radware

DDoS SURVIVAL HANDBOOK

The Ultimate Guide to Everything You Need To Know About DDoS Attacks

How to:

- » Identify Attack Types and Understand Their Effects
- » Recognize Attack Tools
- » Protect Your Organization Against DoS and DDoS Attacks

SHARE THE DDoS SURVIVAL HANDBOOK

Table of Contents

1 Introduction

Although the Internet was designed to allow for easy sharing of information between various interconnected computers and networks, it was not designed with security in mind. The digital equivalents of viruses, pathogens, and other threats have been around since the dawn of the Internet. In 1988, when the Internet's precursor, ARPANET, consisted of roughly 60,000 connected machines, a self-replicating computer program called the Morris Worm unintentionally caused about 10% of these machines to malfunction by exhausting their computing resources. Yet some individuals, businesses, and other organizations still do not properly protect themselves.

With over 1 billion users today, the Internet has become a conduit for people and businesses to regularly access useful information, perform tasks such as banking, and shop at many different retailers. The rise of social media has also rendered the Internet an invaluable place for businesses and other organizations to use for critical branding and other core customer interactions – often generating significant revenue in the process. The downside of all this convenience is vulnerability to disruption. Malicious users are often able to steal information or halt normal computer operation, with motives ranging from industrial espionage and revenge to financial gain and political aims.

A cyber attack by a malicious party aiming to disrupt a website on the Internet (or any device connected to it) is called an availability-based attack. Using a wide spectrum of different attack vectors (TCP floods, HTTP/S floods, low rate attacks, SSL attacks, etc.), availability-based attacks is one of the most serious security threats affecting websites. They are commonly referred to as denial-of-service (DoS) attacks. When the attack is carried out by more than one attacking machine, it is called a distributed denial-of-service (DDoS) attack.

DoS and DDoS attacks make news headlines around the world daily, with stories recounting how a malicious individual or group was able to cause significant downtime for a website or use the disruption to breach security, causing financial and reputational damage. While information security researchers have yet to develop a standardized strategy to collect data regarding the number or nature of DoS and DDoS attacks that occur around the world, it is estimated that over 7,000 such attacks occur daily – a number that has grown rapidly in recent years.[1]

Every organization with a website – especially one that requires its users to have regular access to sensitive information – should take urgent and appropriate steps to protect against DoS and DDoS attacks. Failure to do so can result in huge financial losses as well as a damaged public reputation.

The DDoS Survival Handbook is your key to survival against cyber attackers that may be stalking you right now without your even knowing it. This handbook offers trusted, proven tips for safeguarding your business against DoS and DDoS attacks. Its goal is to increase your familiarity with DoS and DDoS attacks and help you understand how they can affect your organization. It will explain how DoS and DDoS attacks work, how they can impact your business, who is behind the attacks, what tools they're using, and what resources are available at your disposal as a means of defense.

[1] http://www.prolexic.com/pdf/Prolexic_corp_brochure_2012.pdf

2 Understanding DoS and DDoS Attacks

What is a DoS attack? What is a DDoS attack? What's the difference? How are they created? What are their strengths and weaknesses? Before discussing any survival techniques, you must first understand from what you are trying to survive.

To provide a figurative example of a DoS attack, imagine yourself walking into a bank that only has a single teller window open. Just as you are about to approach the teller, another person rushes into the bank and cuts in front of you. This person begins making small talk with the teller, and has no intention of performing any bank-related transactions. As a legitimate user of the bank, you are left unable to deposit your check, and are forced to wait until the "malicious" user has finished his or her conversation. Just as this malicious user leaves, another person rushes into the bank, again cutting to the front of the line ahead of you and forcing you to keep waiting. This process can continue for minutes, hours, even days, preventing you or any of the other legitimate users who lined up behind you from performing bank transactions.

During DoS attacks, attackers bombard their target with a massive amount of requests or data – exhausting its network or computing resources and preventing legitimate users from having access. More simply, a DoS attack is when an attacker uses a single machine's resources to exhaust those of another machine, in order to prevent it from functioning normally. Large web servers are robust enough to withstand a basic DoS attack from a single machine without suffering performance loss (imagine if the bank in the above example had many teller windows open for you to use to avoid waiting for the busy one).

