Design of an Autonomous Anti-DDoS (A2D2) Network

Angela Cearns and C. Edward Chow

Department of Computer Science
University of Colorado at Colorado Springs
1420 Austin Bluffs Parkway Colorado Springs, CO 80933-7150, USA
TEL: +1-719-262-3110 FAX: +1-719-262-3110
e-mail: {acearns,chow}@cs.uccs.edu

Abstract – The threat of DDoS attack are mainly directed at home and SOHO network that lacks the incentive, expertise, and financial means to defend themselves. This paper proposes an Autonomous Anti-DDoS Network Design (A2D2) that integrates and improves existing technologies. A2D2 enables SOHO networks to take control of their own defense within their own boundary. Testbed results show that A2D2 is effective in ensuring Quality of Service (QoS) during bandwidth consumption DDoS attack.

I. Introduction

Since early 2000 when a number of high profile sites such as eBay and Yahoo.com were halted by Distributed Denial of Service (DDoS) attacks [Dit00], the initial furor has subsided but the continual threat has ascended. The prevalence of DDoS attacks was verified by a recent study conducted by the University of California, San Diego (UCSD), that detected approximately 12,805 Denial of Service (DoS) attacks against more than 5,000 targets during a three-week period in mid-2001 [MVS01]. Even CERT, the authority that warns Internet users on security threats, fell victim to DDoS in May 2001 [ITW01].

The increase of DoS attacks can partly be attributed to the development of more sophisticated and “user-friendly” tools such as mstream and Stacheldraht [Dit99]. Such tools enable attackers to easily create distributed channels through which a massive DoS attack can be launched as illustrated in Figure **1**. The distributed nature of the DDoS attack has made it extremely difficult to trace and stop the attack. The overall attack impact is also exponentially amplified by the large number of attack agents around the globe.

I.A Mitigation Systems Against DoS and DDoS

To date, a myriad of commercial devices have been introduced to attempt to combat DoS and DDoS attacks such as Mazu TrafficMaster Enforcer and the Reactive FloodGuard [For01]. While these systems provide some automatic mitigation against DDoS, the cost ranges from a monthly charge that starts at $5,000 to a one-time product purchase price of $150,000. Administrators interested in assembling their anti-DDoS systems do not face cheaper alternatives. An average IDS such as the Cisco IDS, the Dragon IDS and ISS RealSecure costs $7,500 to $25,000 [Des02]. The cost quickly becomes prohibitive after adding a firewall and a few routers. Despite the financial investment, it is impossible for these devices to defend against all types of DDoS completely due to the changing nature of the attacks.

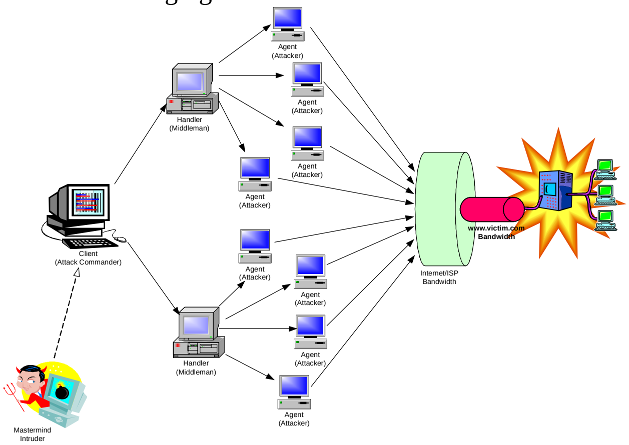

Figure 1 – A typical DDoS Architecture

According to the research conducted by UCSD, a predominant number of DDoS are targeted towards home networks and to smaller and medium-sized businesses [MVS01]. Since the commercial systems against DDoS are expensive and yet imperfect solutions, small networks may see their needs to guard against DDoS as a low priority and thus increase their chance of being victimized. Other home and medium-sized businesses may not have the resources, knowledge base, and financial means to implement the anti-DDoS commercial systems described above. Therefore, this paper explores an autonomous defense-architecture against DDoS that can be easily deployed in a small and medium sized network where administrators’ time is scarce and financial support is limited. Specifically, the Autonomous Anti-DDoS (A2D2) network proposed in this paper aims to maximize the quality of service of the victim network automatically during a DDoS bandwidth consumption attack.

II. DDoS Defense Related Research

In general, DDoS defense research can be roughly categorized into three areas: intrusion prevention, intrusion detection, and intrusion response. Intrusion prevention focuses on stopping attacks before attack packets reach the target victim. Intrusion detection explores the various techniques used to detect attack incidents as they occur. Intrusion response research investigates various techniques to handle an attack once the attack is discovered. In addition to these three research areas, intrusion tolerance, once a sub-field of intrusion response, is emerging as a critical research domain and is the focus of this paper. Intrusion tolerance responds to attacks by minimizing the attack impact. This section reviews key research in the intrusion tolerance domain.

II.A Intrusion Tolerance

Intrusion Tolerance research accepts the fact that it is impossible to prevent or stop DDoS completely. Instead of defeating DDoS, research in this category focuses on minimizing attack impact and maximizing the quality of its services. Many advances in intrusion tolerance are developed based research on quality of service (QoS).

II.A.1 Quality of Service (QoS)

Quality of Service (QoS) describes the assurance of the ability of a network to deliver predictable results and services for certain types of applications or traffic [Comp, ZOS00]. Among the most standard QoS techniques used to mitigate DDoS are rate-limiting and class-based queuing [Cis02, HMP+01]. These techniques are elucidated in Section II.A.1.i. Often, various QoS techniques are integrated to enable a system that demonstrates superior intrusion tolerance and some of these systems are explicated in Section II.A.1.ii.

II.A.1.i Intrusion Tolerant QoS Techniques

Queue management controls the length of packet queues by dropping or marking packets. One of the oldest and most widely applied queuing techniques is Class-based queuing (CBQ). CBQ or traffic shaping sets up different traffic queues for different types of packets and for packets of different Type Of Service (TOS). A certain amount of outbound bandwidth can then be assigned to each of the queues. For example, a Linux router can limit ICMP traffic to only 5% of the bandwidth while allowing multi-media traffic 80% of the available bandwidth. Class-based queuing has shown to maintain QoS during DDoS attack on clusters of web servers [KMW01, WO01]. The concept of CBQ is illustrated in Figure 2.

While queuing or traffic shaping determines the way in which data is sent and manages how the outbound link is utilized, the queuing discipline has no control over the inbound link and how fast packets arrive. Another QoS technique rate-limiting or traffic policing applies filters to limit the arrival rate of packets. For example, in response to a ping-flood DDoS attack, a system administrator can configure network routers to accept only 10 ICMP packets per second and discard the rest of the incoming ICMP packets.

Often, various QoS techniques are integrated to enable a system that demonstrates superior intrusion tolerance. To alleviate administrators' workload and to minimize mitigation response time during an attack, an autonomous system-approach is necessary.

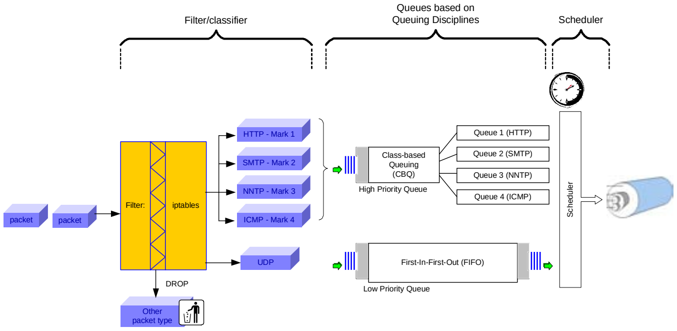

Figure 2 - Implementing QoS using CBQ

II.A.1.ii Intrusion Tolerant QoS Systems

Various autonomous architectures have been proposed that demonstrated intrusion tolerant during DDoS bandwidth consumption attacks. Some representative systems are the XenoService [YEA00], the pushback mechanisms proposed by Ioannidis and Bellovin [IB02], and the autonomic response architecture supported by The Defense Advanced Research Projects Agency (DARPA) [SDW+01].

