, smurf attacks are quite rare in the Internet.

### 2.4.3 Application-Based Attacks

Another way to amplify attack power is to force the target to execute expensive operations. For example, many Web sites provide search engines to allow users to find a particular Web page. An attacker can exploit this application by sending a large number of queries to a Web site’s search engine.

In this way, the Web site is forced to perform CPU and memory-intensive database operations and leave few resources to serve legitimate users. These types of attack are called as an application-based attack, which aims to take advantage of the disproportionately large resource consumption at the server. This section describes attacks that target two important Internet applications, namely, the World Wide Web and Voice over IP described by Rosenberg et al (2002).

### 2.4.4 HTTP Flood Attack

The World Wide Web (WWW) is one of the most popular applications currently running on the Internet and has driven the rapid growth of the Internet described by Wang (1999). WWW applications generally use the Hypertext Transfer Protocol (HTTP) over TCP port 80. Thanks to this popularity, most firewalls on the Internet will leave TCP port 80 open to allow HTTP traffic to pass. Unfortunately, the ubiquity of WWW applications has also made HTTP a prime target for attackers. Generally, an HTTP flood refers to an attack that bombards Web servers with HTTP requests.
According to a recent study by Honeynet (2005), HTTP floods have become a common feature in most botnet software. A botnet (also known as a zombie) is a number of Internet computers that, although their owners are unaware of it, have been set up to forward transmissions (including spam or viruses) to other computers on the Internet. To send an HTTP request, a valid TCP connection has to be established, which requires a genuine IP address. Attackers can achieve this by using a botnet’s IP address.

Moreover, attackers can craft the HTTP requests in different ways in order to either maximize the attack power or avoid detection. For example, an attacker can instruct the botnet to send HTTP requests to download a large file from the target. The target then has to read the file from hard disk, store it in memory, load it into packets, and then send the packets back to the botnet. Hence, a simple HTTP request can incur significant resource consumption in the CPU, memory, input/output devices, and outbound Internet link.

However, the behavior of the HTTP requests of the previous example can be conspicuous. Repetitive requests to a large file can be detected and hence blocked. To better mimic legitimate traffic, attackers can instruct the botnet to send an HTTP request to the target Web site, then parse the replies and follow the links recursively. In this way, the HTTP requests from the attacker are very close to normal Web traffic, which makes it extremely difficult to filter this type of HTTP flood.

2.4.5 SIP Flood Attack

In the past few years, the deployment of Voice over IP (VoIP) telephony has become popular thanks to its low cost. A widely supported open standard for call setup in VoIP is the Session Initiation Protocol (SIP) described by Rosenberg et al (2002).
Generally SIP proxy servers require public Internet access in order to accept call setup requests from any VoIP client. Moreover, to achieve scalability, SIP is typically implemented on top of UDP in order to be stateless. For example if a person A wants to talk to B, A will first send an Invite packet to B. Generally, this packet is sent to A SIP proxy server, which will look up the address of B’s SIP proxy server and send an Invite packet to that proxy. When B’s SIP proxy receives the Invite packet, it will pass it to B’s registered address and B’s phone will ring. After this, either Bob picks up the phone to start the conversation or there is no answer.

In one attack scenario, the attackers can flood the SIP proxy with many SIP Invite packets that have spoofed source IP addresses quoted by Sisalem et al (2005), Kuhn et al (2005), Chen (2006). To avoid any anti-spoofing mechanisms, the attackers can also launch the flood from a botnet using nonspoofed source IP addresses.

There are two categories of victims in this attack scenario. The first categories of victims are the SIP proxy servers. Not only will their server resources be depleted by processing the SIP Invite packets, but their network capacity will also be consumed by the SIP Invite flood. In either case, the SIP proxy server will be unable to provide VoIP service. The second categories of victims are the call receivers. They will be overwhelmed by the forged VoIP calls, and will become nearly impossible to reach by legitimate callers.

2.4.6 Distributed Reflector Denial of Service Attacks

An important goal for attackers is to hide the true sources of their attack traffic. Paxson (2001) illustrates another type of bandwidth attack called a Distributed Reflector Denial of Service (DRDoS) attack as shown in figure 2.2, which aims to obscure the sources of attack traffic by using third
parties either routers or a Web servers to relay attack traffic to the victim. These innocent third parties are also called reflectors. Any machine that replies to an incoming packet can become a potential reflector.

