The Changing Face of Distributed Denial of Service Mitigation

In this paper we will review the traditional best practices and tools for DDoS mitigation, discuss the inherent weaknesses of these best practices, review the developing legal issues and trends that may soon be forcing change on how DDoS attacks are combated, and look at the new generation of tools becoming available for mitigating these attacks.

Justin Stephen
Version 1.2e

The Changing Face of Distributed Denial of Service Mitigation

On February 7, 2000, Internet mainstay Yahoo.com experienced a lengthy service outage. What became clear over the following hours was that the site had been victimized by a distributed denial of service (hereafter abbreviated to “DDoS”) attack from hundreds of geographically dispersed Internet-connected machines sending millions of request for service packets. Over the next few days, additional attacks were launched against six other major Web sites, among them some of the favorite sons of the then-burgeoning e-commerce revolution. The ultimate victims were Yahoo.com, Amazon.com, Buy.com (attacked a mere hour after their initial public stock offering), ZDNet.com, E-Trade.com, eBay.com, and CNN.com. According to the Yankee Group, estimated costs of the attack totaled $1.2 billion cumulative and the attack on Amazon alone cost between $200,000 and $300,000 per hour[i]. Losses of customer goodwill, corporate reputation and public trust may have been even greater.

Mainstream media coverage of these attacks was very heavy because of the sheer scale and likely because one of their own, CNN, was among the victims. Although the first recorded DDoS attacks had occurred years earlier, these February 2000 incidents marked a public coming out party for this type of cyberattack. Almost more disconcerting than the attacks themselves was the revelation of the identity of the perpetrator. A 15-year-old Canadian teenager, who went by the alias “Mafiaboy”, had researched and downloaded several hacker tools, such as AMDEX, Trank, and Slice 3, and launched the attacks using a DDoS tool called Tribe Flood Network (a.k.a. Trinoo). By some estimations, the only reason he was ultimately caught was because he bragged about his exploits in Internet chat rooms.

Major DDoS attacks still make the news. In January, Microsoft became the victim of such an attack. Microsoft’s main Web site and affiliated sites for MSN, online travel site Expedia.com, the auto sales site Carpoint, and the Microsoft email service Hotmail were unreachable for several hours. This past May, a DDoS was launched against the CERT Coordination Center, the government-appointed InfoSec watchdog that, for many, symbolizes security on the Web. By some estimates, losses from this attack total $100,000.

“We get attacked every day,” said Richard D. Pethia, a CERT director. “This is just another attack. The lesson to be learned here is that no one is immune to these kinds of attacks. They cause operational problems, and it takes time to deal with them.”[ii]

Just last month, the Internet-connected world was rocked by the Code Red worm. Exploiting a buffer overflow vulnerability in Microsoft Internet Information Server, the worm was able to infect 359,000 machines worldwide in just 14 hours. Those machines hosting sites whose default language was English were defaced and all infected machines served as a springboard for vicious propagation code that attempted to spread the worm to other machines. Part of the attack pattern ("phase 2") of the original iteration of this worm was to launch a DDoS attack against whitehouse.gov. Fortunately for the White House IT staff, not only was the worm hard-coded to check to make sure that port 80 at whitehouse.gov was active before launching, the IP address to be attacked was hard-coded as well. Whitehouse.gov systems administrators simply turned off the DNS server at the target IP (192.137.240.91), rerouting all requests to the other server. Additionally, ISPs worked together to "black hole" packets sent to the target IP.

Mainstream media coverage of the Code Red worm has also been very heavy, most focusing on the rapid spread of the worm. Truth be told, however, we all dodged a very large bullet with this worm. Despite its impressive rate of propagation, minimal damage was done. Hopefully, Code Red will serve as a wake-up call. It should also serve as an extremely nefarious omen of bigger and nastier DDoS attacks to come. Instead of the traditional model of DDoS slave, or "zombie", acquisition employed by Mafiaboy and others, wherein it can take weeks or months to crack into the slave machines needed for a large attack and plant the attack software, the Code Red worm built a slave army of 359,000 machines in just about 14 hours.

There still does not exist a tool or process that can fully protect a Web site from a DDoS attack. By many accounts, those seven Web sites victimized by Mafiaboy in February 2000 are only marginally better prepared to thwart such attacks today, well over a year later. The frequency of DDoS attacks continues to increase, going up 60 percent in the past 3 years. One-third of the respondents to the 2001 Computer Crime and Security Survey report having experienced denial of service attacks. It is also safe to say that the problem is under-reported. Many attacks go undetected at all and many organizations, fearing bad publicity and the consequent effect on their customers and stockholders, do not report those that are detected. Additionally, the attack tools available for launching these attacks are becoming more and more sophisticated and their schemes are getting increasingly complex. Security experts have identified more than seven primary DDoS tools and variants are appearing continuously. The painful reality is that any bored teenager can download most of these tools from the Web and launch his/her own DDoS in relatively short order.

