Denial of Service Attacks in IP Networks

Golriz Cherazi, Susanne Koch
TDDC03 – Information Security, Linköping Universitet
[golch838, dorko489]@student.liu.se
Version 4.1, May 11th, 2005

Abstract

More and more people all over the world use the Internet. The World Wide Web encourages the possibility to exchange data and information. Applications for e-business and e-commerce are used by a variety of people. This ability to connect any computer to any other computer anywhere does not only constitute major progress, but also presents a threat to sensitive data.

1. Introduction

The basic aim of computer and information security is to fulfil the CIA criteria, which means systems have to provide confidentiality and integrity. Resources as well as data have to be always available. Denial-of-Service (DoS) attacks, which are threats against availability, have become increasingly popular in recent years. These attacks are major and widespread security-related challenges, which face computer networks and information systems. They are attacks against availability of resources and hinder the legitimate access to information. DoS attacks have increased in frequency, severity and sophistication in the last 15 years.

This report presents the reader a comprehensive overview of the area. It discusses the cause as well as the impact of DoS attacks and provides an outline of the background, attack tools as well as measurements.

2. Background

When the Internet was designed, developers did not take into account that it would get so popular. It was planned with functionality, not security in mind. That is one of the major reasons why threats such as DoS attacks are possible. If you are unfamiliar with the Internet architecture, you can refer to Appendix A, which includes a brief overview of the design of the Internet as well as its central protocols.

3. Attacks

3.1. State of the Art

»A DoS attack is characterized by an explicit attempt by an attacker to prevent legitimate users of a service from using the desired resources «[7].

Attackers carry out DoS attacks by making a resource inoperative. They occupy large amounts shared resource that other users have no or little resource left [1]. Thereby attackers do not damage data directly, but they intentionally compromise the availability of the resource.

3.1.1. How is an attack accomplished?

Attackers start DoS attacks by using tools to exploit vulnerabilities and then either obtain unauthorized access to an appropriate process or to use a process in an unauthorized way. The attacker completes the attack by using some method to destroy files, degrade processes, degrade storage capability, or cause a shutdown of a process or of the system [1].

Attacking a single host fits in the beginnings of DoS attacks, although it is still popular. Nowadays the trend goes over to DoS attacks that are aimed at World Wide Web services, file sharing services and the DNS. The consequences of such attacks are much more extensive, since a wide range of people and companies are affected. The entire network connected to the compromised host will suffer from lower or no performance.

So called Distributed Denial of Service (DDoS) attacks are just another flavour of DoS threats. The attacker uses a whole army of hosts (also called "Zombies") to generate large volume of synchronized DoS attacks. This is achieved by installing programs on vulnerable computers that can be remotely controlled by the intruders to carry out the attack.

In earlier times, the vulnerable computers had to be selected by hand, but nowadays there are self-propagating programs, which automatically find vulnerable computers, attack them and install their program on them. This process is repeated so that large attack networks can be built very quickly. This network-building phase creates lots of traffic and occasionally leads to congested networks as well. After enough computers are compromised, the intruders are ready for attacking the chosen victim.

Sometimes the malicious program is not remotely controlled by the intruders, but is designed to automatically execute the attack at a certain time, e.g. MyDoom worm in 2004, which was programmed to stop spreading after February 12th, 2004 [2].

3.1.2. Why can DoS attacks be accomplished?

The interdependence of Internet security is one cause. DDoS attacks are usually launched from systems that are subverted through security-related compromises. Regardless of how well secured the

victim system may be, its weakness to DDoS attacks depends on the state of security in the rest of the global Internet [5].

Another cause is that Internet resources are limited and can only be used by a limited number of users. Still another, not network related cause, is that many computers are poorly secured. They are not properly patched or have no anti-virus software or do not have updated software.

The architecture of the Internet is vulnerable for attacks as well. Based on the end-to-end communication paradigm, networks rely on intelligent end hosts. The network has little control about what is sent from one host to another. Each end host has to filter out traffic willing to accept. This way the amount of processing in intermediate networks is limited and packets can be forwarded quickly and at minimal cost. To support fast forwarding of packets, high bandwidth pathways exist in intermediate networks to enable large throughput. On the other side end networks invest in only as much bandwidth as needed. Therefore, Attackers can misuse the resources of intermediate networks for overflowing their victims [31].

3.1.3. Who is the target and what are the goals?

In general, Denial-of-Service attacks over the Internet can be directed against three types of targets: single user, host computer or network.

One aim of DoS attacks is to damage a victim, either for personal reasons, e.g. revenge or for popularity. From the technical point of view this is rather easy and attackers need little knowledge in order to be successful in attacking an unprotected machine.

DoS attacks can be carried out by hacker kids who want to show off their knowledge or gain respect in a hacker community. Moreover DoS attacks are accomplished for material gain and political reasons. Launching DoS attacks on a popular company during a high sales period has immense impact on the company. It reduces the profit, leads to bad reputation, annoyed customers and at last the loss of customers, since the service is not available. When the target is a Web server, DoS attacks (if successful) lead to very high costs, since entire companies, e.g. popular e-business companies, can be affected.

An even bigger threat is the unavailability of components of the core Internet Infrastructure like Domain Name Servers [8] or BGP routers. If name servers are successfully attacked, plenty of people (the whole Internet community) can not reach requested web-pages and requested data. [8] In October 21, 2002 Internet's root Domain Name System servers were victims of a massive DDoS attack. The consequences were not catastrophic because the DNS servers' information is stored redundantly in many distributed DNS servers. Additionally the fairly simple designed DDoS attack made it unproblematic for administrators to detect and block off the bad traffic in a few hours. [8]

If Border Gateway Protocols routers, which are used to exchange routing information for the Internet, are attacked, it could cause large portions of traffic to be misdirected at will. Although this attack is not trivial and includes several attack steps beside the DoS attack, experts mention this as a possible availability problem for web services. [4] A successful attack of that kind has not been observed yet.