![Diagram of DRDoS attack]

CS – Computer System

**Figure 2.2 DRDoS attack**

The DRDoS attack contains three stages. The first stage is very similar to the typical DDoS attack. However, in the second stage, after the attacker has gained control of a certain number of “zombies,” instead of instructing the “zombies” to send attack traffic to the victims directly, the “zombies” are ordered to send to the third parties spoofed traffic with the victim’s IP address as the source IP address. In the third stage, the third parties will then send the reply traffic to the victim, which constitutes a DDoS
attack. This type of attack shut down www.grc.com, a security research Web site, in January 2002, and is considered to be a potent and increasingly prevalent Internet attack.

In comparison to a traditional DDoS attack, the traffic from a DRDoS attack is further dispersed by using the third parties, which makes the attack traffic even more distributed and difficult to identify. Moreover, the source IP addresses of the attack traffic are from innocent third parties, which make attack source traceback extremely difficult. Finally, as informed by Gibson (2002), DRDoS attacks have the ability to amplify the attack traffic, which makes the attack even more potent.

2.4.7 DNS Amplification Attack

Particularly an effective form of reflector attack makes use of the existing Domain Name System (DNS) server described by Mockapetris (1987). The role of the Domain Name System is to provide a distributed infrastructure to store and associate different types of resource records (RR) with Internet domain names, such as amity.edu.au. Relevant examples of resource records include type TXT RR, which allow an administrator to insert arbitrary text into a DNS record; type A RR, which map a host name into a 32-bit IP address; and type Start of Authority (SOA) RR, which provide the name of the primary source of an Internet domain and other related information.

One important function of DNS is to translate domain names into IP addresses. A recursive DNS server accepts a query and resolves a given domain name on behalf of the requester. Generally, a recursive name server will contact other authoritative names servers if necessary and eventually
return the query response back to the requester. The sizes of the DNS query and query response are disproportional.

Normally, a query response includes the original query and the answer, which means the query response packet, is always larger than the query packet. Moreover, one query response can contain multiple types of RR, and some types of RR can be very large. The relation between a request and the corresponding response is known as the amplification factor and is computed using the following formula:

\[
\text{Amplification Factor} = \frac{\text{size of (response)}}{\text{size of (request)}}
\]

For example, if a DNS name server receives a 60-B Extended DNS (EDNS) query described by Vixie (1999) containing a large buffer advertisement, its reply can include a 122-B type A resource record, a 4000-B type TXT resource record, and a 222-B type SOA resource record defined by Vaughn and Evron (2006). This renders an amplification factor of 73.

### 2.4.8 Infrastructure Attacks

An infrastructure attack aims to disable the services of critical components of the Internet. The result of an infrastructure attack is potentially catastrophic as the whole Internet may be affected. For example, DNS root servers provide information about the servers that are responsible for top-level domains, such as .com. They are indispensable elements to enable DNS to function.

An infrastructure attack can tie up both the network and host resources of a DNS root server, disrupting all Internet services that depend on these servers. On 21 October 2002, all Internet DNS root servers were attacked simultaneously by coordinated distributed denial of service attacks. The attack lasted about 1 h and 15 min, and the attack volume was approximately 50 to 100 Mb/s (100 to 200 kpts/s) per root name server,
yielding a total attack volume of approximately 900 Mb/s (1.8 Mpkt/s) described by Vixie et al (2002). If the attacker increased the attack traffic rate or extended the attack time, more catastrophic damage would have been done to the overall Internet. A detailed analysis of attacks against DNS can be found in Cheung (2006).

2.5 NETWORK ATTACK DEFENSE TECHNIQUES

Generally, there are four broad categories of defense against network 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 other word it refers to filtering spoofed packets close to or at the attack sources, which is one of the most effective defense approaches for DoS attacks that use spoofed traffic. Attack detection aims to detect DoS attacks when they occur.

Attack detection is an important procedure to direct any further action. 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 DoS 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.
2.6 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 only valid (nonspoofed) 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. Moreover, some types of filtering schemes require wide deployment to be effective. Unfortunately, the Internet is an open community without central administration, which makes prevention a taxing and overwhelming task.

2.6.1 Ingress/Egress Filtering

Ingress filter and egress filter are implemented at the end router to filter out unwanted IP packets developed by Ferguson and Senie (2000). The aims of ingress filter are filtering the traffic coming into a local network, and egress filtering means filtering the traffic leaving your local network.