In this paper we will review the traditional best practices and tools for DDoS mitigation, discuss the inherent weaknesses of these best practices, review the developing legal issues and trends that may soon be forcing change on how DDoS attacks are combated, and look at the new generation of tools becoming available for mitigating these attacks.

Traditional Defenses

Many of the basic practices that can help prevent or mitigate DDoS attacks should be included in any defense-in-depth enterprise security plan, even one not overly concerned with this particular risk. Among these are:

- Timely application of patches and system updates, especially to potentially exposed machines. For example, update and maintain a current build of BIND on DNS servers.
- Deployment of only strictly necessary network services.
- Intrusion detection systems.
- Firewalls.
- Anti-virus software.
- Good password policies.
- Use of Tripwire or other similar tools to detect changes in configuration information or other important files.
- Paying heed to “Top 20” vulnerability lists provided by the information security community and evaluating these risks against one’s environment.
- Establishment and maintenance of regular backup schedules and policies.
- As a network is only as secure as its weakest link, protection of mobile and remote machines with personal firewall/intrusion detection software.

Other best practices that can be employed at the user organization level that will help mitigate the risk of denial of service attacks include:

- Carefully architect the DNS server network, distributing DNS servers around the edge of the network and consider establishment of back-up relationships with other parties. Poor DNS server network design was a crucial factor in the January DDoS attack on Microsoft mentioned above. Additionally, safeguard information about the architecture and thus vulnerabilities of DNS networks.
- Address filtering, also known as “egress filtering”, of packets leaving the enterprise. This can ensure that packets leaving carry source addresses within the ranges of those sites. It can also ensure that no traffic from unroutable addresses (see RFC 1918) leave those sites.
- Turn off ICMP echo and chargen services unless there is a specific need for these services. This will prevent “smurf attacks” and similar vulnerabilities.
- Patches are available to help prevent TCP SYN flood attacks. Test and install them.
- Establish baselines for normal activity. This will help enable administrators to determine if there is a problem.
- User organizations should check their systems regularly to determine whether they have malicious software installed. There are a number of tools, many of them free, to assist in this effort. Some examples:
  - National Infrastructure Protection Center (NIPC)’s “find_ddos” tool is able to detect several old and more current DDoS tools including mstream, TFN2000 client and daemon, Trin00 daemon and master, TFN daemon and client, stacheldraht master, client and daemon and TFN-rush client.
  - RID, from David Brumley at Stanford University, is able to detect Trin00, TFN, and stacheldraht agents.
  - Zombie Zapper from Bindview Inc. works against Trinoo, TFN, stacheldraht, Troj_Trinoo (the trinoo agent ported to Windows), and Shaft.

- Invest in hot spares, machines that can be placed into action quickly in the event that a similar machine is disabled.
- Invest in more bandwidth to lower your vulnerability to flooding attacks. Invest in redundant load-balancing networks and servers. If there are multiple versions of the same Web site operating on different network segments, rogue packets can be distributed evenly amongst them making it more unlikely that any given server will crumble under the weight of an attack.
- Education and communication throughout the community can be extremely helpful. When organizations fail to share information about attacks, this helps give the hacker community an even greater advantage. Systems administrators should participate in industry-wide early warning systems. Information about attacks should be disseminated to vendors and response teams so that it can be applied to the defenses of others.

There are certainly things that network and hosting providers do now that can assist in the DDoS mitigation efforts. NSPs can utilize ingress filtering, similar to egress filtering but on a larger scale, to help combat IP spoofing. They can, and often do, respond to information from their customers and from other NSPs to combat malicious packets. The "black holing" of packets destined for whitehouse.gov during the recent Code Red attacks are but one example of this. Lastly, they can perform traffic and load monitoring that can provide early warning of some attacks.

Weaknesses of Traditional Defenses

There are certainly drawbacks to the practices described above. Ultimately, these drawbacks can be summed up as onerous levels of effort and the ultimate inability of user organizations to truly determine their own fate when it comes to DDoS attacks.

Keeping up to date with, researching, testing and implementing every applicable software patch and system update is a time consuming process, as are tuning, monitoring and updating firewalls and intrusion detection systems. Additionally, firewalls and IDSs were designed to detect discrete attacks against individual hosts or Web servers, not to detect and counter attacks against the network. As such, they do not provide the ability to monitor and characterize floods of abnormal network traffic in real time. By the time the attack hits, customer traffic has already been affected.

Egress filtering and use of tools to find DDoS malware on our own systems has far more benefit to our peers than to our own networks. Yes, if everyone took these steps, the incidences of DDoS attacks could be lowered dramatically. However, "depending on the other guy" is hardly a strategy likely to bear much fruit when the attacks can come from any Internet-connected system anywhere in the world.

The type of rate-limiting or service denial that works well against such ICMP-based attacks as "smurf attacks" is almost useless against TCP traffic. Current filtering capabilities on most routers is too coarse-grained, inflexible, and slow to effectively handle the work we need them to do. Firewalls, on the other hand, possess

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE