1-1-2003

Centralized prevention of denial of service attacks

Michael R Sthultz
University of Nevada, Las Vegas

CENTRALIZED PREVENTION OF DENIAL OF SERVICE ATTACKS

by

Michael R. Sthultz

Bachelor of Arts
Claremont McKenna College, Claremont, California
1965

Bachelor of Science
University of California, Berkeley, California
1967

A thesis submitted in partial fulfillment
of the requirements for the

Master of Science Degree in Computer Science
Howard R. Hughes College of Engineering
Department of Computer Science

Graduate College
University of Nevada, Las Vegas
May 2003

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

UNLV
UNIVERSITY OF NEVADA LAS VEGAS

Thesis Approval

The Graduate College
University of Nevada, Las Vegas

April 18, 2003

The Thesis prepared by

Michael R. Sthultz

Entitled

Centralized Prevention of Denial of Service Attacks

is approved in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science

Ajay Kumar Dalta
Examination Committee Chair

Dean of the Graduate College

Examination Committee Member

John V. Harrison
Examination Committee Member

04/18/2003
Graduate College Faculty Representative

ABSTRACT

Centralized Prevention of Denial of Service Attacks

by

Michael R. Sthultz

Dr. Ajoy K. Datta, Examination Committee Chair
Professor of Computer Science
University of Nevada, Las Vegas

The world has come to depend on the Internet at an increasing rate for communication, e-commerce, and many other essential services. As such, the Internet has become an integral part of the workings of society at large. This has lead to an increased vulnerability to remotely controlled disruption of vital commercial and government operations – with obvious implications. This disruption can be caused by an attack on one or more specific networks which will deny service to legitimate users or an attack on the Internet itself by creating large amounts of spurious traffic (which will deny services to many or all networks). Individual organizations can take steps to protect themselves but this does not solve the problem of an Internet wide attack. This thesis focuses on an analysis of the different types of Denial of Service attacks and suggests an approach to prevent both categories by centralized detection and limitation of excessive packet flows.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The rapid growth of the Internet over the past ten years and its incorporation into almost all aspects of business and government has brought us to a critical point. As the usage of the Internet gets more widespread and as more vulnerabilities are discovered and exploited, the risk of a major disruption of essential services is cause for concern.

The most dangerous vulnerability is Denial of Service (DoS). This thesis discusses the different types of DoS with emphasis on bandwidth consumption since this is the most difficult for an individual organization to control and therefore has the broadest implications.

Chapter 2 presents an overview of the different types of DoS. Chapter 3 summarizes some of the DoS detection and prevention proposals that have appeared in the literature. Chapter 4 suggests a simplified and effective approach to control all types of bandwidth consumption Denial of Service attacks. Finally, Chapter 5 provides a summary and conclusions.

CHAPTER 2

TYPES OF DENIAL OF SERVICE ATTACKS

Denial of Service (DoS) can be achieved by an attack that renders a system totally inaccessible. It can also be realized by an attack that overloads system, network, or Internet resources to the extent that a legitimate user cannot easily access the system. As the use of the Internet has grown, the number and types of these attacks has grown with it. In order to better understand the different types of DoS and be able to discuss a mechanism that could be used to prevent it, we will categorize DoS as resource starvation, redirection, and bandwidth consumption. Bandwidth consumption will be further delineated as direct and indirect.

Resource Starvation

Resource starvation is defined as an attack on the resources of the system itself. The attack can cause the system to go into an infinite loop ("hang"), halt ("crash"), or just be kept artificially busy by an excessive use of CPU cycles.

In addition to the obvious physical intervention (e.g. power interruption), an attack can come from within the system (local resource starvation) through an employee, a contractor, identity theft, or via a compromised system. Local attacks usually involve process killing, a change in system configuration, filling up the process table, filling up the file system, or the planting of a logic bomb.

Remote resource starvation generally exploits an Operating System specific vulnerability. A number of these attacks have been described in the literature and have even been given names:

A Land attack sends a spoofed packet in which the source and destination IP addresses are the same. The source and destination ports are generally the same as well. This can cause the TCP/IP stack to get confused and the system crashes.

A Ping of Death sends an oversized ping packet. The TCP/IP stack cannot properly handle it and the system crashes.

SSPing sends a series of highly fragmented, oversized ICMP packets. TCP/IP has to keep track of additional information to reassemble the packets and the result is a memory overflow. The system either hangs or crashes.

Jolt2 sends a stream of packet fragments, none of which has an offset of zero. This means that none of the fragments looks like the first one in the packet. TCP/IP attempts to reassemble these bogus fragments and this consumes all processor capacity.

Teardrop sends packet fragments in which the fragment offsets are set to incorrect values. The packets appear to be overlapping and do not align properly. TCP/IP attempts to rebuild an impossible packet and the system crashes.

Winnuke sends random data to an open file sharing port (TCP port 139). The system attempts to process the data that it assumes to be in Server Message Block (SMB) format. It cannot properly interpret the block and it crashes.

Stream/Raped sends a stream of TCP ACK packets to a series of ports. The packets use random sequence numbers and random source IP addresses. This consumes all processor capacity.

Email Bombing sends a large number of very large emails. This disrupts the ability of the system to send or view email but it can also block other services.

There are many other resource starvation attacks that have been identified: CPU Hog, RPC Locator, Bubonic, Newtear, Bonk, Syndrop, etc. Unfortunately, there are also many others that will be developed.

Redirection

Redirection, as the name implies, redirects a request for service to a system other than the intended system. This can be done either by altering the routing process or by affecting the DNS resolution of Fully Qualified Domain Names (FQDN) to IP addresses.

Routers maintain their routing tables primarily through a dynamic process utilizing updates controlled by routing protocols. If these updates can be spoofed, the routing tables will contain incorrect information and packets will be misrouted. Another technique is to use source routing whereby the source system generating the packet specifies the route the packet will take as it traverses the network. All responses to source-routed packets will inverse the route of the originating packet. An intermediate system can thus be set up to intercept the response packets and deny service.

DNS redirection is done through DNS Cache Poisoning. A remote DNS server is used to put incorrect DNS records in the cache of a victim DNS server. A query to this server would then return an incorrect IP address for a particular FQDN. The poisoning could be accomplished either through a query from the remote DNS server (with an imbedded query response) or through a response to a query forced from the victim DNS server.

Direct Bandwidth Consumption

A direct bandwidth consumption attack is directed at the bandwidth serving an individual system. It is possible to initiate a local form of a bandwidth consumption attack by generating a large amount of spurious outbound traffic but this is the exception to the rule. For the most part, a direct bandwidth consumption attack is initiated from one or more remote systems.

An attack is often initiated from more than one remote system. This is considered a Distributed Denial of Service (DDoS) attack. The attacker compromises a large number of systems that are referred to as "zombies". These zombies can be located anywhere in the world and hundreds or even thousands of them are used in a specific attack. In the past, these were often systems at universities, companies or ISPs. With the growing popularity of "always on" systems utilizing DSL or cable-modem, these systems are being frequently used as zombies. The use of this technology by relatively unsophisticated home users makes it particularly vulnerable. In addition to zombies, attackers will often use one or more compromised systems as master systems. These masters are used to trigger all of the zombies to simultaneously conduct a DoS attack on the victim system. The use of masters and zombies has the added advantage of distancing the attacker from the victim and making tracing of the actual source of the attack more difficult.

A relatively new type of DoS is a Distributed Reflection Denial of Service (DRDos) which has been described as a next-generation DDoS attack [10]. This involves SYN flooding Internet routers and high bandwidth servers with TCP connection-requesting SYN packets to legitimate service ports. These packets are sent with a spoofed IP source

address of the victim system. The victim system then gets flooded with the SYN/ACK response packets. This type of attack is particularly dangerous since it cannot be blocked by the victim (it uses legitimate service ports), the intermediary routers/servers have a high bandwidth capability, and the attack is automatically multiplied since the unanswered SYN/ACKS will be repeated several times.