However, attackers will often carry out DDoS attacks, which employ multiple machines for increased effectiveness, in effect, by trying to tie up all of the tellers at all of the open windows. In that scenario, it can often be harder to detect and block attackers manually, so special defenses are necessary to detect and defend against such large-scale attacks. Additionally, attackers almost never legitimately control their attacking machines; rather, they infect thousands of computers spread across the world with specialized malware in order to gain unauthorized access to such machines. A collection of hundreds or thousands of compromised machines acting as an army under the control of one attacker is called a “botnet”, and oftentimes the actual owners of machines that are part of a botnet are unaware that their computers have been compromised and are being used to launch DDoS attacks.

Amassing a Botnet

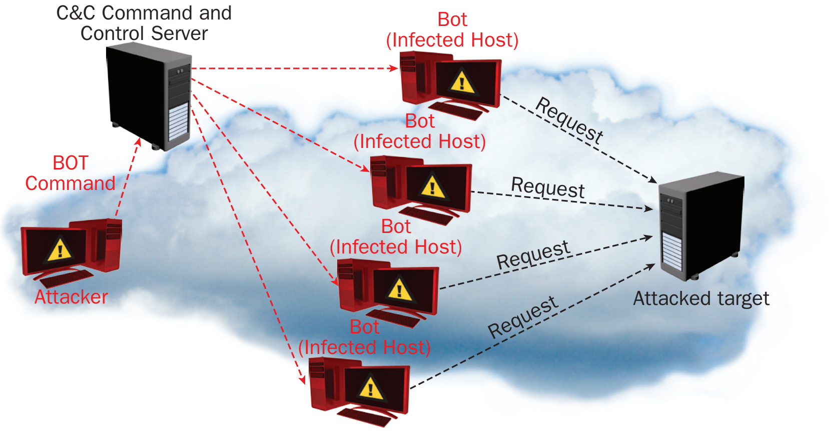
In order for attackers to create large botnets of computers under their control (referred to colloquially as zombies), they have two options: the more common option of using specialized malware to infect the machines of users who are unaware that their machines are compromised, or the relatively newer option of amassing a large number of volunteers willing to use DoS programs in unison.

In the former scenario (by far the most common), attackers will develop or purchase from various underground cyber crime forums specialized malware, which they spread to as many vulnerable computers as possible. Any users tricked into running such malware will often disable antivirus functionality on their computer, and install a “backdoor”, or access point, for attackers. Infected computers begin accepting communications from “command and control” (C&C) servers, centralized machines that are able to send commands to botnet machines, usually by means of Internet Relay Chat (IRC), a communication protocol designed for chat rooms. Anytime attackers want to launch a DDoS attack, they can send messages to their botnet’s C&C servers with instructions to perform an attack on a particular target, and any infected machines communicating with the contacted C&C server will comply by launching a coordinated attack.

When law enforcement officials attempt to dismantle a botnet, it is often necessary to locate and disable C&C servers, as doing so prevents most botnets from remaining operational. One particular botnet that was dismantled in 2010, called “Mariposa” (Spanish for “butterfly”), was found to contain nearly 15.5 million unique IP addresses around the world with many associated command and control servers.[2] More recent and advanced botnet software such as TDL-4, however, has implemented special inter-bot communication abilities over public peer-to-peer networks to help circumvent efforts to dismantle botnets solely through the disabling of C&C servers.

[2] Mariposa Botnet Takedown (Part 1) - Chris Davis, Defense Intelligence.pdf

In the case in which many computers are voluntarily acting in unison, hackers sponsoring an attack will publish its details via a social networking site or an IRC channel, including a date and time, a target IP or URL, and instructions on which of the available attack tools to use. Some attack campaigns following this model have succeeded in recruiting many supporters. The main drawback for such voluntary, coordinated DDoS attacks, however, is that the majority of the attack tools used does not mask their users' identities. One such tool, Low Orbit Ion Cannon (LOIC), was notorious for this – many LOIC users failing to use external means to hide their IP address have been located and arrested by the FBI and other law enforcement organizations around the world for participating in coordinated voluntary attacks. News of these recent arrests may deter some new users from opting to participate in such voluntary, coordinated attacks.