The XenoService [YEA00] proposed an infrastructure of a distributed network of XenoService web hosts replicate a web site that is under attack. The goal is to acquire more network connectivity for the web site to absorb a packet flood. The pushback architecture is a promising mitigation technique where routers instruct their upstream routers to rate limit during attacks [IB02]. DARPA has supported research on sophisticated autonomic response systems based on the Cooperative Intrusion Traceback and Response Architecture (CITRA) and the Intruder Detection and Isolation Protocol (IDIP) [SDW+01]. IDIP is a special protocol for reporting intrusions and coordinating attack trace-back and response actions among network devices. CITRA refers to the architecture of network communities and network devices that use IDIP such as firewalls or routers. CITRA communities can cooperatively trace and block network intrusions as close to their source as possible [SDW+01].

While these autonomous systems propose promising architectures, they require adoption of a new protocol, expensive infrastructure investment or extensive cooperation of different entities on the macro level. In order to achieve dynamic replication, the Xeno infrastructure requires ISPs worldwide to install Xenoservers. The pushback techniques real value can only be realized when ISPs worldwide make agreements on how to honor pushback requests. Special communication protocols such as IDIP and the CITRA infrastructure are gaining acceptance but the standard has not been established. The specification of IDIP is not available to the public domain.

A small business owner does not have influence over the network design or the partnerships of his or her service provider. A small network also does not have the expertise and financial resources to support the global implementation of a certain infrastructure. Therefore, the current paper evaluates what strategies can be implemented on a micro level at every network. This paper aims to design an Autonomous Anti-DDoS (A2D2) network by integrating and improving existing methodologies that enable small and medium-sized networks to take control of their own defense within their own boundary.

III. The Proposed Autonomous Anti-DDoS Network (A2D2) Design

The A2D2 network is specifically designed to enhance quality of service during bandwidth consumption DDoS attack. The A2D2 design follows four main guiding principles:

- Affordable
- Manageable
- Configurable
- Portable

The target audience for the A2D2 network is home networks and small to medium sized companies. To ensure affordability, A2D2 will make use of open source and existing technologies wherever possible. In addition, the A2D2 network should be easily managed with minimum administrator intervention, can be quickly configured for network of various sizes and readily ported to mitigate attacks other than DDoS.

The design of the A2D2 network will center around four main and is illustrated in Figure 3:

1. Intrusion Prevention
   - Setup of a De-militarized Zone (DMZ)
   - Firewall policy
2. Intrusion Detection
   - Snort IDS – flood preprocessor add-on
3. Intrusion Tolerance – QoS
   - Multilevel Rate-Limiting
   - Class-Based Queuing (CBQ)
4. Autonomy System
   - Interface among the various components within the A2D2 DMZ

III.A Intrusion Prevention

III.A.1 Setup of a De-militarized Zone (DMZ)

The principal security policy applied is the separation of public services from the private network and the access control of the public services. Indeed, the design of A2D2 centers on the design of the anti-DDoS Demilitarized Zone (DMZ). A DMZ is a small network inserted as a "neutral zone" between a company's private network and the outside public network [Whatis]. An intruder penetrated the DMZ hosts' security can only access the web pages and other public information and no other company information would be exposed.

III.A.2 Firewall policy

The firewall implements a set of rules or chains based on the network security policy. A recommended approach is to set a deny policy where all traffic are denied into the DMZ. Based on the services started in the DMZ public servers, additional rules are applied to allow traffic to access the specific port

III.B Intrusion Detection

III.B.1 Snort Overview

Among all the well recognized and broadly deployed IDSes, snort is the only free, open source lightweight intrusion detection system and is selected to be the detection component of A2D2 [Sao02]. Snort detects attack mainly based on a signature recognition detection engine as well as a modular plugin architecture for more sophisticated behavior analysis.

III.B.2 A2D2 Snort Module Plugin – Flood Preprocessor

At present, snort has not included a logic that detects generic bandwidth consumption flooding launched against a network. DoS and DDoS detections are carried out by the base detection engine based on the rules defined. For example, two Snort rules are created to detect the word "skillz" and "ficken" in a packet's payload. These two words are used in communication messages sent between an attack agent and its handler. Attackers can easily change the payload content and new rules will need to be added.

To reduce management and maintenance hassle, A2D2 is required to detect generic flooding attack independent of specific DDoS tools. Unlike signature matching, flood detection needs to be designed as a preprocessor modular plugin. The flood preprocessor will perform an "x packets over y time" logic evaluation. Should x packets arrive within y seconds from the attack source, an attack alarm will be raised. Administrators or users can set an incoming packet rate threshold (x packet over y time) that deviates from their normal network traffic significantly. This flood threshold is set in the snort.conf file and provides a flexible configuration channel compatible with existing preprocessors of snort.

III.B.2.i Subnet Flood Detection

Nowadays, most bandwidth consumption DDoS attackers spoof the source IP addresses of the attack machines such as the situation illustrated in Figure 4. To counter DDoS IP Spoofing, A2D2 is designed to detect subnet flooding as well as individual host flooding. The three types of generic flooding that are being detected are:

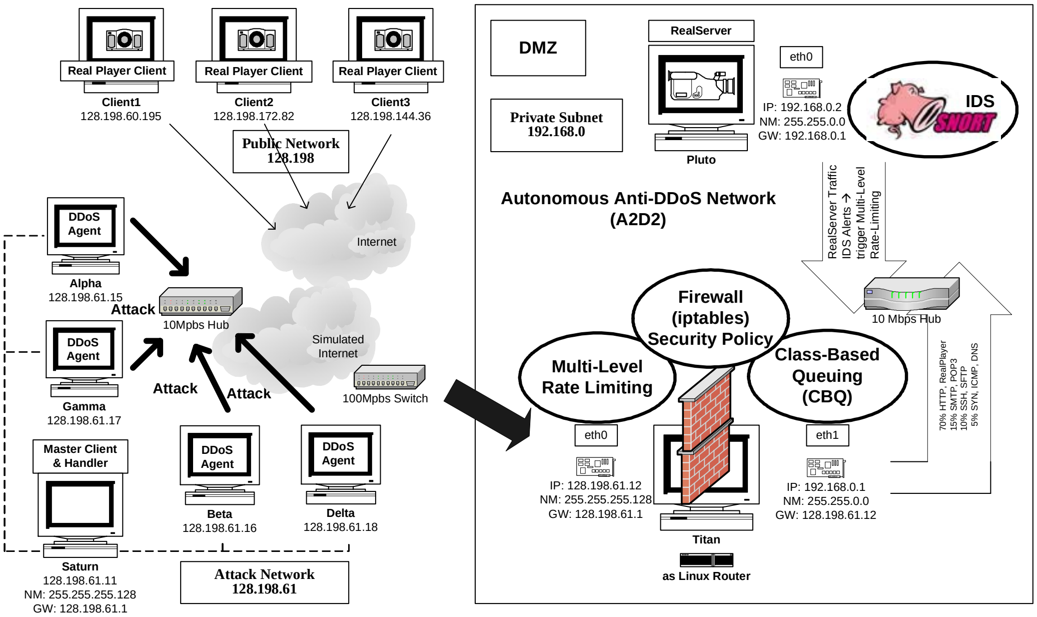

Figure 3 – A2D2 Implementation Test-bed

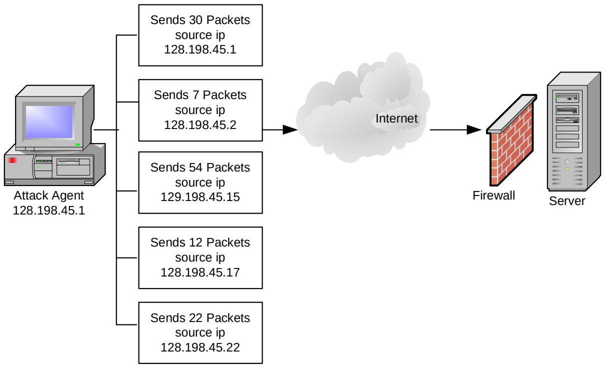

Figure 4 - DDoS IP Spoofing Each Attack Agent

- Individual attack host against individual victim host
- Subnet attack agents against individual victim host
- Subnet attack agents against victim subnet hosts

With current technology, it is still impossible to identify from which subnet a packet initiated. Therefore, certain design assumptions have been made regarding subnet flooding detection. For subnet flood detection, A2D2 will assume packets come from a /24 network based on the Classless Inter-Domain Routing (CIDR) addressing scheme [RL93]. A /24 network is equivalent to a traditional Class C network with 253 hosts. The /24 subnet flood is also based on the assumption that many networks are applying egress filtering to discard packets that do not initiated from their own subnets.

Considerations have been given to /22 and /16 subnet flood detection. There are 1021 hosts in a /22 network and 65,536 hosts in a /16 network respectively. These networks can legitimately generate a large amount of traffic. A /22 and /16 subnet flood detection adds extra reassurance but may also produce more false positives. Therefore, the A2D2 design will assume a /24 subnet detection.

III.B.3 A2D2 Snort Module Plugin Add-on – Flood IgnoreHosts Preprocessor

It is conceivable that an administrators or a valued customer may generate significantly more than the threshold level traffic during special occasion or network performance tests. To accommodate such situations, A2D2 IDS detection includes another preprocessor add-on FloodIgnoreHosts which allow the flood processor to ignore packets coming from hosts specified by the FloodIgnoreHosts preprocessor.

III.C Intrusion Tolerance – QoS

III.C.1 CBQ and Multi-Level Rate-Limiting

Based on user access policy, a certain percentage of available outbound bandwidth can be assigned to packets of various Types of Services (TOS). On the ingress side of the firewall gateway, multi-level rate-limiting will be applied. If a source can be confidently identified as an attacker, it is more effective for the firewall rule to drop all packets from that source. However, attack source identification is often difficult, especially with IP spoofing. Rate-limiting is often applied to suspicious source but legitimate traffic may be unnecessary discarded for a long time. A flood mitigation mechanism that is able to stop the most attack traffic while having the smallest impact on legitimate traffic is considered better than a mechanism that blocks a lock of legitimate traffic [For01]. To maximize the efficiency of rate-limiting, A2D2 proposes a multi-level rate-limiting mechanism.

It is conceivable that a network may generate a sudden burst of connection traffic. Such burst lasts for a very short period of time while the connection is established. Traffic from initiating network hosts temper off over time as the abundant of traffic should flow from the servers serving files or streaming video to the clients. A multi-level rate-limiting imposes stricter limits as the confidence that a source is malicious increases.

For example, if a source sends out 500 request packets per second, A2D2 can limit it to only 100 packets per second for a period of time. The surge of requests are usually temporary, if the suspicious source continue to send out the maximum allowable rate, the firewall can further restrict the incoming packet rate to 50 packets per second for a longer period of time. If the trend continues where a source continuously consume the maximum allowable rate, the source will be blocked completely.

III.D Autonomy System

To enable autonomous response, communication channels are established for the various components of the A2D2 detection and response systems. In addition, any firewall rules need to be autonomously applied and expired without administrators' intervention.

IV. Test-bed Performance Results

An A2D2 test-bed and a normal network without DDoS mitigation strategy were set up to test the effectiveness of the design. The setup of the test-bed followed the illustration of Figure 3. A 6-minutes video clip was served by RealServer to three clients. The DDoS tool Stacheldraht was used and attacks were launched at 150 seconds into each data collection period. Data was collected at the clients to show the number of packet received during the showing of the clip.

The traffic pattern experienced by clients of both the A2D2 and a regular network is presented in Figure 5. RealPlayer showed that about 15779 packets were received in the 6.5 minutes and no packet is lost. During the Stacheldraht DDoS attack, clients of a regular network that had no mitigation strategy experienced major interruption of service as shown in Figure 6. In fact, the RealPlayer was completely timed out and disconnected at 250 seconds into the data collection period, approximately 100 seconds after the launch of the attack. RealPlayer indicated that the clients of the regular network sent out 5476 retransmission requests but only recovered 56 packets before the application timed out. On the other hand, clients of the A2D2network enjoyed the same QoS during a DDoS attack as indicated in Figure 7. No packet was lost and no retransmission request was needed by the A2D2 clients.

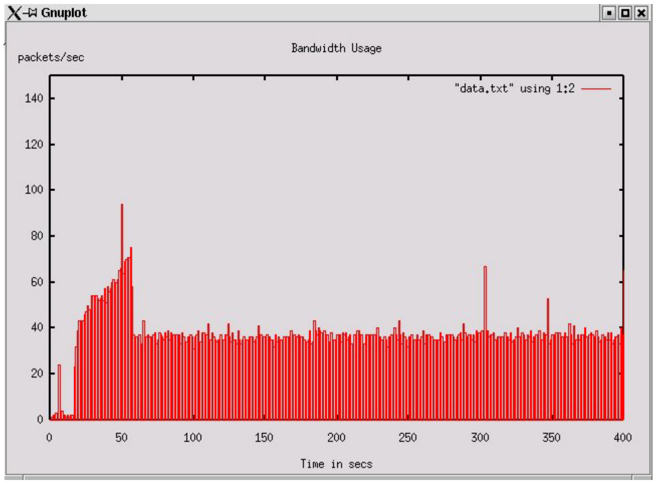

Figure 5 – QoS Experienced by Clients (Normal Traffic)

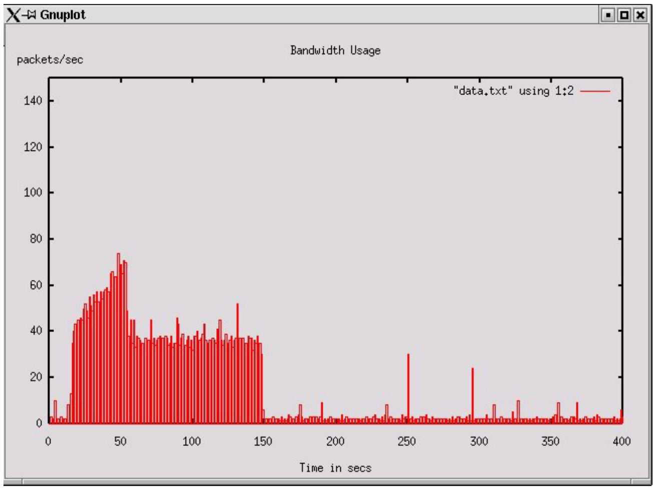

Figure 6 – QoS Experienced by Normal Network Clients (During DDoS Attack)

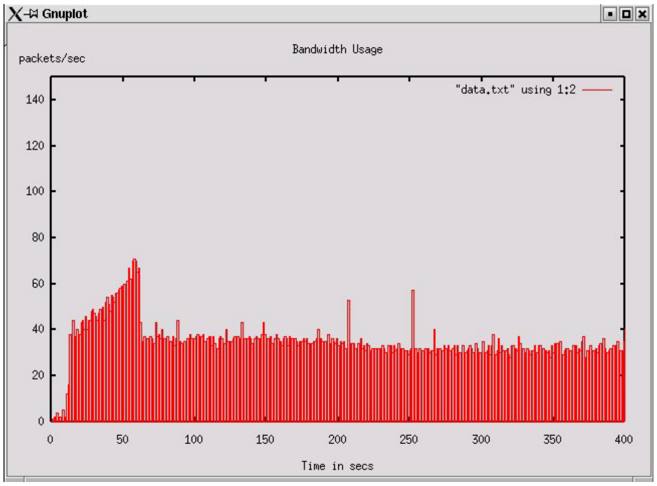

Figure 7 – QoS Experienced by A2D2 Clients (During DDoS Attack)