Direct bandwidth consumption attacks can be connection oriented (where actual TCP connections are established with the attacking system) but these are not commonly used because of the ease of identifying the source IP address and the relative ease of applying filtering to stop the attack. There are a number of known types of commonly used connectionless direct bandwidth consumption attacks:

A SYN flood can be a form of resource starvation or, if amplified through DDoS, a form of bandwidth consumption. In this attack, a TCP SYN packet is sent with a spoofed source IP address of an inactive system. The victim system sends a SYN/ACK and waits for an ACK response. The connection queue fills up with half open connections and no new connections can be initiated by legitimate users.

A Smurf attack is used with DDoS. Multiple echo request packets are sent from zombies. These echo request packets are sent as a directed broadcast to one or more intermediary networks with the source IP address spoofed as being from the victim system. All active systems on the intermediary network will send echo replies. The victim system will be overwhelmed with these echo replies and will be unable to respond to legitimate users.

A Fraggle attack is very similar to a Smurf attack but it uses UDP instead of ICMP. The zombies are directed to send UDP packets to an IP broadcast address with a

destination UDP port that will send a response, such as the UDP echo service (port 7). The victim system is then overwhelmed with the UDP responses.

There is a direct bandwidth consumption attack called Papa Smurf that is a combination of Smurf and Fraggle.

Echo-Chargen is an attack that sends UDP packets to the Chargen ports (port 19) of numerous hosts with the source IP address spoofed as being from the victim system. The source port is Echo (port 7). Chargen then responds with a pseudo random string of characters. Traffic will then reflect back and forth between the Chargen and Echo ports. Both bandwidth and processor capacity are affected.

Indirect Bandwidth Consumption

An indirect bandwidth consumption attack affects the bandwidth among multiple systems – or even the bandwidth of the entire Internet. The primary vehicle for this type of attack is through a self-propagating program that is often referred to as a "worm".

The term "worm" was first used in 1982 by Shoch and Hupp of Xerox PARC to describe a benign program that was designed to propagate through a local area network to perform system maintenance functions. The first well-known malicious worm that exploited security vulnerabilities in host software to infect several thousand systems was reported in 1988. The fast growth rate of this worm caused a noticeable disruption in Internet-wide communication [18].

Worms weren't much of a problem on the Internet until July, 2001 when "CodeRed" was released [18]. Code-Red exploited a buffer overflow vulnerability in Microsoft IIS web server via a HTTP request [27]. The worm infected almost 360,000 systems in less

than 14 hours with a peak infection rate of 2,000 systems per minute [18]. CodeRed had a limited bandwidth effect due, in part, to its being based on TCP.

Another well-known worm, Nimda, was released in September, 2001. Nimda infected over 2.2 million systems in a 24-hour period. This worm propagated via email and local network scanning for Microsoft IIS vulnerabilities (Unicode Web Traversal exploit). Nimda's bandwidth DoS effect were more localized due to its method of propagation [28].

These propagation speeds were considered fast until January 25, 2003 when the Sapphire/Slammer worm was released. Slammer doubled in size every 8.5 seconds and infected 90 percent of vulnerable systems worldwide within 10 minutes. This worm propagated via UDP and exploited a buffer overflow vulnerability in Microsoft's SQL Server or MSDE 2000 (Microsoft SQL Server Desktop Engine). The usage of UDP is significant in that the worm did not have to wait for a response (or timeout) to a TCP SYN packet. This means it was bandwidth-limited rather than latency-limited. It was able to achieve a scanning rate of over 55 million scans per second after approximately three minutes [19]. The effect of this was numerous network outages, which resulted in canceled airline flights, interference with elections, interruption of emergency services, and ATM failures. Fortunately Slammer did not have a malicious payload.