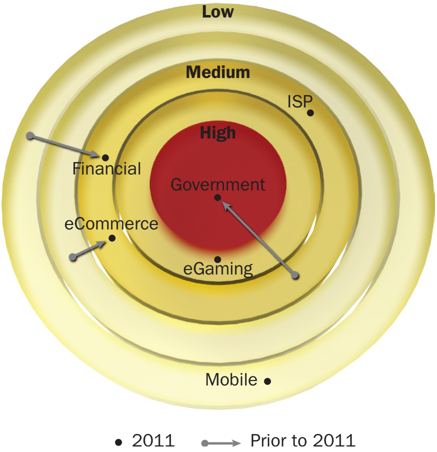


Launching an Attack

With the exception of amassing a botnet, launching a DDoS attack is not a particularly difficult task to carry out, even for a non-technical individual. Users do not need to create their own botnets in order to launch large-scale attacks, as various dedicated pay-for-hire DDoS services are available for anyone to use. Anyone using such a service can launch a powerful DDoS attack on a target of their choice for anywhere from $5 to $200 per hour, depending on the attack size and duration.

Business Impact

Various surveys on DDoS attacks have highlighted interesting facts on the impact of DDoS on targeted companies. According to a Neustar survey, 70% of the surveyed companies were victims of a DDoS attack that caused some level of damage.[3] While DDoS attacks may have had more industry-specific targets in the past, such attacks target all sectors today – financial services, governments, online retailers, and online gaming, among others. The following diagram taken from Radware's 2011 Global Application and Network Security Report[4] illustrates this trend.



The business impact of a DDoS attack is substantial, and can affect a victim over a period of time depending on the extent of the attack. According to both the Neustar and Radware reports, the DDoS attacks perpetrated in 2011 lasted anywhere from several hours to several days, with an average duration of about 24 hours. The effects from a DDoS attack can vary depending on the sector a target company belongs to and the volume of its online business. Often, these effects are both qualitative and quantitative, and can involve financial losses, reputational damage, and legal repercussions.

Financial Losses

The cost to an organization when its Website experiences downtime varies significantly depending upon the sector to which that particular

[3] Neustar Insight – DDoS Survey Q1 2012
[4] 2011 Global Application and Network Security Report

organization belongs. The Neustar survey found that organizations depending mainly or exclusively on the Internet for their business (notably online retail or gaming sites) estimated an average daily revenue loss of $2,000,000 – nearly $100,000 per hour – in the case of downtime, while other sectors, such as financial services, report a smaller yet significant average loss of $10,000 per hour in the event of downtime.

This calculation takes into account a few different elements: the cost of the attack itself, revenue loss from customers' and potential customers' inability to access the Website, time spent answering customer support calls, and possible additional financial penalties. Most serious attackers carefully plan their attacks, striking during critical periods for their target Website, for example during the holiday shopping season for an online retailer.

The wave of DDoS attacks that targeted major Websites such as Yahoo and Amazon in 2000 was estimated cumulatively to have cost over $1.2 billion in damages.[5] The total cost of the more recent attacks on Sony's Websites remains unclear and is difficult to estimate. Over $170M has been spent by Sony for cleanup related to the DDoS attack and loss of data, but some analysts estimate an ultimate cost of hundreds of dollars to Sony per each one of the 77 million compromised user accounts – amounting to billions of dollars in damages.[6] Regardless of analyst estimates, one thing is clear: the cost incurred by an organization that is not adequately protected against DDoS attacks can be exorbitantly high.

Customer Attrition

The most significant business impact outlined by surveyed companies is that related to its customers. A customer who attempts to access an organization's Website but is unable to do so because of downtime cannot buy anything, access information, or generally use any services. If he or she is unsatisfied, complains, requests for financial restitution, or even increased business for competitors may result.

According to the American Express 2011 Global Customer Service Barometer, consumers spend more money wherever they have a

[5] SANS Institute's "The Changing Face of Distributed Denial of Service Mitigation"
[6] Kazuo Hirai's Letter to the US House of Representatives

positive purchase experience and encounter good customer service.[7] Google engineers have discovered t the average online customer is not willing to wait an extra 400 milliseconds for a page to load – "literally the blink of an eye" as per a New York Times article8. Online customers require quick access to information, and according to Microsoft, would visit a Website less often if it is slower than that of its competitors by more than 250 milliseconds.[8] Consequently, a DDoS attack that prevents the targeted company's Website from providing adequate service to its users can result in customer dissatisfaction, angry support calls, and even customer attrition.

Reputation Loss

Businesses want to make headlines by showing off merits and achievements. Management teams dislike being forced to admit vulnerabilities in the media. When it becomes publicly known that a company has been a victim of a cyber attack that has compromised its customers and their data, the ensuing bad publicity can have devastating effects on both reputation and future sales. Any company falling prey to hackers becomes an example of "what not to do", and the ensuing fallout often involves replacing the IT team that allowed the disruption or break, corporate rebranding, and expensive public relations to regain the trust of the public.

Legal Pursuits

Customers affected by the unavailability of online services who can prove that they suffered damages may attempt to pursue financial restitution by means of filing a lawsuit, often arguing that the company did not take enough precaution against the possibility of such an attack. In one example, a major stock exchange, hit by a DDoS attack in 2011, was forced to suspend trading and pay penalties to trading firms to compensate for their inability to provide normal service.

Conclusion

The ability of an organization to protect itself against DoS and DDoS attacks is essential for its success. Without proper protection mechanisms, an organization targeted by a DoS or DDoS attack is likely to experience financial loss, reputational damage, and legal expense – all of which are likely to permanently affect its future.

[7] http://about.americanexpress.com/news/docs/2011x/AXP_2011_csbar_market.pdf

[8] http://www.nytimes.com/2012/03/01/technology/impatient-web-users-flee-slow-loading-sites.html?pagewanted=all

3 Evolution of DDoS

The Early Days

The first ever DoS attack occurred in 1974 and was carried out by David Dennis, a 13-year-old student at University High School, located across the street from the Computer-based Education Research Laboratory (CERL) at the University of Illinois Urbana-Champaign. David had recently learned about a new command that could be run on CERL's PLATO terminals called "external" or "ext", meant to allow for interaction with external devices connected to the terminals. When run on a terminal with no external devices attached, however, it would cause the terminal to lock up and require a shutdown and power-on to regain functionality. As a mischievous 13-year-old, he wanted to see what it would be like for a room full of users to be locked out at once, so he wrote a program that would send the "ext" command to many PLATO terminals at the same time. One morning, he went over to CERL and tested his program; it resulted in all 31 users having to power off at once. He continued to test his program at other locations around town and the country, eventually delighted to see mass postings about PLATO terminals locking up. Eventually the acceptance of a remote "ext" command was switched off by default, fixing the problem.

During the mid-to-late 1990s, when Internet Relay Chat (IRC) was becoming popular, some users fought for control of non-registered chat channels, where an administrative user would lose his or her powers if he or she logged off. This behavior led hackers to attempt to force users within a channel to all log out, so they could enter the channel alone and gain administrator privileges as the only user present. These "king of the hill" battles in which users would attempt to take control of an IRC channel and hold it in the face of attacks from other hackers were fought through the use of very simple bandwidth-based DoS attacks and IRC chat floods. Such attacks are akin to a stronger person physically pushing weaker people off of a designated hill or out of another area in a real-world "king of the hill" game.

Since DoS and DDoS attacks were predominant then in the world of IRC but not elsewhere, the public did not pay much attention to their potential impact. Many organizations banned the use of IRC, either blocking the servers or moving them to a demilitarized zone (DMZ)

– a separate logical sub network within an organization’s computer network that exposes any devices within it to the Internet. This practice not only did not solve the DoS problem, but it also created a perfect environment for DoS attacks to develop into the powerful form of cyber attacks they are today.

The Spread of DDoS and DDoS Tool Democratization

One of the first large-scale DDoS attacks occurred in August 1999, when a hacker used a tool called “Trinoo” to disable the University of Minnesota’s computer network for over two days. Trinoo was basic and without any anonymity features; it consisted of a network of compromised machines called “Masters” and “Daemons”, allowing an attacker to send a DoS instruction to a few Masters, which then forwarded instructions to the hundreds of Daemons to commence a UDP flood (see Chapter 7 for descriptions of specific attack types) against the target IP address. The tool made no effort to hide the Daemons’ IP addresses, so the owners of the attacking systems were contacted and had no idea that their systems had been compromised and were being used in an attack. Other early tools include Stacheldraht (German for “barbed wire”), which could be remotely updated and supported IP spoofing, and tools such as Shaft and Omega, which had the ability to collect attack statistics from their victims. Because hackers were then able to get information about their attacks, they could better understand the effect of certain types of attacks, and receive notification when an attack was detected and stopped.