V. Conclusion

Results clearly show that the A2D2 design can effectively ensure QoS to its clients during a Stacheldraht DDoS attack. One limitation observed is that snort sometimes has to make several attempts before its alert messages can be accepted by the firewall machine. This delay may be attributed to the processing power of the firewall machine. The firewall gateway is a relatively basic Intel Pentium III 500 Mhz machine with 256 MB of memory. During an attack, the firewall computer is logging considerable activities and is heavily engaged in the rate-limiting, dropping, and forwarding of a large number of packets. Another possible reason to the delays in relaying snort alerts is that the internal link within the A2D2 may experience some level of congestion. The internal network between the Realserver and the firewall is connected by a 10 Mbps hub while the external links to the Internet is connected by a 100 Mbps switch. Therefore, it will be beneficial to conduct the test-bed experiment with a computer that has a faster processing power and with a 100 Mbps hub in the internal DMZ network.

Since Snort runs on a vast variety of platforms including Linux, Net/Open/FreeBSD, Solaris, SunOS 4.1, HP-UX, AIX, IRIX, Tru64, MacOS X Server and the Win9x/NT/2000 platform [Snort], the A2D2 network design can be easily ported to all type of networks used by small and medium networks. This study shows that the A2D2 design is a viable solution to combat the increasing DDoS threat targeted against small and medium networks that lack the financial and knowledge resources to set up an elaborate expensive infrastructure.

Bibliography

[Cis02] Cisco Systems. Overview. Cisco IOS Release 12.0 Quality of Service Solutions Configuration Guide.

[Comp] Computer Networking Glossary.

[Des02] Paul Desmond. Cisco, Enterasys Deliver New IDS Products. boston.internet.com. May 16. 2002.

[Dit99] David Dittrich. The "stacheldraht" distributed denial of service attack tool.
The DoS Project's "trinoo" distributed denial of service attack tool.
The "Tribe Flood Network" distributed denial of service attack tool
The "mstream" distributed denial of service attack tool

[Dit00] David Dittrich. "Usenix Security Symposium 2000 DDoS – Is there Really a Threat".

[For01] Jeff Forristal. Fireproofing Against DoS Attacks. Network Computing. December 10, 2001.

[HMP$^+$01] Allen Householder, Art Manion, Linda Pesante, George M. Weaver, Rob Thomas. Managing the Threat of Denial-of-Service Attacks. CERT Coordination Center. October 2001.

[IB02] John Ioannidis and Steve M. Bellovin. Implementing Pushback: Router-Based Defense Against DDoS Attacks. AT & T Research Lab. 2002.

[ITW01] ITWorld.com. CERT hit by DDoS attack for a third day. May 24, 2001.

[KMW01] Frank Kargl, Joern Maier, and Michael Weber. "Protecting Web Servers from Distributed Denial of Service Attacks". University of Ulm Germany, May 2001.

[MVS01] David Moore, Geoffrey M. Voelker and Stefan Savage. Inferring Internet Denial-of-Service Activity. 2001

[RL93] Y. Rekhter and T. Li. RFC 1518: An Architecture for IP Address Allocation with CIDR. September 1993.

[Sao02] Greg Saoutine et al. Barbarians at the Gate. Microsoft Professional Magazine. September 29, 2002.

[SDW$^+$01] Dan Sterne, Kelly Djahandari, Brett Wilson, Bill Babson, Dan Schnackenberg, Harley Holliday, and Travis Reid. "Autonomic Response to Distributed Denial of Service Attacks". Recent Advances in Intrusion Detection. 4th International Symposium, RAID 2001 Davis, CA, USA, October 10-12, 2001 Proceedings.

[Snort] Snort. The Open Source Network Intrusion System. http://www.snort.org.

[Whatis] Whatis?com. http://whatis.techtarget.com/definition/

[WO01] Jeroen Wortelboer and Jan Van Oorschot. Linux Firewall – the Traffic Shaper. January 15, 2001.

[YEA00] Jianxin Yan, Stephen Early, Ross Anderson. "The XenoService – A Distributed Defeat for Distributed Denial of Service. Proceedings of ISW 2000.

[ZOS00] Weibin Zhao, David Olshefski and Henning Schulzrinne. Internet Quality of Service: an Overview. Columbia University. 2000.