3.1.4. How often are servers attacked?

John D. Howard's analysis of security incidents on the Internet [1] shows that between 1989 and 1995 the number of DoS attacks increased by 50 percent per year, whereas other attacks decreased. It seems plausible that the exponential growth rate of the Internet also dramatically increased the growth rate of attackers, knowledgeable enough to carry out the technically very easy DoS attacks. The largest single method used for denial-of-service attacks, as recorded in CERT records, was the use of mail spam to degrade storage capacity.

Records in the year 2000 indicated that attackers are increasingly developing tools to coordinate distributed attacks from many separate sites. CERT [9] registered 21756 network incidents in the year 2000, 82094 in the year 2002 and 137529 were reported in the year 2003. There is no precise number of DoS attacks available on CERT. But the overall numbers show a rather negative development.

This negative trend is confirmed by a team of researchers, made up of CAIDA's David Moore supported by Geoffrey M. Volker and Stephen Savage, both of UCSD's Computer Science and Engineering Department. They studied DoS attacks across the Internet for three one-week periods in 2001, centering on the number, duration, and focus of the attacks. They found more than 12,000 DoS attacks in the period studied. A small percentage of these attacks aimed to compromise devices crucial to the operation of the Internet, including routers and name servers. However 95 percent of the victims were attacked fewer than five times [10].

According to the IMlogic Threat Center, IM and P2P threats increased 271% in Q1 2005 over Q1 2004. As both legitimate and unapproved use of IM clients and P2P networking increases, new worms and viruses are gradually more using these mechanisms to spread. [15] Even though this number includes all kinds of incidents the authors assume that the number of DoS attacks is drastically increasing as well. Recent DoS attacks against large companies like *PlayOnline* (April 2005) and *Heinz Heise Verlag* (January/February 2005) endorse this assumption [16].

3.2. Attack Types

Depending on the resource that is compromised through the attack, there are three different concepts to classify DoS attacks. [16]

Attacks, which overload network resources, can be carried out through flooding. Attackers try to shut down the target by sending large amounts of incomplete or erroneous connection requests.

Attacks, which allocate bandwidth, are often DDoS attacks. The target system is troubled with requests from thousands of computers, which overload the target's network connection and leaves the target system temporarily unavailable. The attacker himself needs vast network connection or a network of zombies to bundle bandwidth.

Attacks, which engage individual web services, are based on programming errors or holes of target system. Malformed packets or exceptional high usage of target services successfully shut down servers.

A possible setting of DoS attacks could also be a mixed version of the concepts mentioned above. However the rule of thumb is to achieve maximum impact with minimum resources.

3.2.1. Flooding

The expression "flooding" is used to indicate that a target is overflowed by bogus data in great number until it collapses under the load. There are different kinds of flooding attacks, often based on special design decisions of TCP or UDP protocols.

3.2.1.1. TCP SYN Flooding and Land attack

A SYN flood attack exploits a vulnerability in the TCP connection management. When a client attempts a TCP connection to a server the client begins by sending a SYN message. The server answers with a SYN-ACK. Depending on the device, a limited amount of resource available to handle the requested session is reserved. Usually the client replies to a SYN-ACK with another ACK to finish the handshake with the server.

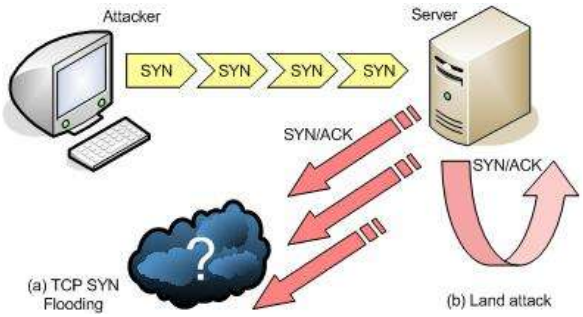

Figure 1: TCP SYN Flooding / Land attack

In a TCP SYN Flooding attack, the focal point is to exhaust the facility of the device to serve incoming requests for TCP connections by quickly consuming the limited resources of the device. This can be done by flooding the device with requests to which the client never responds with the third handshake. The attacker sends SYN messages to the server. These requests appear to be legitimate but in fact reference to a different client that is unable to respond to the SYN-ACK messages. This means that the final ACK message will never be sent to the server. By continuously flooding the device with requests that cannot be closed, the system can be slowed down or crashed.

The Land attack is a variant of the TCP-SYN attack. The difference is that the spoofed IP source address is the same as the destination address. The victim system is set up to acknowledge requests for its own services, creating a fatal loop back condition. This attack came first up in 1997 and was a successful attack against operating systems like FreeBSD, MacOS or Windows.

Another similar kind of attack occurs when attackers send IP fragments to victim hosts, but never enough to build a complete datagram, which consumes an ever-increasing amount of storage over time. This kind of attack is possible, because IP connection setups require some processing before coming to the conclusion to just throw the packet away.

It is possible to configure a firewall to drop all packets from a known attacker host, but through IP spoofing the attacker cannot be traced back.

3.2.1.2 Smurf and Fraggle

These two attacks function by sending ECHO requests to the broadcast address of access routers. The IP protocol includes broadcasting functionality that allows any traffic stream to be sent to all addresses within the network domain. When a router obtains such an ECHO packet, it automatically forwards the content of the packet to all addresses bound to the network.

In the smurf attack the packet sent to the router is an ICMP echo request. The fraggle attack is based on UDP. Under the principles of TCP/IP the receiving node must send an ECHO-REPLY to the source address. If the source address is the valid address of an external attacker, the router attached to the external network experiences a flood of ECHO-REPLY packets. The attacker has to send an incessant stream of ECHO requests to the access router to shut it down.

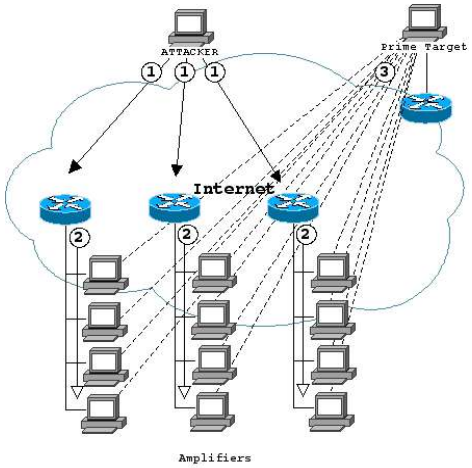

Figure 2: Smurf attack; (1) an attacker sends ECHO packets to broadcast addresses of access routers with a spoofed IP address; (2) routers broadcast the packets to their entire network; (3) the target device is flooded with ECHO-REPLY packets; picture taken from [35]

If this attack is combined with IP Spoofing, the attack can be launched against critical servers instead of routers. The remote IP address in the ECHO-packets will suffer from flooding of ECHO-REPLY messages.

3.2.2. Distributed Denial of Service Attacks

A DDos attack aggregates junk data traffic from thousands of computers into a formidable volume and floods and effectively blocks a certain victim. [3] The difference between a DDoS attack and a DoS attack is the amount of hosts to attack one source.

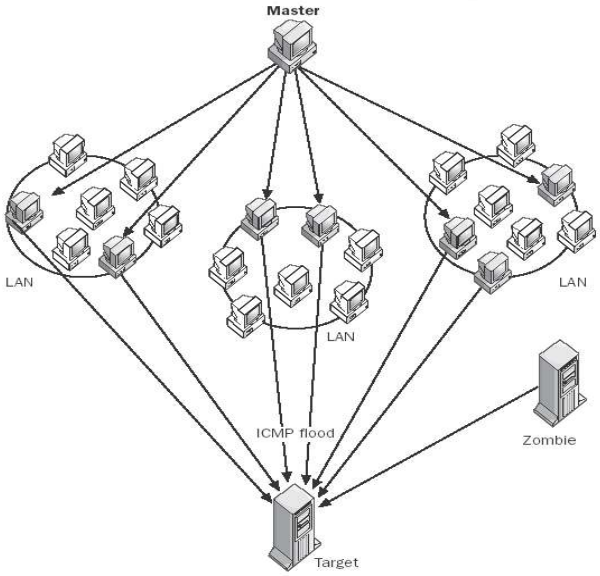

Figure 3: Example DDoS attack with master and zombie networks; picture taken from [36]

This kind of attack contains several steps to compromise a victim. First an attack network is built. The attacker accomplishes this by gaining access to poorly protected network computers. This process is usually made automatically through scanning of remote machines, looking for security holes. The discovered vulnerabilities are used to gain access to the machines and to infect them with a slave program. These hosts often peacefully wait for commands from a master program running on the attacker's machine. They can be used for recruiting new agents as well [6]. This process is automated. Once a large number of such slave programs ("zombies") are running, the master program contacts und instructs each of them to launch synchronized DoS attacks directed at the same target host.

This coordinated traffic [12] disables the services of the victim. The result is mostly disastrous, since the attacks come from so many directions at once.

The most recent DDoS attack was launched in April 2005 against the company "PlayOnline", which distributes Final Fantasy XI. The company mentions in a press release on their web site, that attackers could not been found yet through changing attack methods over time. The DDoS was successful and interrupted the company's game servers and held up customers to reach the game portal.

3.2.2.1. Reflector Attack

A reflector attack is one special type of DDoS attacks, consisting of three major components: the attacker, the amplifying subnet (i.e. reflectors), and the victim. [14]

An indirect attack is executed by using intermediary nodes (routers and various servers), better known as reflectors or innocent attack launchers. [13] Any host able to return packets can take the role of a reflector. Since the packets received by the victim are sent by reflectors, it is quite difficult to trace this kind of attack. Almost none of previous used traceback techniques can handle this attack. [14] Some major reflector attacks such as smurfing, SYN flooding, RST flooding, ICMP flooding and DNS reply flooding are summarized in [13].

DDoS Smurf attacks are typical reflector attacks. The attacker compromises a network of hosts called slaves first and it instructs each slave to send ICMP echo packets to a broadcast address of amplifying networks. Since the destination is a broadcast address of a local network, all hosts will respond to each of the packets (expected they are configured to respond to ICMP broadcast packets). This kind of attack can consume huge network and host resources with relatively few spoofed packets.

3.2.3. Worms

A worm is a self-replicating program, similar to viruses. A virus attaches itself to another executable program; a worm is self-contained and does not need to be part of another program to propagate itself. Although they are not designed for DoS attacks in first place (usually designed to exploit the file transmission capabilities found on many computers) they can cause network bandwidth saturation, scanning the network for hosts to infect.

At the time of writing, the most successful worms were MyDoom and Lovsan. In February 2004 the e-mail worm MyDoom forced the SCO group's website (www.sco.com) to shut down, while it barely distressed general web traffic. The worm represented almost 10 percent of worldwide e-mail volume. [18] In August 2003 the e-mail worm Lovsan/W32.Blaster attacked Microsoft's update-page. This attack remained unsuccessful through deactivation of the domain names by Microsoft. [19]

3.2.4. Malformed Packets

In January 26th 2005, CERT published new DoS vulnerabilities in Cisco's Internet Operating Systems (IOS) [11].

The implementation of »Multi Protocol Label Switching« (MPLS) in Cisco's IOS has a vulnerability that allows malformed MPLS packets to cause an affected device to reload. An unauthenticated attacker can send these malformed packets on a local network segment that is connected to a vulnerable device interface.

The way Cisco's IOS handles a series of specially formed IPv6 packets can cause an affected device to

reload, resulting in DoS. This vulnerability is exposed on both physical interfaces (i.e., hardware interfaces), and logical interfaces (i.e., software defined interfaces such as tunnels) that are configured for IPv6.

The third published vulnerability of Cisco's IOS is based on the »Border Gateway Protocol« (BGP). An IOS device that is enabled for BGP and set up with the *bgp log-neighbor-changes* option is vulnerable to a DoS attack via a malformed BGP packet.

Although these three vulnerabilities have different origins, in each case a remote attacker could cause an affected device to reload software. Repeated exploitation of these vulnerabilities can result in a denial-of-service condition since packets are not forwarded through the affected device while it is reloading.

Those recent flaws in Cisco's IOS were not the first of their kinds, there have been several in the past. An error in the HTTP service facility of Cisco's IOS was exploited in 2000. The HTTP service facility in Cisco's IOS gives remote administration capability using any web browser as client. This feature is commonly used to supervise remote routers and switches with a simple and user-friendly web interface. An error in the HTTP server allows attackers to access the HTTP service port to crash the device and force software reload. The service is enabled by default in almost all Cisco routers and switches running IOS versions 12.0 and 12.1. By sending an HTTP request with the URI http://switch-server/cgi-bin/view-source?/, the device crashes and reloads software, network connectivity is disrupted during this time. By continuously sending such HTTP requests, a DoS attack can be launched against the device and the entire network connected to it [34].

Since the beginning of computer networks vulnerabilities could be found and were used to attack hosts. Viruses, worms and malformed packets are used to compromise network devices. Malformed packets enclose both, sending invalid packets to network devices or sending packets unlike they are intended to be sent. The Cisco IOS flaws show recent examples of malformed packets. The most popular attacks in the past were Ping of Death, Teardrop and WinNuke. Although these attacks are not effective any more it is interesting to see, how these kinds of attacks work.

3.2.4.1. Ping of Death

The Ping of Death is a DoS attack caused by an attacker intentionally sending ICMP ECHO packets bigger than the 65.536 bytes permitted by the IP protocol.

A feature of TCP/IP is fragmentation, which enables single IP packets to be divided into smaller segments. In 1996, attackers started abusing this feature and divided packets into fragments that could be added up to more than 65.536 bytes. This attack affected at least 18 operating systems, reacting with a crash, freezing or reboot. [20]

3.2.4.2 Teardrop and Newtear

Teardrop attacks misuse the way TCP/IP requires a packet that is too large for the router to handle, to be divided into fragments. The fragment packet identifies an offset to the beginning of the first packet that allows the entire packet to be reassembled by the receiving device.

In the teardrop attack, the attacker puts a confusing offset value in on of the following fragments. (e.g. fragment one contains bytes 1-600 and fragment two starts already with byte 400) If the receiving operating system is not prepared for the situation, it can cause the system to crash or freeze. [20]

The Newtear attack is a modified version of the Teardrop attack and was launched about nine months after the first Teardrop attack. Attacker use in this case malformed UDP packets that are double the size of valid packets, which initiate an exception of the Windows TCP/IP stack. Users of a compromised host get an exception message "STOP 0x0000000A" or "STOP 0x00000019" and the device shuts down. [21]

3.2.4.3. WinNuke

WinNuke attacks manage to shut down Windows95, Windows98 and Windows NT by sending special out-of-band packets to Windows hosts. Newer versions of Windows are not affected by this attack. [20]

The attacker sends junk TCP packets also known as out-of-band packets with set URG-Flag to an arbitrary port. Port 139 (Netbios) is often used since it is part of the operating system. This attack causes the host to shut down. Users of the affected host often experience the "blue screen of death" if the device is compromised.

3.3. Attack Tools

Recent attacks demonstrate the power of DoS attacks. Attack tools are programmed in open source environments much quicker than countermeasures. The degree of automation in attack tools is growing significantly. The progress of automated attacks seems to involve four phases.

Scanning for potential victims starts a possible attack. Widespread scanning tools have become common since 1997 [17]. Nmap is one example of a popular scanning tool. It is a free open source scanning tool, uses raw IP packets to resolve hosts on the network, services, operating systems, types of packet filters/firewalls and a variety of other characteristics. Nmap runs on a variety of computers and both console and graphical versions are available. Nmap is free software, which is available with full source code under the terms of the GNU GPL. [22]

In the past, vulnerabilities were exploited after a widespread scan was complete. This would be the second step of an attack. Currently many tools include vulnerability exploitation as part of the scanning activity. This enables propagation on a high speed.

Broadcasting the attack is the third step in the attack phase. Before 2000 attack tools needed human interaction to start additional attack cycles. In recent

years attack tools are enabled to self-initiate new attack cycles. Code Red and Nimda are two examples. These attack tools self-propagate to a point of global saturation in less than 24 hours.

The last step in an attack phase is to launch the final DoS attack against the victim. Recent distributed attack tools are capable of launching DoS attacks more efficiently. Coordination functions use advantages of available communication protocols such as Instant Messaging (IM) or Internet Relay Chats (IRC).

Attack tools do not just use advanced techniques to launch attacks, they also include methods to hide their nature. This makes it much more difficult for security experts to analyze and understand new programs. Attack tools deal with dynamic behavior. They can vary their attack patterns and behaviors are based on random selection, decision paths or through direct intruder management.

A good example of a multifunctional attack tool is *Stacheldraht*, developed 1999 as a hybrid of *Trinoo* and *TFN*. It supports a variety of attack mechanisms as there are ICMP- SYN- and UDP flooding. As a special feature and a development to earlier tools it communicates to handlers and agents in an encrypted way. These agents and handlers execute automatically code and install updated versions from the attacker device. [25]

3.4. Countermeasures

Detecting and defending against DoS attacks at the target is difficult, since attacking traffic can hardly be differentiated from normal traffic. Usually victims are completely overloaded with malicious packets and they are not capable to take any protective measures. In some cases the victim can still access services on the Internet but with degraded performance, this way it is hard to distinguish between malicious and normal traffic. In other cases the connection can be completely interrupted, making it almost impossible to report to any network entity through the Internet. However there are plenty of countermeasures available to reduce risks and dampen the effects of attacks [22]. It is highly unlikely to find one tool that protects the network against every possible attack when used alone. Usually there is a set of countermeasures at different parts of the system continuously updated and extended to provide security.

Overprovision of bandwidth (resource multiplication) works well, if affordable, but it just moves the bottleneck to some other components and even servers with high amount of resources and high bandwidth connections are vulnerable to DDoS attacks.

3.4.1. Filtering

Filtering makes it difficult for attackers to launch attacks using spoofed IP addresses. Egress filtering or filtering of outbound traffic is often not considered as important as ingress filtering, which is done by creating inbound access rules to control what traffic is allowed to get in internal networks from the internet [27].

A firewall residing between a network and the internet should apply ingress filtering on external interfaces and drop all packets that have source addresses belonging to its internal network, since these packets have to be spoofed.

Egress filtering is applied on internal interfaces on outgoing packets if source addresses do not belong to the network address space. Firewalls only allow messages with legal address spaces to leave the network. This way, it is possible to ensure that no spoofed packets are leaving internal networks. This stops an attacker from using hosts within internal networks as DDoS agents.

Both mechanisms ensure that outgoing traffic may only have spoofed addresses belonging to the same network, which enables to easily trace back the attacker, because a valid and legitimately reachable source address has to be used. Since ingress and egress filtering is implemented in many networks, attacks that rely on IP spoofing can be stopped. But ingress and egress filtering can not provide protection against bandwidth based DDoS attacks [28] and flooding attacks that originate from valid IP addresses.

One possible defence mechanism against smurfing attacks is filtering packets with broadcast source addresses. This can be done at border routers. Another solution is configuring hosts to ignore ping packets or not to respond to broadcast ping packets. Thus, the drawback is the loss of the ICMP functionality.

Filtering is one practical prevention method applied at end-systems. Another approach is to trace back malicious traffic to its actual source or close to the source. This method helps to defend against ongoing attacks, e.g. by denying network access. These kinds of countermeasures are mostly applied at routing protocols by adding functionality to routers, so that attack routes can be reconstructed and routers can drop malicious packets that have been determined to have invalid IP addresses.

3.4.2. Traceback

One of these network based approaches is the IP traceback technique aiming to identify the attacker, in order that additional measures can be taken to stop attacks. It is a process going from one upstream router to the next to find out the hops taken by packets. The different traceback techniques try to solve approximate traceback problem. That means to find candidate attack paths, which contain the true attack path as a suffix. For example, the seven different routers in the figure beneath are a valid approximate solution, since the real attack path is included. Tracing spoofed packets to its true source is a difficult task and takes lots of effort, since many different ISPs are involved. For that reason co-operation between different networks is required. There are two major approaches to traceback packets. The hash-based traceback technique can only be used tracing single packets. The second technique could be employed to trace the attacker of a DDoS attack, using a large set of packets.

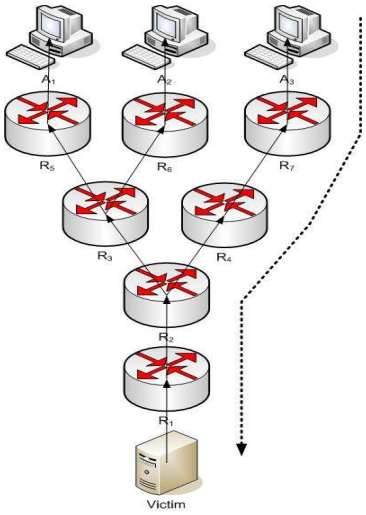

Figure 4: Network as seen from the victim of an attack. Routers are represented by R_i, and potential attackers by A_i. The dotted line represents a particular attack path between an attacker and the victim. Picture adopted from [29]

In the hash-based technique, routers store some information when packets traverse them, so that a victim can find out which routers a certain packet has passed. A disadvantage of this approach is that routers need to store additional information and after updating logs, routes can not be traced back anymore [7]. The latter technique uses a large number of methods including link testing, ICMP traceback and packet marking based mechanisms.

3.4.2.1. Link Testing

Starting with the router closest to the victim, links are tested from one upstream router to another until the attacker's router is determined. In order to accomplish this traceback, the attack has to remain active until the completion of the trace. One link testing scheme, called input debugging, is a mechanism implemented at routers allowing an operator to filter particular packets on some egress port and determine from which ingress port it arrived. For this mechanism to be successful, the victim has to recognize that it is being attacked and find out a common feature contained in all attack packets refered to as attack signature. This signature has to be communicated from one upstream router (mostly manually via administrative help) to another to reveal the upstream router where the traffic starts to come from.

A link testing technique that doesn´t require any support from network operators is developed by Burch and Cheswick [37]. It is a controlled flooding mechanism that tests links by flooding them with large bursts of traffic and observing how this disturbs traffic from the attacker. The disadvantage of this method is the additional network traffic. Besides, during a DDoS attack, when multiple upstream links are contributing to the attack, it is difficult to distinguish paths being exploited

3.4.2.2. Packet Marking

Packet marking is a technique that does not send additional packets, but rather modifies the IP ID field in each packet to carry information about addresses of routers that are traversed. All marking algorithms contain two components, the marking procedure executed by routers and a path reconstucting procedure implemented by the host. A router marks packets by attaching additional small pieces of information about its path, so the victim can, after observing marked packets, reconstruct the complete way back.

The simplest marking algorithm is to append each router´s address to the end of the packet as it travels through the network. But since the length of the path is not known from the beginning, it is not possible to ensure that enough space is available for additional information. This approach can cause problems with IPv6 because of different header format.

3.4.2.3. ICMP Traceback

Bellovin et al. illustrates another traceback mechanism using ICMP-traceback messages. A router sends ICMP traceback messages to the destination of every 20,000th packet, traversing it. The traceback packet contains the IP address of the router sending it, a TTL (or hop limit) of 255, and information about the adjacent routers along the path to the destination. During an attack, a sufficient number of these packets will reach the target for it to reconstruct the path taken by the malicious data [30].

Savage et al. describes a traceback mechanism for tracing anonymous, directly generated flooding attack packets toward their source using above described techniques. For reducing the overhead of packets, this approach does not store the whole path in each packet, but a router chooses with some probability one packet to write its address in. This is accomplished by reserving a 32 bit "node" field in the header of the packet for holding a router´s address. After receiving enough packets, the victim has at least one sample of every router in the attack path. From the relative number of samples per router (the distribution of samples) the attack path can be reconstructed. This can be done because routers are arranged serially and the probability that a packet will be marked by a router and not by any following downstram routers is a decreasing function of the distance to the victim. That means the nearer a router is placed to the victim, the more samples exist. So after receiving enough malicious packets, the entire path can be reconstructed [29].

There are further implementations of traceback techniques. For example, Z. Chen and M. Lee [14] propose a reflective algebraic marking scheme, which is based on the mathematical theory of linear algebra and coding theory. This method can be used for tracing ordinary as well as reflector attacks and it uses low network and router overheads. The marking scheme uses upstream routers map to speed up attack paths reconstruction. The biggest disadvantage of Chen and Lee's method is that marked information is not

authenticated. A compromised router could alter the information marked by its upstream routers and could lead the victim to reconstruct wrong paths.

One problem concerning lots of traceback techniques and DDoS is that sources of reflector attacks can not be traced back. After reflectors processed attack packets and sent reply packets to victims, the router looses any information of the attacker. In common, problems with traceback techniques can occur when the attacker compromises routers, which breed the packets to be marked with forged information. Besides, source addresses in attack packets cannot be trusted, since they are easy to forge. If all routers in the internet would implement source address filtering, tracing back packet routes would have been simplified.

3.4.3. Mitigating techniques

Firewalls or intrusion detection systems are used to detect attacks in progress, and notify upstream elements accordingly. Aggregate-based congestion control and pushback are seen as mitigating techniques, complimentary to above described approaches. E.g., a good map of networks with reliable historical traffic profiles can be used to determine sudden changes in traffic behaviour that signal attacks or to help determine how to allocate rate limits in pushback messages.

Pushback mechanism on top of enhanced routers is one possible mitigation technique. Routers are sequentially informed to filter traffic designed to specific addresses. The pushback mechanism is a network-based approach, implemented by adding functionality to routers and upstream routers to detect and drop packets that are probably malicious. This way, resources can be used for legitimate traffic. In case of normal traffic the standard TCP congestion control ensures fair use of available resources. DDoS attack traffic that does not obey to congestion control rules causes legitimate packet dropping. This mitigation approach uses a daemon process that saves information about dropped packages and draws conclusion about the congestion level of the network. The daemon tries to identify packets responsible for congestion with an aggregate-based congestion control (ACC) mechanism, based on packets' destination IP address. When the dropping rate grows above a certain level, the pushback daemon starts to drop specified packets while communicating this information to its upstream router. This approach assumes that the routers inside the network are of the same type [32].

The active network based DDoS defence is yet another method to mitigate DDoS attacks. It is based on possibilities of underlying active network infrastructures. There is a central management station at each domain, established to receive alerts. Alerts are generated by traffic rate monitoring applications running on each host. When the central management station receives an alert, it sends an active program to the next active router residing nearest to the victim. The active program filters traffic and replicates itself. These copies move towards the source of the attack. Nevertheless, the central management station is a single point of failure in this approach. Besides, this solution fails if an attacker succeeds in cutting of the path between central management station and active routers [26].

Eyrich et al. introduces a framework with distributed architecture avoiding a single point of failure. Furthermore, the framework relieves single hosts and networks. Mechanisms are located in the network itself, so that other nodes (hosts and routers) within the network can react even if certain nodes are enabled to react by an attack. This way, network routers and hosts are protected against resource starvation. The framework proactively provides a structure of main network paths in order that routers are able to block or reduce malicious traffic. Neighboured enhanced routers can quickly distribute information about an ongoing attack. Enhanced routers are also used to trace back actual sources of attacks. These routers are able to detect running attacks. If an attack is detected, warning pulses are distributed. Besides, methods are provided to quickly disburden an attacked system. This is done by nodes exchanging their states with pulses. In case of an attack pulses with mitigation rules are propagated toward the source of the attack by enhanced routers that are distributed over the Internet. Furthermore the framework provides methods for fast recovery of systems and keeps access restrictions as limited as possible without need of administrative configuration [26].

Secure Overlay Services (SOS) is a securing architecture for the communication between a pre-determined location and authorized users on top of the IP infrastructure. It is a proactive approach that filters and blocks all incoming packets to a target, whose source addresses are not approved. In this architecture only packets coming from a small number of nodes, called servlets, are assumed to be legitimate client traffic that can reach servlets through hash-based routing inside an overlay network. All other requests are filtered by the overlay. A secret servlet node computes keys for known hash functions based on the site's IP address. Each of these keys will identify a number of SOS nodes known as *beacons* for that site, that verifies the validity of the received information and forwards traffic in a random manner inside the SOS. In order to gain access to the overlay network, a client has to authenticate itself with one of the replicated secure overlay access points (SOAPs), that verifies that the source point has a legitimate communication for the target. SOAP routes packets to an appropriate beacon that eventually forwards them to secret servlets before they reach the target site. The probability of successful attacks against systems is reduced by a combination of secure overlay tunneling, routing via consistent hashing, filtering and introducing anonymity within the system, making it difficult for an attacker to target nodes along the path to a specific SOS-protected destination. Thus, SOS is a distributed system that offers a good prevention mechanism to a specified system at the cost of modifying client systems. Attackers can bypass the

mechanism if they can take control of a router that lies in the path between one of the approved overlay nodes and the target's filtering router [23].

J.Jiang and S. Papavassiliou present a new network attack diagnostic methodology, based on the characterization of dynamic statistical properties of normal network traffic. With mathematical approach, network anomalies provoked by attacks are detected and packets are marked as unacceptable and filtered if significant deviations from the expected behavior occur. The technique is called anomaly detection that identifies normal behavior patterns to recognize any unacceptable significant deviation from usual behavior, which can indicate an attack [33]. Next a method for accurate traffic prediction is applied. Dynamic thresholds and anomaly detection conditions are created to detect possible network attacks. The second technique is misuse detection, done by using signatures of known attack patterns. The latter technique fails to detect new kinds of attacks. Still, it was proved to be quite effective in detecting mail bombing and UDP flooding attacks.

Despite above described countermeasures, there are many DoS attacks that can hardly be filtered yet, since these attack packets are not different from legitimate packets. However, there are several means used to effectively defend a good number of DoS attacks. Beside those described prevention and detection methods, the first step to secure a device is general prevention. It is lively to keep the device up-to-date and apply patches as soon as they are available. Though, the past showed that patches alone are no sufficient measure for vulnerable systems, it helps to make the attackers' life more difficult.

4. Conclusions

This paper gave a comprehensive survey on DoS attacks. It described the nature, motives and kinds of DoS attacks as well as statistics on the topic. The authors listed a number of countermeasures against DoS attacks and specified their advantages and their drawbacks.

DoS attacks have increased by number and impact over the last years. Advanced attack tools are developed continuously in open source groups. On the other side countermeasures get more efficient and patches are developed as soon as vulnerabilities are discovered. But there are still lots of systems insufficient protected and a number of DoS attacks can not be prevented yet. For the authors, it is important to globally implement defense mechanisms to secure each single network. Hosts connected to the internet need to be patched and up-to-date. Network routers need to implement filtering methods, so that spoofed and malformed packets can be eliminated. Early recognition and removal of bad traffic is better than any traceback technique or mitigation mechanism at the victim's side. Although it is very unlikely to establish a globally defense mechanism, any employed prevention measure is a step in the right direction.

6. References

[1] J. Howard, »An Analysis of Security Incidents on the Internet 1989 – 1995«, *dissertation Carnegie Mellon University*, April 7th 1997, http://www.cert.org/research/JHThesis/Start.html

[2] M. Hypponen, K. Tocheva, S. Rautiainen, »F-Secure Virus Descriptions: Mydoom«, January 27th 2004, http://www.f-secure.com/v-descs/novarg.shtml

[3] J. Kurose, K. Ross, »Computer Networking – A top-down approach featuring the internet«, 3.edition, Addison Wesley, 2004

[4] S. Elias, »Is The Border Gateway Protocol Safe? «, *Whitepaper,* April 5th 2003, http://www.sans.org/rr/whitepapers/protocols/1046.php

[5] CERT Coordination Center, »Trends in Denial of Service Attack Technology«, *Whitepaper*, 2001, http://www.cert.org/archive/pdf/DoS_trends.pdf

[6] N. Weaver, »Warhol Worm: The Potential for Very Fast Internet Plagues «, *Proceedings of the 11th USENIX Security Symposium***,** February 2nd 2002, http://www.cs.berkeley.edu/~nweaver/warhol.html

[7] F. Lau, S. Rubin, M. Smith, L. Trajkovic, »Distributed denial of service attacks« , vol. 3, pp. 2275 –2280, *IEEE International Conference on Systems, Man and Cybernetics*, 2000

[8] Computerworld, »Net's Vulnerability Exposed«, October 28th 2002, http://www.computerworld.com/securitytopics/security/story/0,10801,75454,00.html

[9] CERT/CC, Statistics 1988-2003, http://www.cert.org/

[10] Sam Costello, »Study measures number of DoS attacks«, May 23rd 2001, a*rticle*, http://iwsun4.infoworld.com/articles/hn/xml/01/05/23/010523hndosrep.html

[11] CERT, »Multiple Denial-of-Service Vulnerabilities in Cisco IOS«, January 26th 2005, http://www.us-cert.gov/cas/techalerts/TA05-026A.html

[12] X. Geng, A. Whinston, »Defeating Distributed Denial of Service Attacks«, *Article »IT Pro«*, April/May 2000

[13] R. Chang, »Defending against Flooding-Based Distributed Denial-of-Service Attacks: A Tutorial«, October 2002, *IEEE Communications Magazine, Volume: 40 Issue: 10*, pp. 42 -51.

[14] Z. Chen, M. Lee, »An IP Traceback Technique against Denial-of-Service Attacks«, 2003,

http://www.acsac.org/2003/papers/100.pdf

[15] IMlogic Threat Center, »Q1 2005 IM Security Threat Report«, 2005, http://www.imlogic.com/pdf/Q105_IMThreatReport.pdf

[16] Wikipedia, »Denial of Service«, l*iving document*

[17] CERT Coordination Center, »Overview of Attack Trends«, w*hitepaper*, 2002 http://www.cert.org/archive/pdf/attack_trends.pdf

[18] CSO, »Safty in Numbers«, February 4th 2004, http://www.csoonline.com/metrics/viewmetric.cfm?id=659

[19] WinFuture, »MS schickt LovSan-Wurm geschickt ins Leere«, August 16th 2003, http://www.winfuture.de/news,10439.html

[20] Fyodor, »Exploit world! «, January 13th 2000, http://www.insecure.org/sploits_all.html

[21] Microsoft, »STOP 0x0000000A or 0x00000019 Due to Modified Teardrop Attack«, June 28th 2004, http://support.microsoft.com/?scid=kb%3Ben-us%3B179129&x=10&y=5

[22] Insecure.org, »Nmap Free Security Scanner«, http://www.insecure.org/nmap/index.html

[23]A. Keromytis, V. Misra and D. Rubenstein, »SoS: secure overlay services. In: Proceedings of the ACM SIGCOMM'02 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications«, ACM Press, New York (2002), pp. 61–72.

[24] J. Jiang, *Detecting Network Attacks in the Internet via Statistical Network Traffic Normality Prevention*, Journal of Network and Systems Management Vol. 12, No. 1, March 2004

[25] D. Dittrich, »The "Stacheldraht" distributed denial of service attack tool«, December 31th 1999, http://staff.washington.edu/dittrich/misc/stacheldraht.analysis

[26] M. Eyrich, A.Hess, G.Schäfer, L.Wartenberg, »A Proactive Distributed Denial of Service Protection Framework«, *International Infrastructure Survivability Workshop (IISW'04)*, November 15th 2004

[27] Egress Filtering v 0.2, Global Incident Analysis Center, SANS Institude, 2000

[28] P. Ferguson et. al. RFC 2267., »Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing«, *technical report*, The Internet Society, 1998.

[29] S. Savage, D. Wetherall, A. Carlin, T. Anderson., »Practical Network Support for IP Traceback«, *in ACM SIGCOMM*, pages 295–306, October 2000.

[30] S. Bellovin, M. Leech, T. Taylor, »ICMP Traceback Messages«, i*nternet draft, work in progress*, October 2001, http://www.ietf.org/

[31] J. Mirkovic, »D-WARD: Source-End Defense Against Distributed Denial-of-Service Attacks«, *dissertation*, University of California, Los Angeles, 2003

[32] J. Ioannidis, S. Bellovin, »Implementing Pushback: Router-Based Defense Against DDoS Attacks.« *In Proceedings of Network and Distributed System Security Symposium*, Catamaran Resort Hotel San Diego, California 6-8 February 2002

[33] W. Lee, D. Xiang, »Information-theoretic measures for anomaly detection«, *Proc. IEEE Symposium on Security and Privacy* (S&P 2001), pp. 130–143, 2001.

[34] I.Arce, »Vulnerability Report For Cisco IOS Web Administration DoS «, *mailing list,* muc.lists.bugtraq, October 25th 2000

[35] M. J. Martin, »Router Expert: Smurf/fraggle attack defense using SACLs«, *article*, October 17th 2002, http://searchnetworking.techtarget.com

[36] M. Tulloch, »Microsoft® Encyclopedia of Security«, Microsoft Press, 2003

[37] H. Burch, B. Cheswick, »Tracing anonymous packets to their approximate source«, *Proc. 2000 USENIX LISA Conf.*, Dec. 2000, pp. 319–327.

Appendix A

This Appendix provides an overview of protocols used in different network layers within the Internet network architecture as well as how they are used in order to initiate a communication channel between different computer systems. In order to understand why DoS attacks can be accomplished, it is helpful to know about the security lacks that the protocols have and the assumptions that were made when designing them. DoS attacks are possible, because they exploit these kinds of weaknesses.

The network model of the Internet architecture consists of different layers with different protocols. A protocol defines interfaces used for the communication between different machines.

Beneath the application layer, where protocols like HTTP, SMTP and DNS are situated, the transport layer is placed. Primarily at that layer DoS attacks take place. There are two different protocols negotiating between

network applications. These are TCP (Transmission Control Protocol) and UDP (User Datagram protocol).

The network layer is found beneath the transport layer. The IP protocol as well as the ICMP protocol is used for communication here. The latter one is build upon IP.

DNS (Domain Name Service) is the naming system of the Internet. It builds up a hierarchy to find e.g. the domain for each country on the top-level. Name-Servers map host-names to their correspondent IP-addresses, which are used, amongst others, by routers. A naming service is kind of a middleware that fills a gap between applications and the underlying network. Clients send queries to name servers and name servers respond with the requested information, which is a final address or a pointer to another server.

TCP, which is a connection oriented service, offers a reliable data transfer between the end systems (hosts). This is done by a handshaking protocol using acknowledgements and timeouts. Control information is exchanged by attaching a header to the message. The header includes for example sequence numbers for ensuring the right order of received messages. Additionally exchanged data includes the IP address´ of the host and the port numbers of the process´ in order to identify the recipient and the source of messages.

The communication of processes is built upon the Client-Server model, which is also used by HTTP, FTP, SMTP and DNS. All these protocols are built upon TCP. For establishing a TCP connection three steps are necessary. First the client sends a SYN packet to the server that it wants to connect with. When the SYN packet reaches the server host, the server allocates buffer space for the not fully established connection and answers with a SYN ACK packet. The client acknowledges receive of the SYN ACK with an ACK packet and the connection is established. This procedure is commonly referred to as a three-way handshake. A similar procedure is used for closing a connection.

UDP provides an unreliable, connectionless service. Because of less overload and no handshaking procedure, it offers a faster data exchange and is preferable used in many multimedia applications. The problem with filtering UDP packets is that it is very hard to anticipate where they are going, because of the lack of additional header properties.

The network layer packets are called datagrams. When the datagram is too big to be transported over a network channel, it is fragmented into smaller parts. On the receiver side the parts are defragmented to a full size datagram again.

ICMP is the Internet Control Message Protocol. Routers, which are responsible for forwarding packets, communicate on this layer. ICMP is used for error-reporting. For example the report “Destination network unreachable” has its origins in ICMP. This happens when an IP-router wasn’t able to find a path to a specified host.

For more details refer to [3].