CHAPTER 3

DENIAL OF SERVICE DETECTION AND PREVENTION

There are a number of steps the user can take to defend against resource starvation and redirection types of DoS attacks. These include maintaining operating system patch levels, disabling any unused or unneeded network services, enabling quota systems, maintaining appropriate password policies, physical security, firewalls, intrusion detection systems, etc. [4]. Bandwidth consumption attacks are, however, out of the direct control of the user. There are a number of discussions in the literature regarding proposed mechanisms to detect and prevent bandwidth consumption DoS attacks. These can be categorized as either new protocols or traffic analysis.

New Protocols

The National Science Foundation has recently launched a $12 million research project to develop a "secure, decentralized Internet infrastructure that is resistant to failure and attack" [9]. The Infrastructure for Resilient Internet Systems (Iris) project will involve researchers at several prominent computer science laboratories including MIT and UC Berkeley. The project intends to use a distributed hash table technology that will support distributed applications on the Internet that will tolerate individual nodes that might be insecure or unreliable. This would only protect specific users using

selected applications and would not be effective against an indirect attack. It will also be several years before the design can be completed, tested, and implemented.

Harrison proposes a new overlay architecture called edge-to-edge traffic control [12]. It would involve spreading congestion across edge nodes to allow break up of congestion at interior nodes. This is achieved by establishing control loops (virtual links) that operate between domain edges to regulate the aggregate traffic between each edge pair. Control packets are exchanged between edge-to-edge nodes through the virtual link. This research does not appear to be complete as far as an Internet wide implementation. It is therefore not proven that it would be effective in preventing DoS attacks.

Aura proposes an authentication protocol based on client puzzles [2]. The concept here is to present the client with a puzzle (e.g. a brute-force reversal of a one-way hash function) as part of an authentication protocol to cause the client to commit resources prior to the server allocating its own resources. This would effectively prevent DoS by causing the client to utilize more resources than the server during the initial conversation.

Savage suggests a packet marking mechanism which would facilitate tracing attacks back to their source [24,25]. This has limited usefulness in preventing DoS since it is primarily an "after-the-fact" technique and attackers have many ways of hiding their true identity.

Moore proposes an active packet design in which packets carry actual programs rather than the standard headers [20,21]. The programs in the packets are then used to control more powerful node-resident services. This is not a new concept. Previous active packet designs have been used in applications such as reliable multicast and application directed routing. Moore suggests his active packet design (SNAP: Safe and Nimble

Active Packets) is more practical in terms of safety and efficiency. It uses a special-purpose interpreted language based on a virtual machine. This technique could be used to prevent DoS through a "surveyor" program that would visit a series of nodes and query the traffic load. The program carries a list of nodes to query and visits each in sequence. At each node, it queries the local traffic load and keeps a running sum. The sum could then be used to project bandwidth requirements. The entire program only uses 80 bytes of code (20 instructions).

Traffic Analysis Methods

There is a recent paper that focuses specifically on methods for detecting and containing worm infections [18]. It proposes identifying and blocking traffic from infected hosts (address blacklisting) and dropping packets that match a worm database of content signatures (content filtering). This method is dependent on a reaction time for detection, information propagation, and containment strategy activation. It correctly predicts (the paper predates the introduction of the Slammer worm) that future worms will include "aggressive worms that cannot be effectively contained" using this approach.

Gil proposes a new data structure for routers called MUlti-Level Tree for Online Packet Statistics (MULTOPS) [11]. The data structure is kept in a tree that dynamically changes its shape to reflect changes in packet rates. Bandwidth attacks are detected by searching for asymmetries between packet rates to and from different subnets. This approach could be effective in detecting UDP and ICMP based attacks but cannot reliably detect TCP based attacks (because TCP is an adaptive protocol).

A research project being funded by the Defense Advanced Research Projects Agency is called COSSACK (Coordinated Suppression of Simultaneous Attacks) [16]. This project will set up dedicated computers to continuously monitor the Internet, looking for signs of an attack. When an attack is detected, they will filter out the attack packets and send a notification message. The results of this project will not be available until at least 2004.

Martinez proposes a network security method, which is designed to protect the system being attacked [17]. The method is based on an Emergency Pulse Packet (EPP) System that consists of an intrusion detection system that uses analytical agents and an IP address communicator that sends information to a firewall to block the source of the attack. It consists of four components: an Intrusion Detection System that discovers and logs any unauthorized use of resources, an Analysis agent that scans the alert file and identifies event signatures, a Send Packet agent that sends source and destination addresses to the Deny agent, and a Deny agent that configures the firewall with a new rule. This approach relies on signatures of known attacks or classes of attacks for detection.

Ramanathan proposes a DoS detection mechanism based on wavelet signatures [23]. He calls this method WADeS (Wavelet based Attack Detection Signatures). He separates traffic into short-term and long-term flows with a certain portion classified as miss traffic (traffic for which the packet count does not exceed a certain threshold). He then computes a wavelet transform over the miss traffic and correlates it over long-term time-scales. Differences in wavelet signatures indicate a DoS attack.

Ye proposes an Anti-Flooding Flow-Control (AFFC) model for detecting and controlling flooding type DoS attacks [33]. This model includes traffic classification, dynamic buffer management, packet scheduling, and early-traffic-regulation (ETR). Her centralized bandwidth usage analysis is complicated by the diversity of the Internet, its growth over time including growth of different types of traffic, and the variance in usage over time of day and day of the week.

CHAPTER 4

PROPOSED APPROACH FOR PREVENTING DENIAL OF SERVICE ATTACKS

It is a well-known fact that the Internet is growing at an ever-increasing rate. Not only is it becoming a part of everyday life for the majority of people, it has become a vital factor in the functioning of society as a whole. Many companies and governmental bodies rely on the Internet for communication (IP telephony as well as email) and supply chain management. With the popularity of just-in-time inventory management, an interruption of the supply chain would cause manufacturers, distributors, and many retailers to be unable to continue operation in a matter of a few days. This could cascade into a loss of public transportation, emergency services, and health care. Not long thereafter, individuals would begin to run out of food and fuel. According to the co-chairman of the president's Critical Infrastructure Protection Board, the flow of electricity and natural gas is susceptible to cyberattack [31]. The potential effect of either a targeted or widespread attack on the Internet is frightening.

According to Symantec Corporation, a leading Internet security firm, the number of cyber attacks on corporate networks rose 20 percent in the second half of 2002 and the number of reported network vulnerabilities nearly doubled from a year earlier [7]. It no longer requires a high degree of skill to initiate a cyber attack. The operating system and other software developers publish a full description of attack vulnerabilities on their web sites. Easy to use software tools to automate cyber attacks are described in books [8] and

made available for download on innumerable web sites. More and more private individuals are using DSL and cable modem Internet connections. The combination of a high-bandwidth, "always-on" connection in use by an individual who is usually unaware of network security requirements makes these systems an easy target for compromise as zombies or other attack vehicles. The growing popularity of Windows XP for home systems, which has been designed to provide full raw socket support (as opposed to Windows 9x), makes packet crafting readily available.

It is a well-known fact that the initial design of the Internet was not done with security in mind. Some of the proposed protocols outlined in Chapter 3 [20,21,24,25] involve a change in the packet structure to enhance security. There are a number of active Internet design projects such as Internet2 [13], Next Generation Internet [30], and Large Scale Networking Coordinating Group [29]. A complete or partial redesign of Internet architecture could eliminate or mitigate the security deficiencies in the long term but it does nothing to solve the immediate DoS vulnerabilities

Other proposed protocols rely on an overlay architecture [2,12]. Overlay architectures depend upon the cooperation of all participants, including the attackers. This is an unrealistic assumption.

Many of the proposed traffic analysis methods discussed in Chapter 3 rely on content signatures of known attacks. Since new attacks are constantly being introduced, there is an unacceptable time delay in identifying and implementing new signatures.

Other methods [11,23,33] use relatively sophisticated analysis techniques. The dynamic nature of the Internet and the diversity and geographic dispersion of its users would require that any analysis and control technique be capable of being adjusted as

needed. This adjusting would be done by network personnel who are not necessarily Computer Scientists (e.g. they may not understand wavelet theory). A simpler approach would be more effective.

All bandwidth consumption attacks have one thing in common. They have to use a protocol at the Transport or Internet layer (referencing the TCP networking model) that is routable across the Internet. This is limited to three: TCP, UDP, and ICMP. If the entry flow rate of these packet types is appropriately limited at the edges of the Internet, bandwidth consumption attacks could be controlled. If this were coupled with source IP address verification (Ingress Filtering) at the same time, source IP address spoofing would be virtually eliminated (at least confined to known subnets) and attackers could be more easily identified.

A relatively immediate solution would be to mandate by law that all Internet Service Providers (ISPs) institute router controls that are already available. This would include Ingress Filtering and Rate Limiting.

Ingress Filtering is implemented by configuring the ISP edge router to only accept traffic with source addresses belonging to the customer network [5]. On a Cisco router, for example, this is done by applying an Access Control List on the incoming interface:

```
access-list 100 permit ip {customer network} {customer network mask} any
access-list 100 deny ip any any [log]
interface {ingress interface}
ip access-group 100 in
```

This process can be made more efficient by enabling Reverse Path Forwarding on the incoming interface:

ip verify unicast reverse-path

Reverse Path Forwarding requires that Cisco Express Forwarding be enabled on the router.

Rate Limiting would be used to limit the total amount of bandwidth that could be used by a particular packet type. Studies have shown (although confined to host based techniques) that rate limiting can be effective in restricting attack propagation [32]. Again, using a Cisco router as an example, limits could be placed using a Committed Access Rate (which requires Cisco Express Forwarding). For ICMP:

```
interface {ingress interface}
rate-limit input access-group 101 3000000 512000 786000 conform-action
        transmit exceed-action drop
access-list 101 permit icmp any any
```

where 3000000 is the average bps, 512000 is the normal burst (in bytes), and 786000 is the excess burst (in bytes).

To configure rate limiting for SYN packets:

```
interface {ingress interface}
rate-limit input access-group 103 45000000 100000 100000 conform-action
        transmit exceed-action drop
rate-limit input access-group 102 1000000 100000 100000 conform-action
        transmit exceed-action drop
access-list 102 permit tcp any any
access-list 103 permit tcp any any established
```

where average bps and bursts are defined as above. The 103 Access Control List will permit any established TCP traffic (packets with either the ACK or RST bit set) to be sent at a higher rate. Any TCP traffic with just the SYN bit set (connection attempts) will be limited to the lower rate. If this rate limit is being exceeded, obviously the host is not pausing to wait for acknowledgements and this is a possible attack.

For UDP traffic:

```
interface {ingress interface}
rate-limit input access-group 104 1000000 100000 100000 conform-action
        transmit exceed-action drop
access-list 104 permit udp any any
```

The actual values for average bps and bursts would have to be set very carefully to avoid dropping legitimate traffic and would depend on customer type, speed of the connection, etc.

International access points could be ordered blocked in the face of an attack to protect United States infrastructure.

This is only a partial solution but it would be an immediate one. ISPs have obviously been unwilling to do this voluntarily. Security experts have recommended (as input to the National Strategy to Secure Cyberspace being prepared by the cybersecurity panel headed by President Bush's computer security advisor) that more responsibility be put on Internet providers to screen data traffic for attacks. ISP objections were so strong that this recommendation was omitted from the draft report released in September 2002. One ISP executive claimed that their concern was "security tends to slow down data transmission" [6]. Other reasons might be the amount of effort involved and the potential

claim of legal liability in the event of a subsequent successful DoS attack. Many individuals (including a computer crime attorney at the U.S. Department of Justice) seem to think that ISPs will soon be held liable for DoS attacks whether or not there is a law enacted that gives them primary prevention responsibility [14,22].

Effort should begin to develop appliances that limit protocols by packet type count per unit of time. These would simplify the administration of rate limits and would be independent of a specific router type or brand. Limits could vary by client type and specific authorization. Ingress filtering would still be required.

These appliances could be made very fast by implementing most of the rate limiting process in hardware rather than software. They could use Application Specific Integrated Circuits (ASICs) similar to current multilayer switching technology. There are already similar devices on the market for use primarily on the customer side but they are too generalized (and too expensive) [1,3,32,].

It would be difficult to use a packet count approach at international access points and that may require the use of a more sophisticated technique similar to those described in Chapter 3. We may be restricted in the ability to prevent the introduction of an attack from outside the United States but ingress filtering and packet count limitations, as described above, would control its spread and effect within the United States. We could still retain the option of blocking international access points to protect U.S. infrastructure.

CHAPTER 5

CONCLUSIONS

This thesis has given an overview of the different types of Denial of Service attacks. It has demonstrated the potential danger represented by these vulnerabilities and has shown that none of the detection and prevention mechanisms that have been discussed in recent literature would give adequate protection. The recent demonstration of attack methods with extremely fast propagation times and wide-reaching negative effects demands that action be taken now.

The simplified approach for DoS prevention proposed in Chapter 4 has excellent potential but there are some possible limitations that would require further study:

- DDoS UDP attacks may still be successful. If a very large number of zombies were used, each with a relatively low rate of UDP packets, this could circumvent the edge node rate limiting controls. This circumstance might require some sort of centralized rate limiting.
- TCP ACK attacks would not be limited by using the standard router configuration commands. This could be addressed by the customized appliance proposed as a long-term solution.
- SMTP could still be a problem, in spite of its sender/receiver dialogue requirements, since it is possible to send multiple messages during a single TCP connection. This is of particular concern since it is known that some worms

- include an SMTP server capability. This could also be addressed by the customized appliance.

The current situation with the Internet can be compared to safety standards for the auto industry (although automobiles have never been nearly as critical to our infrastructure as the Internet has become). Although the safety vulnerabilities and the danger to the public were demonstrated in numerous ways, the auto industry declined to make any safety related design changes until mandated by Federal law. Since the various auto safety standards have been enacted, the public is much safer and the auto industry has suffered no ill effects.

The time for action is now. Although the Internet was designed in a much safer environment when the only damage its interruption could do was delay some research projects, it has grown into something much more dangerous. The implementation of the solutions proposed in Chapter 4 of this thesis could prevent this danger from becoming a reality.

BIBLIOGRAPHY

[1]Andress, Mandy. Denial of service: Fighting back. www.nwfusion.com/reviews/2002/0902rev.html. 2002

[2]Aura, Tuomas, Pekka Nikander, and Jussipekka. DOS-resistant Authentication with Client Puzzles. Citeseer. 2000

[3]Brindley. Adrian. Denial of Service Attacks and the Emergence of "Intrusion Prevention Systems". www.sans.org/rr/firewall/prevention.php. 2002

[4]CERT Coordination Center. Denial of Service Attacks. www.cert.org/tech_tips/denial_of_service.html. 2003

[5]Cisco Systems. Strategies to Protect Against Distributed Denial of Service (DDoS) Attacks. www.cisco.com/warp/public/707/newsflash.html. 2000

[6]CNN. Cybersecurity plan to offer tips, not rules. www.cnn.com/2002/TECH/internet/09/17/cybersecurity.ap/index.html. 2002

[7]CNN. Report: Serious Web attack threats loom. Cnn.technology.printthis.clickability.com/pt/cpt?. . . 2003

[8]Cole, Eric. Hackers Beware. Indianapolis IN: New Riders. 2002

[9]Collins, Kristen. MIT, Berkeley, ICSI, NYU, and Rice Launch the IRIS Project. Cambridge MA: MIT News. 2002

[10]Gibson, Steve. DRDoS Distributed Reflection Denial of Service. Grc.com/dos/drdos.htm. 2002

[11]Gil, Thomer M. MULTOPS: a data structure for denial-of-service attack detection. Ann Arbor MI: UMI Digital Dissertations. 2000

[12]Harrison, David and Shivkumar Kalyanaraman. Edge-To-Edge Traffic Control for the Internet. Citeseer. 2000

[13]Internet2. FAQs about Internet2. www.internet2.edu/about/faq.html. 2003

[14]Kiefer, Kimberly B. and Randy V. Sabett. Am I Liable?. www.intek.net/Secure/White/121302404am_I_liable.htm. 2003

[15]McClure, Stuart, Joel Scambray, George Kurtz. Hacking Exposed. Third Edition. Berkeley CA: Osborne/McGraw-Hill. 2001

[16]Mankin, Eric. COSSACK Rides to the Aid of Web Operators. www.usc.edu/isinews/stories/38.html. 2001

[17]Martinez. Network Security: A Theory for Securing Computer Networks Against Denial of Service Attacks. Ann Arbor MI: UMI Digital Dissertations. 2001

[18]Moore, David et al. Internet Quarantine: Requirements for Containing Self Propagating Code. Citeseer. 2003

[19]Moore, David et al. The Spread of the Sapphire/Slammer Worm. www.caida.org/outreach/papers/2003/sapphire/sapphire.html. 2003

[20]Moore, Jonathan T. Practical Active Packets. Ann Arbor MI: UMI Digital Dissertations. 2002

[21]Moore, Jonathan T. and Scott M. Nettles. Towards Practical Programmable Packets. Citeseer. 2001

[22]Narayanaswamy Ph.D., K. ISPs should emphasize security to avoid DoS attack liability. www.techrepublic.com/printerfriendly.jhtml?id=r00620020611 jdt01.htm&rcode=. 2002

[23]Ramanathan, Anu. WaDeS: A Tool for Distributed Denial of Service Attack Detection. Ann Arbor MI: UMI Digital Dissertations. 2002

[24]Savage, Stefan, David Wetherall, Anna Karlin, and Tom Anderson. Practical Network Support for IP Traceback. Citeseer. 2000

[25]Savage, Stefan R. Protocol design in an uncooperative Internet. Ann Arbor MI: UMI Digital Dissertations. 2002

[26]Skoudis, Ed. Counter Hack. Upper Saddle River NJ: Prentice Hall. 2002

[27]Symantec Corporation. CodeRed Worm. Securityresponse.symantec.com/avcenter/venc/data/codered.worm.html. 2002

[28]Symantec Corporation. Responding to the Nimda worm: Recommendations for addressing blended threats. Securityresponse.symantec.com/avcenter/reference/nimda.final.pdf. 2001

[29]United States Government. Introduction The Federal agencies of the Large Scale Networking Coordinating Group. www.hpcc.gov/iwg/lsn/lsn-workshop-12mar01/1.pdf. 2001

[30]United States Government. Next Generation Internet initiative. www.ngi.gov/white-house/background.html. 1996

[31]Verton, Dan. Energy: The First Domino in Critical Infrastructure. Computerworld. 2002

[32]Williamson, Matthew M. Throttling Viruses: Restricting propagation to defeat malicious mobile code. Stoke Gifford UK: Hewlett-Packard Company. 2002

[33]Ye, Baoquing. Network Denial-of-Service Classification, Detection, Protection. Ann Arbor MI: UMI Digital Dissertations. 2001

VITA

Graduate College
University of Nevada, Las Vegas

Michael R. Sthultz

Home Address:
3284 N. Serene Drive
Las Vegas, Nevada 89108

Degrees:
Bachelor of Arts
Claremont McKenna College, Claremont, California

Bachelor of Science
University of California, Berkeley, California

Thesis Title: Centralized Prevention of Denial of Service Attacks

Thesis Examination Committee:
Chairperson, Dr. Ajoy K. Datta, Ph. D.
Committee Member: Dr. Wolfgang W. Bein, Ph. D.
Committee Member: Dr. John Harrison, Ph. D.
Committee Member: Dr. Henry Selvaraj, Ph. D.