Once hackers began to focus on distributed denial-of-service attacks, DoS attacks began to attract public attention. The “distributed” nature of a DDoS attack makes it significantly more powerful, as well as harder to identify and to block its source. With such a formidable weapon in their arsenal, hackers began to take on bigger and more prominent targets using improved tools and methods.

DDoS Attacks Make the Headlines

During February 2000, DDoS attacks truly caught the public’s attention. Several of the most well-known Internet sites at the time were targeted, including Yahoo, CNN, Amazon, Buy.com, E*Trade, and ZDNet. Even the Website of the FBI, the foremost prosecutor of cybercrime, was brought offline for three hours by a DDoS attack. Every site that was targeted was, and still is, a carefully monitored and well-provisioned site, accustomed to heavy, fluctuating volumes of traffic. Despite this, each targeted Website experienced some level of downtime as a result of the February 2000 DDoS attacks. If these organizations were vulnerable, it is not hard to see how the average business would be exposed.

Another notable DDoS attack that took place during the early 2000s targeted all 13 of the Internet's root domain name service (DNS) servers in 2002. DNS is an essential Internet service, as it translates host names in the form of uniform resource locators (URLs) into IP addresses. In effect, DNS is a phonebook maintaining a master list of all Internet addresses and their corresponding URLs. Without DNS, users would not be able to efficiently navigate the Internet, as visiting a Website or contacting a specific device would require knowledge of its IP address. DNS is a hierarchical system, as smaller DNS servers rely on other larger DNS servers; on the highest level there are 13 root name servers, without which the world's DNS system would fail.

The effect of a powerful DDoS attack on all 13 of the root name servers simultaneously would be catastrophic – Internet browsing would be slow or even unusable for everyone in the world. During the 2002 attack on the root name servers, all 13 servers experienced heavy load, and some of them were unreachable from parts of the global Internet. Although the Internet was still usable, for about an hour users may have noticed delays of up to a few seconds for some name queries. Even though the attack was not entirely successful, it proved that with enough resources, such an attack could have a much more significant impact.

Criminal Extortion and Furthering a Political Agenda

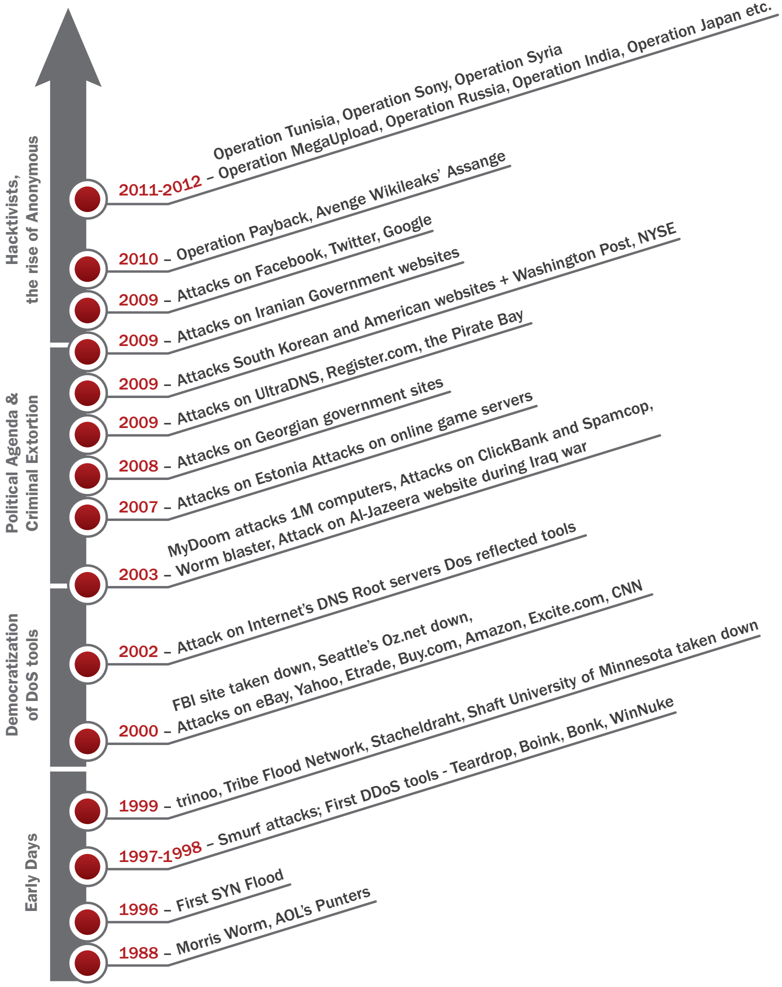
As DDoS attacks continued to occur around the world, motivations began to evolve. Hackers started specifically launching attacks as a means of attempting extortion. They sent messages to online retailer sites, gambling sites, and pornography sites, saying that they could prevent a future attack by the "third party" that perpetrated the original attack for some amount of "protection money". Sites that complied could be branded as "payers", and used as targets in subsequent attacks. Websites Clickbank and Spamcop were the target of such attacks in 2003.

In a different vein, instances of politically motivated and cyber-warfare-related DDoS attacks have increased. During the Second Gulf War, a DDoS attack took down Qatar-based Al-Jazeera News; in 2004, North Korean hackers attacked computers in South Korea and Japan; and in 2007-2008, Russia emphasized its use of DDoS attacks as a part of its cyber wars against Estonia and later Georgia.

The Rise of Anonymous

While the number of criminal extortion and cyber-warfare-related DDoS attacks continue to grow, many instances of politically motivated attacks are kept secret by the targeted companies in an effort to avoid bad publicity. In particular, attacks by Anonymous, a politically motivated “hacktivist” group, started to make the headlines from 2007-2008 (see Chapter 5 for more on Anonymous) beginning with “Project Chanology”, an attack that targeted the Church of Scientology. Since then, Anonymous has appeared frequently in the news, actively posting videos and messages on social networking sites in order to coordinate its protests – in the form of both cyber attacks and physical gatherings.

Timeline



Hacktivists, the rise of Anonymous

2011-2012 – Operation Tunisia, Operation Sony, Operation Syria Operation MegaUpload, Operation Russia, Operation India, Operation Japan etc.

2010 – Operation Payback, Avenge Wikileaks’ Assange

2009 – Attacks on Facebook, Twitter, Google

2009 – Attacks on Iranian Government websites

2009 – Attacks South Korean and American websites + Washington Post, NYSE

Political Agenda & Criminal Extortion

2009 – Attacks on UltraDNS, Register.com, the Pirate Bay

2009 – Attacks on Georgian government sites

2008 – Attacks on Estonia Attacks on online game servers

2007 – Attacks on ClickBank and Spamcop,

2003 – MyDoom attacks 1M computers, Worm blaster, Attack on Al-Jazeera website during Iraq war

Democratization of DoS tools

2002 – Attack on Internet’s DNS Root servers Dos reflected tools

2000 – FBI site taken down, Seattle’s Oz.net down, Attacks on eBay, Yahoo, Etrade, Buy.com, Amazon, Excite.com, CNN

Early Days

1999 – trinoo, Tribe Flood Network, Stacheldraht, Shaft University of Minnesota taken down

1997-1998 – Smurf attacks; First DDoS tools - Teardrop, Boink, Bonk, WinNuke

1996 – First SYN Flood

1988 – Morris Worm, AOL’s Punters

4 Who is Behind the Attacks and What are the Motives?

The frequency of cyber attacks has increased sharply in recent years, as the number of individuals and organizations choosing to launch such attacks on their competitors or enemies have also increased, as has the use of potentially vulnerable computers and computer networks. While a large number of attacks are financially motivated – anything from crippling a business competitor to criminal extortion – many others are politically motivated or even just for the "lulz" (Internet slang for "fun"). No one, however, should doubt the seriousness or potential cost of a successful attack.

Financial Gain

Organizations using DDoS attacks for the purpose of financial gain fall into two categories: those intending to gain an advantage over competitors and those attempting to carry out criminal extortion. Any legitimate organization that employs a third-party pay-for-hire DDoS service to attack competitors can put that competitor at a significant disadvantage; as such attacks are disproportionally costly to the subject of the attack compared to what the attacking company pays for the DDoS services.

Entities offering pay-for-hire DDoS services will often resort to criminal extortion. Criminal extortion by means of DDoS begins with the extorting company picking a target business and launching a relatively small "sample" DDoS attack against them. This attacking company will then send a message to its target, suggesting that they have the power to prevent an additional, more severe DDoS attack from the "third party" that already launched an attack, and will do so for some amount of money (usually in the range of thousands of dollars). If the attacked company complies with a payment, they risk being branded a "payer" by the DDoS-for-hire service and used as a target for future extortion attempts. In this situation, it often becomes necessary to deploy some form of DDoS mitigation solution to prevent future attacks.

Anonymous—a loosely associated computer "hacktivist" group responsible for many of the major politically motivated cyber attacks that have occurred over the last few years – formed in 2003 on the imageboard 4chan as a joking referral to the name "Anonymous" assigned to each user's post. Anonymous has perpetuated its opposition to Internet censorship through both physical and cyber protests as an anarchistic decentralized body. Because Anonymous is completely decentralized and has no leadership or ranking system, anyone can "join" by simply wanting to do so. Protests and cyber attacks are coordinated by means of imageboards, forums, wikis, IRC, YouTube, and social networking services and any member of Anonymous can organize events as a means of working toward a set of his or her own goals parallel to the "Anonymous" agenda.

In cyberspace, Anonymous's attacks are often perpetuated through the distributed use of flooding tools such as Low Orbit Ion Cannon (LOIC) and its newer cousin High Orbit Ion Cannon (HOIC). By recruiting a large number of users to voluntarily participate in such attacks – usually over IRC, as it is a more anonymous means of communication – Anonymous effectively creates a "voluntary botnet" of thousands of computers. Using a vast number of machines running LOIC or HOIC to target even a fairly large server often results in a denial-of-service condition, making Anonymous formidable as a cyber attacker.

Political Motivation

Aside from financial gain by crippling competitors or resorting to criminal extortion, others are motivated to launch DDoS attacks for political or entertainment motivations (often a combination of both). These relatively newer motivations mark an evolution in the world of cyber attacks, leading to the coining of the term "hacktivism", meaning the use of cyber attacks to further a political agenda. Various hacker groups, such as Anonymous and (the now dismantled) LulzSec, perpetrate such attacks, often targeting supporters of legislation they deem unfavorable and various governmental agencies related to such legislation. Aside from the anti-piracy-related Operation Payback, other attacks (or attempted attacks) by Anonymous and other "hacktivist" groups have included "Operation AntiSec", "Operation Blackout", and "Operation Defense". Some of the most famous attacks have targeted large government agencies around the world, including the United States FBI and British SOCA.

Advanced Persistent Threats and Cyber Warfare

Any organization or individual with both a persistent motive and the advanced means to execute such a non-indiscriminate, stealthy cyber attack is known as an advanced persistent threat (APT). APTs are likely to play a large role in the future, as the ability to steal intelligence or cripple an enemy's cyber infrastructure through DDoS and other attacks could prove equally or perhaps even more devastating than physical attacks alone. In recent years, the cyber security world witnessed the discovery of highly intricate pieces of malware such as Duqu, Stuxnet, and Flame, proving that an individual, organization, or nation with enough resources is able to create such a powerful cyber warfare tool and effectively deploy it without detection.

Even without proprietary malware, APTs can rent or employ their own massive botnets – large networks of infected machines – to launch non-vulnerability-based DDoS attacks that can cause significant damage to network infrastructure, preventing legitimate users from accessing crucial servers or network devices. Furthermore, terrorist APTs can use such advanced pieces of malware or other computing resources to inflict damage on both government and civilian computer infrastructure, causing significant harm to those who have their data stolen or their computers malfunction.