The principle operation of ingress/egress filtering is to strictly allow data traffic to enter or leave the network if its source addresses are within the expected IP address range. For an example an internet service provider X (ISP X) provides internet access to a leaf network of an organization where router 1 is the edge router for the leaf network which is connected to router 2 the edge router for ISP X. similarly router 3 is another edge router for ISP X which is used to interconnect with other networks.
In this scenario, suppose an attacker A resides within the leaf network. An input filter is placed in the input port of router 2 that is connected to the leaf network. This input filter only admits packets having a source IP address with the 220.65.201.0/24 prefix. If attacker A sends traffic with spoofed IP addresses that do not have the 220.65.201.0/24 prefix, that traffic will be dropped by the input filter in router 2. This filtering function provided by router 2 is called ingress filtering as it deals with traffic coming into the network of ISP X. However, if router 1 provides the same function, that function is called egress filtering as it deals with traffic leaving the leaf network.

A key requirement for ingress or egress filtering is knowledge of the expected IP addresses at a particular port. For some networks with complicated topologies, it is not easy to obtain this knowledge. One technique defined by Baker (1995) known as reverse path filtering can help to build this knowledge. The technique works as follows. Generally, a router always knows which networks are reachable via any of its interfaces. By looking up source addresses of the incoming traffic, it is possible to check whether the return path to that address would flow out the same interface as the packet arrived upon. If they do, these packets are allowed. Otherwise, they are dropped. Unfortunately, this technique cannot operate effectively in real networks where asymmetric Internet routes are not rare.

### 2.6.2 Router-Based Packet Filtering

Router based Packet Filtering (RPF) developed by Park and Lee (2001a) extends with the working principle of 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.

In order to understand the operation of RPF, it is mandatory to know the key concepts of inter domain routing in the Internet. The Internet is divided into a set of routing domains, known as Autonomous Systems (ASs), where each AS corresponds to one or more networks that are controlled by a single administration entity.

During the data packets transmission traffic is routed between ASs by border routers that use the Border Gateway Protocol (BGP) described by Rekhter and Li (1995). Each AS has one or more border routers depending on its topology, and is identified by a unique 16-bit Autonomous System ID. When viewed at the level of ASs, the whole Internet is connected by border routers.

For example, in Figure 2.2 each node represents a border router for one AS. At this point, assume the terms AS and border router interchangeably. RPF uses information about the BGP routing topology to filter traffic with spoofed source addresses.

Consider the example network in Figure 2.2, where an attack source in AS7 is flooding a target in AS4 with DoS attack traffic. The attack traffic (shown using bold arrows) has been spoofed so that its source address appears to come from AS3. Suppose RPF is deployed at AS6. The attack traffic from AS7 can be filtered if AS6 knows the BGP routing topology in the network. In particular, consider the routing topology for all paths from AS3 (shown as normal arrows), which is the spoofed source address of the attack traffic. Given this routing topology, there is no way that traffic from AS3 could arrive at the RPF at AS6 on the link from AS7 to AS6. Thus, all
attack traffic that uses the spoofed source address of AS3 can be filtered at AS6, since it arrives on the link from AS7.

There are several limitations of this scheme. The first limitation relates to the implementation of RPF in practice. Given that the Internet contains more than 10,000 ASs, RPF would need to be implemented in at least 1800 ASs in order to be effective, which is an onerous task to accomplish. Moreover, RPF requires modifications to the BGP message scheme by Rekhter and Li (1995), so that source addresses are included in BGP messages. This would significantly increase the size and processing time for BGP messages.

The second limitation is that RPF may drop legitimate packets if there has recently been a route change. For example, consider the case where the route from AS3 to AS6 has changed due to a link failure or a policy change. The new route traverses the AS path 3-5-10-9-7-6, as shown by the dashed arrows in Figure 2.3. If the RPF in the border route of AS6 has not been updated with this information, then legitimate packets from AS3 to AS4 will be dropped at AS6.

The third potential limitation is that RPF relies on valid BGP messages to configure the filter. If an attacker can hijack a BGP session and disseminate bogus BGP messages, then it is possible to mislead border routers to update filtering rules in favor of the attacker. Finally, the filtering rules in RPF have a very coarse granularity, that is, at the AS level. The attacker can still spoof IP addresses based on the network topology. Alternatively, the attacker can launch the attack from compromised systems, without resorting to IP address spoofing.
Figure 2.3 Route based packet filtering

RPF is proposed for deployment in core networks. In general, a packet needs to pass multiple RPF filters before reaching the destination. Since it is difficult for an attacker to choose a path without a single RPF filter, RPF is effective against randomly spoofed DoS attacks. However, the filtering granularity of RPF is low. This means that the attack traffic can still bypass the RPF filters by carefully choosing the range of IP addresses to spoof.

2.6.3 Spoofing Prevention Method

Another new approach for filtering spoofed IP packets called as spoofing prevention method (SPM) is proposed by Bremler Barr et al (2005) and this method enables routers closer to the destination of a packet to verify the authenticity of the source address of the packet. This stands in contrast to standard ingress filtering which is effective mostly at routers next to the source and is ineffective otherwise. In the proposed method a unique temporal key is associated with each ordered pair of source destination networks of autonomous systems. Each packet leaving a source network S is tagged with
the key \( K \) associated with \((S, D)\) where \( D \) is the destination network. Upon arrival at the destination network the key is verified and removed. Thus the method verifies the authenticity of packets carrying the address which belongs to network \( S \).

It is an efficient method ensuring not to overload the routers. The major benefits of the method are the strong incentive it provides to network operators to implement it, and the fact that the method lends itself to stepwise deployment, since it benefits networks deploying the method even if it is implemented only on parts of the Internet. These two properties in this approach make it an attractive and viable solution to the packet spoofing problem.

### 2.6.4 Automatic Peer-To-Peer Anti-Spoofing (APPA)

Automatic Peer-to-Peer based Anti-spoofing method (APPA) is a signature and verification based IP spoofing prevention method and proposed by Shen et al (2008). APPA has two levels: intra-AS (autonomous system) level and inter-AS level. In the intra-AS level, the end host tags a one-time key into each outgoing packet and the gateway at the AS border verifies the key. In inter-AS level, the gateway at the AS border tags a periodically changed key into the leaving packet and the gateway at border of the destination AS verifies and removes the key.

The most prominent characteristic of APPA is the automatically synchronizing state-machine, which is used to update keys automatically and effectively. The advantages of APPA are it prevents IP address spoofing with the end systems encapsulate even spoof addresses in the same AS or subnet, providing very low running and management costs and supporting anti-replay attacks and incremental deployment.
Another spoofing prevention method proposed by Bi (2009) where IP Source Address Spoofing is prevented using the concept of State Machine along with the APPA method. In this method, signatures are tagged into the packets at the source peer, and verified and removed at the verification peer where packets with incorrect signatures are filtered.

A unique state machine, which is used to generate signatures, is associated with each ordered pair of APPA peers. As the state machine automatically transits, the signature changes accordingly.

KISS random number generator is used as the signature generating algorithm, which makes the state machine very small and fast and requires very low management costs. APPA has an intra AS (autonomous system) level and inter AS level. In the intra AS level, signatures are tagged into each departing packet at the host and verified at the gateway to achieve finer grained anti-spoofing than ingress filtering. In the inter AS level, signatures are tagged at the source AS border router and verified at the destination AS border router to achieve prefix level anti spoofing, and the automatic state machine enables the peers to change signatures without negotiation which makes APPA attack-resilient compared with the spoofing prevention method. The results show that the two levels are both incentive for deployment, and they make APPA an integrated anti spoofing solution.

2.6.5 Route-Based Distributed Packet Filtering (DPF)

A novel approach to distributed DoS (DDoS) attack prevention is described and evaluate by Kihong Park, et al (2001) as route-based distributed packet filtering (DPF). This filtering method achieves proactiveness and scalability, and there is an intimate relationship between the electiveness of DPF at mitigating DDoS attack and power-law network topology.
The salient features are two-fold. First, one is able to proactively filter out a significant fraction of spoofed packet flows and prevent attack packets from reaching their targets in the first place. The IP flows that cannot be proactively curtailed are extremely sparse so that their origin can be localized i.e., IP traceback to within a small, constant number of candidate sites.

It is show that the two proactive and reactive performance effects can be achieved by implementing route based filtering on less than 20% of Internet autonomous system (AS) sites. Second, we show that the two complementary performance measures are dependent on the properties of the underlying AS graph. In particular, we show that the power-law structure of Internet AS topology leads to connectivity properties which are crucial in facilitating the observed performance effects.

2.6.6 Inter Domain Packet Filters (IDPF)

Another type of preventing network attack using an inter-domain packet filters (IDPF) implemented by Duan et al (2008). This filter can mitigate the spoofing attack on the Internet. IDPFs are developed from the information implicit in BGP route updates and are deployed in network border routers. A key feature of the scheme is that it does not require global routing information.

The IDPF framework works correctly in that it does not discard packets with valid source addresses. IDPFs can proactively limit the spoofing capability of attackers even with partial deployment on the Internet. In the Internet there are a lot of distributed denials of service (DDoS) attacks. A lot of attacks aim to cause damage to network services applications. One of the efficient methods to protect regular traffic from the attacks called FSN
(Filtering of IP Spoofed packet near the attacker) method developed by Ohtsuka et al (2007a).

FSN method is effective and practical and applicable to the real Internet environment. It uses topology information to detect the attacks and collects topology information using IGP routing protocol, so it is applicable to the environments including asymmetric paths and it doesn't require collected packets to construct neighbor information. Realization of the above method is developed by Ohtsuka et al (2007b) with the concept of Open Shortest Path First (OSPF) to detect the DDoS attack.

2.6.7 Source Address Validity Enforcement (SAVE) Protocol

From the earlier discussion it is understood that 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 limitation, Li et al (2002) have proposed a new protocol called the Source Address Validity Enforcement (SAVE) protocol, 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.

The aim of the SAVE protocol is to provide routers with information about the range of source IP addresses that should be expected at each interface. Similarly to the existing routing protocols, SAVE constantly propagates messages containing valid source address information from the source location to all destinations. Hence, each router along the way is able to build an incoming table that associates each link of the router with a set of valid source address blocks. As shown in Figure 2.4, after receiving the SAVE messages, router C builds a forwarding table such that the IP address range 128.250.110.* is only expected on link 1 and IP address range 128.250.128.* is only expected on link 2.
**Router C’s forwarding table**

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>128.250.110.*</td>
<td>1</td>
</tr>
<tr>
<td>128.250.128.*</td>
<td>2</td>
</tr>
<tr>
<td>____________________</td>
<td>_____</td>
</tr>
</tbody>
</table>

**Figure 2.4 Updation of SAVE message**

SAVE is a protocol that enables the router to filter packets with spoofed source addresses using incoming tables. It shares the same idea with ingress filtering and RPF that the source address space on each link of the router is stable and foreseen. Any packet that violates the expected source address space will be regarded as forged and will be filtered.

SAVE outperforms ingress filtering and RPF in that it overcomes the asymmetries of Internet routing by updating the incoming tables on each router periodically. However, SAVE needs to change the routing protocol, which will take a long time to accomplish. Moreover, as SAVE filters the spoofed packets to protect other entities, it does not provide direct implementation incentives. If SAVE is not universally deployed, attackers can always spoof the IP addresses within networks that do not implement SAVE. Moreover, even if SAVE were universally deployed, attackers could still launch DDoS attacks using nonspoofed source addresses.
2.7 ATTACK DETECTION

Attack detection is the next step in defending against network attacks after attack prevention. A critical measure of performance for any detection scheme is an exposure of the proportion of actual attacks can be detected. Attack detection for DoS attacks is different from general intrusion detection. The exposure of intrusions such as user to root and remote to local attacks, the attacker can hide the attack by changing the system log or deleting any file created by the attack.

Thus these attacks are difficult to detect. However, DoS attacks can be easily detected since the target’s services will be degraded, for example, with a high packet drop rate. Second, false positives are a serious concern for DoS attack detection. Since the intensity of DoS attacks does not depend on the exploitation of software bugs or protocol vulnerabilities, it only depends on the volume of attack traffic. Consequently, DoS attack packets do not need to be malformed, such as an invalid fragmentation field or a malicious packet payload, to be effective. As a result, the DoS attack traffic will look very similar to legitimate traffic. This means that any detection scheme has a high risk of mistaking legitimate traffic as attack traffic, which is called a false positive.

In general there are two measures for DoS attack detection. The first is detection time and the second is false positive rate. A good detection technique should have a short detection time and low false positive rate. Generally there are two groups of DoS attack detection techniques. The first group is called as 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.
Generally network based attacks are observed by the flow rate imbalance between the source and the victim, creation of flow rate imbalance between the source and the victim, attack traffic at the target is highly correlated with abnormal traffic behavior at the attack sources. The above said data flow variations are used to detect the network attack detection methods. In continuation the following section discuss few of specified detection techniques.

2.7.1 MULtilevel Tree for Online Packet Statistics (MULTOPS)

MULTOPS is a denial of service attack detection technique proposed by Gil and Poletto (2001) by monitoring the packet rate in both the up and down links. In this attack detection techniques it is assumes that packet rates between two hosts are proportional during normal operation. A significant, disproportional difference between the packet rate going to and from a host or subnet is a strong indication of a DoS attack.

The limitation of MULTOPS is that it uses a dynamic tree structure for monitoring packet rates for each IP address. This tree structure can itself become the target of a memory exhausting attack. An alternative approach for MULTOPS called as Tabulated Online Packet Statistics (TOPS) proposed by Abdelsayed et al (2003) provides an efficient method for detecting packet flow unbalances based on a hashing scheme that uses a small set of field length lookup tables. This approach avoids the risk of memory exhausting attacks.

2.7.2 Detection of SYN Flood Attack

Wang et al (2002) proposed a simple and robust mechanism for detecting SYN flooding attacks. Instead of monitoring the ongoing traffic at
the front end (like firewall or proxy) or a victim server itself, this method detect the SYN flooding attacks at leaf routers that connect end hosts to the Internet. The simplicity of this detection mechanism lies in its statelessness and low computation overhead, which make the detection mechanism itself immune to flooding attacks. This detection mechanism is based on the protocol behavior of TCP, SYN–FIN (RST) pairs and comparing the ratio of SYN packets to FIN and RST packets to detect the SYN flooding attack.

This detection scheme is based on the fact that a normal TCP connection starts with a SYN packet and ends with a FIN or RST packet. When the SYN flood starts, there will be more SYN packets than FIN and RST packets. Thus more numbers of SYN packets leads to SYN flooding disturbing legitimate user access of network resources. Another attack detection method is proposed by Blažek et al (2001) where a variety of parameters, such as TCP and UDP traffic volume, were used.

In data network data packets flow rate is normally normal but it may abnormal during the exposure of DoS attack that is attack flows have different statistical features compared with normal flows. Based on this concept Cheng et al (2002) proposed a network attack detection method. In this approach, the number of packet arrivals in a fixed interval is used as the signal. In the power spectral density of the signal, a normal TCP flow will exhibit strong periodicity around its round-trip time in both flow directions, whereas an attack flow usually does not. In continuation of the above traffic based attack detection Kulkarni et al (2002) proposed a Kolmogorov complexity based detection algorithm to identify attack traffic.

Since in data network a strong correlation exist between traffic behavior at the target and traffic behavior at the attack source. Keep this behavior in mind Cabrera et al (2001) have proposed a scheme to proactively detect DDoS attacks using time series analysis. There are three steps to this
scheme. The first step is to extract the key variables from the target. The second step is to use statistical tools (e.g., Autoregressive Model) to find the variables from the potential attackers that are highly related to the key variable. The third step is to build a normal profile using the found variables from the potential attackers.

A new DoS attack detection scheme using source IP address monitoring was proposed by Peng et al (2004). 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 prebuilt 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) presented by Brodsky and Darkhovsky (1993). An abrupt change of the proportion of new source IP addresses is a strong indication of a DoS attack. More importantly, this method can improve the detection accuracy by also monitoring the traffic rate per IP address.

2.8 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. This design decision has given the Internet enormous efficiency and scalability, though at the cost of traceability and network security in terms of DoS attacks.
In order to address this limitation, many schemes based on enhanced router functions or modification of the current protocols has been proposed to support IP traceability.

Generally, active IP traceback schemes can locate attack paths reliably and quickly. However, the common shortcoming for all active IP traceback schemes is that substantial control is needed to coordinate all participating routers. IP traceback schemes are method used to trace the attack sources based on the reaction of attack traffic.

One of the traceback techniques called as Backscatter traceback proposed by Gemberling et al (2001) and Morrow and Gemberling (2001). It is a traceback scheme implemented to identify the source of the attacker because network attacker generally use invalid spoofed source IP addresses.

Burch and Cheswick (2000) proposed a link-testing traceback technique. It infers the attack path by flooding all links with large bursts of traffic and observing how this perturbs the attack traffic. 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.

Stone (1999) proposed overlay network (It is a new physical or logical connection of a set of nodes on top of the existing network.) architecture to overcome this limitation. During DoS attacks, attack traffic (traffic to the target) is rerouted to the overlay network which is called CenterTrack. The CenterTrack is normally equipped with routers configured for tracking. Thus, the attack packets can be easily tracked, hop-by-hop, through the overlay network, from the routers close to the target to the attack entry point of the ISP.
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 (2000) proposed to traceback the IP source by probabilistic packet marking (PPM). 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.

The limitations of this method are requirement of specific field in the IP header but no specific field has been reserved for tracking purposes in the current Internet protocol IP v.4 although IP v.6 recommends by Deering and Hinden (1998) is expected to have such a field. Consequently, encoding schemes are needed to squeeze the path information into rarely used fields, such as the 16-bit identification field in the IP header.

Song and Perrig (2001) 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. More details about PPM can be found in Savage et al (2000). In Dean et al (2002) another coding scheme using an algebraic approach to embed path information was proposed to reduce the number of packets needed to reconstruct the attack path.

Bellovin (2000) proposed a similar approach called the ICMP “traceback” scheme. In this scheme, when a router receives a packet to a destination d, the router generates an ICMP traceback message, called an iTrace packet, with low probability. The iTrace packet contains the address of
the router, and is sent to the destination d. For a significant traffic flow, the
destination can gradually reconstruct the route that was taken by the packets
in the flow. The iTrace packets are generated with a very low probability by
routers to reduce the additional traffic, which undermines the effectiveness of
the scheme. To prevent attackers from spoofing the ICMP packets, an
authentication field is used in the iTrace packet. This scheme was later

The probabilistic packet marking schemes, the marking field can be
overwritten, and all the routers use the same marking probability, with the
result that the further the router the less possible it is to receive a marked
packet from that router. To overcome this problem, a scheme called Adjusted
Probabilistic Packet Marking was proposed in Peng et al (2002a). Under this
scheme, each router adjusts the marking probability according to its distance
to the target so that the target can receive the marked packets from all
marking routers with the same probability.

One crucial assumption for all probabilistic approaches is that a
significant amount of attack traffic transmits across the attack path. However,
during a highly distributed denial of service attack such as reflector attacks
Paxson (2001), the attack traffic comes from a large number of links. Hence
the number of attack packets is low on each independent link, where attack
packets come from only one attack source. Therefore, these probabilistic
approaches will fail to traceback the attack sources due to insufficient attack
traffic on independent links.

One of the drawbacks of the probabilistic approaches is the wrong
identification of the attack path when the attack traffic is very scarce on each
independent link during a highly distributed denial of service attack. And also
this approach fails to traceback the attack source, where the attack only
contains a small number of packets. Keep this in mind a better traceback
approach is developed by Snoeren et al (2001) called hash-based IP traceback, to trace individual packets and also this method not affected by traffic volume and is able to traceback even one single packet. In this proposal, routers keep records of every packet passing through the router.

IP Traceback with Deterministic Packet Marking (DPM) is a new approach developed by Belenky and Ansari (2003) for IP traceback which is scalable and simple to implement, and introduces no bandwidth and practically no processing overhead. It is backward compatible with equipment which does not implement it. The approach is capable of tracing back attacks, which are composed of just a few packets. In addition, a service provider can implement this scheme without revealing its internal network topology.

On Deterministic Packet Marking is an IP Traceback based technique proposed Belenky and Ansari (2006) by marking all packets at ingress interfaces.

DPM is scalable, simple to implement, and introduces no bandwidth and practically no processing overhead on the network equipment. It is capable of tracing thousands of simultaneous attackers during a DDoS attack. For sufficient deployment on the Internet, DPM is capable of tracing back to the slaves responsible for DDoS attacks that involve reflectors. In DPM, most of the processing required for traceback is done at the victim.

The traceback process can be performed post-mortem allowing for tracing the attacks that may not have been noticed initially, or the attacks which would deny service to the victim so that traceback is impossible in real time. The involvement of the Internet Service Providers (ISPs) is very limited, and changes to the infrastructure and operation required to deploy DPM are minimal. DPM is capable of performing the traceback without revealing
topology of the providers’ network, which is a desirable quality of a traceback
method.

Flexible Deterministic Packet Marking (FDPM) provides a defense
system with the ability to find out the real sources of attacking packets that
traverse through the network developed by Xiang et al (2009).

FDPM provides innovative features to trace the source of IP
packets and can obtain better tracing capability than others. In particular,
FDPM adopts a flexible mark length strategy to make it compatible to
different network environments; it also adaptively changes its marking rate
according to the load of the participating router by a flexible flow-based
marking scheme.

Earlier discussed path identification (Pi) DDoS defense scheme is a
deterministic packet marking scheme that allows a DDoS victim to filter out
attack packets on a per packet basis with high accuracy after only a few attack
packets are receive. Enhancement of the idea called the StackPi marking, a
new packet marking scheme based on Pi, and new filtering mechanisms.

The StackPi marking scheme consists of two new marking methods
that substantially improve Pi's incremental deployment performance: Stack-
based marking and write-ahead marking. This scheme almost completely
eliminates the effect of a few legacy routers on a path, and performs 2-4 times
better than the original Pi scheme in a sparse deployment of Pi-enabled

A Divide-and-Conquer Strategy for Thwarting Distributed Denial-
of-Service Attacks is an attack mitigation scheme that adopts a divide-and-
conquer strategy. Attack diagnosis (AD) combines the concepts of pushback
and packet marking, and its architecture is in line with the ideal DDoS attack
countermeasure paradigm - attack detection is performed near the victim host and packet filtering is executed close to the attack sources.

AD is a reactive defense mechanism that is activated by a victim host after an attack is detected. By instructing its upstream routers to mark packets deterministically, the victim can trace back one attack source and command an AD enabled router close to the source to filter the attack packets. This process isolates one attacker and throttles it, which is repeated until the attack is mitigated proposed by Chen et al (2007).

An ant-based traceback approach is proposed by Gu Hsin Lai et al (2008) to identify the DoS attack origin. Instead of creating a new type or function or processing a high volume of fine-grained data used by previous research, the proposed traceback approach uses flow level information to identify the origin of a DoS attack.

Another traceback method for detection of DDoS attacks is based on entropy variations between normal and DDoS attack traffic, which is fundamentally different from commonly used packet marking techniques developed by Shui Yu et al (2011).

Hirak Jyoti Kashyap and Bhattacharyya (2012) focuses on finding the most relevant and smallest possible subset of features for DDoS attack detection. A generic architecture of victim end DDoS defense mechanisms is presented and a near real time anomaly detection mechanism with high detection accuracy is introduced. The method is evaluated based on two real time and one benchmark dataset. Zhiyuan Tan (2012) proposed a Denial-of-Service (DoS) attack detection system where a multivariate correlation analysis technique based on Euclidean distance is applied for network traffic characterization and the principal of anomaly-based detection is employed for the attack recognition.
Xiaogang Wang et al (2012) proposed an efficient sequential watermark detection (ESWD) model for tracing network attack flows. The proposed ESWD model is a statistical analysis based sequential attack detector. This model is detect attack without any assumptions or limitations about the distribution of the timing of packets and proves their effectiveness despite traffic timing perturbations.

Qin Qia and Zhiwen Wang (2012) proposed an entropy based method used to analyze the attack detection alert and detect real network attacks in large scale network. The proposed method employ Shannon entropy to examine the distribution of the source IP address, destination IP address, source threat and destination threat and datagram length of IDS alerts; employ Renyi cross entropy to fuse the Shannon entropy vector to detect network attack.

Chen Yonghong (2013) proposed a detection algorithm for the detection of Distributed Denial of Service (DDoS) attacks based on network abnormal traffic.

In computer network the architecture of content-centric network (CCN) may suffer from interest flooding attacks. To mitigate this attack Kai Wang et al (2013) proposed a technique called denial of service against content source (DACS attack). It is a threshold-based detecting and mitigating (TDM) scheme.

The basic idea is to detect DACS attack on the basis of the frequency that pending interest table items in CCN routers expire (recording this frequency by introducing two counters with their corresponding thresholds and one indicator for counter mode) and to mitigate it by implementing the rate limiter in each router, the performance of TDM in terms of detection ability and effect on mitigating malicious traffic.
2.9 SUMMARY

From this comprehensive survey of different types of network attacks, it is understood that most of the network based attacks are exploits the weakness in the design and organization of the Internet. The growing dependence on the Internet makes the impact of these attacks increasingly painful for service providers, enterprises, hosting centers and government agencies etc., Responding to and defeating these attacks in a timely and effective manner is the primary challenge.

To response to these open challenges many research communities are tried to deploy many defense mechanisms. Each mechanism has certain features that make it more suitable to implement based on it requirement.

It is mandatory to fight these types of networks attacks is to increase the reliability of global network infrastructure. At the same time more reliable security mechanisms are still needed for the prevention of unauthorized access of network server.