Many attacks against government agencies are politically motivated attacks. However, the hacker group LulzSec successfully mounted attacks against United States and other governmental agencies during the summer of 2011 mostly for entertainment; their motto was, "The world's leaders in high-quality entertainment at your expense." During the peak of LulzSec's existence – a period of 50 days during which they broke into the computer networks of governments, companies, and other individuals – they made public vast quantities of private information including many usernames, passwords, and personal identifying information. While the original LulzSec is no longer in operation, a new individual or group dubbing itself LulzSec Reborn has already carried out two high-profile attacks in March and June.

With a rise in the use of computers, computer-aided devices, and computer networks has become a significant evolution in the nature and complexity of cyber attacks. Not only are cyber attacks carried out by APTs – individuals or organizations possessing significant resources and a specific target – but also by a variety of other actors ranging from legitimate businesses to organized crime, and even to amateur “hackers” with non-financial motives (such as LulzSec).

5 What It's Like to Get Hit With a DDoS Attack – An Inside View

It is not always obvious to a network or system administrator that the company's infrastructure is under attack. An attack usually starts slowly, and only as the attack progresses further will someone take notice. Below is a hypothetical scenario as described hour-by-hour by a system administrator of a company under a DDoS attack.

5:30 a.m.

I am awakened by the sound of an incoming SMS message on my phone. It reads, "Warning, mainapp server at 30% maximum load."

Such a message is an automatic notification sent by the new server health-monitoring tool we recently installed, while mainapp is the principal online banking application Web server that handles customer requests. Since our CEO has strategically decided to promote online banking and launch a marketing campaign to encourage customers to use the online banking application, the bank has invested a great deal of money to ensure that the mainapp banking application Web server is robust, scalable, and highly available. So far, it seems to have enough processing power and memory to handle current traffic, as last month's statistics showed a server load of no more than 15%.

Receiving a message indication that server load is at 30% is worrisome, but not serious. It is possible that the alert threshold parameters were set incorrectly in the monitoring tool, but I can wait to check that when I get to the office later.

6:00 a.m.

Only a half hour later another SMS message arrives. This one reads "Warning, mainapp server at 50% maximum load." Something is definitely wrong.

Since I did not configure remote access to the health-monitoring tool, I cannot look at its logs. While rushing to get to the office to investigate, I run through the possible causes of such high server load. I try to assure myself that it is probably a simple configuration error, but I begin to worry. My phone rings – it is one of my co-workers, another network administrator. She received the same warning notification as I did and wants to know whether I am aware of the situation.

7:00 a.m.

The customer support manager on duty calls me while I am still on my way, reporting that many customers are calling to complain that the online banking Website is significantly slower than usual. He says that one of the customers is furious because he was unable to perform a time-sensitive money transfer as quickly as usual, and that he switched to online banking so he could avoid that type of problem. This particular customer was so angry that he threatened to sue the bank for his financial losses due to the slow transaction.

Finally I arrive at the office, and rush to a server terminal screen. Mainapp's load has reached 70%—nearly maximum.

Upon a quick check of the health monitoring tool logs, I find out that the alert thresholds are set correctly. Network traffic is still appearing abnormally high, so this is not an alert threshold issue. Thousands of connections have been opened to the server, requesting different pages on the online banking Website.

A few beads of sweat drip down my forehead as I try not to panic. Such a massive amount of network traffic must be originating from a malicious source, but why? Who is behind it? I suddenly remember last week's newspaper headlines, detailing the wave of cyber attacks on financial services. I immediately recall similarities between what our server is experiencing and what I remember reading about in the papers, as I begin to fear that our server is being targeted by a denial-of-service attack.

8:00 a.m.

Assuming the worst, I begin to try and identify the nature and source of the malicious network traffic. First, I check where the connections are originating from and try to isolate the attackers' IP addresses, in order to differentiate the legitimate from the malicious traffic. Meanwhile, my phone has not stopped ringing.

The CIO calls wanting to know what is going on; I tell him that I am trying to solve the problem but that we might be under a denial-of-service attack that's exhausting our server's resources. He does not respond, and I feel a moment of hopelessness. He just tells me that the problem needs to be solved quickly, before the CEO gets involved.

I have no clue how to stop the attack, and I am not even sure that it is actually denial-of-service. I've never seen anything like this in my career. My only knowledge on the subject comes from some reading I did on the Internet after attending last month's security seminar.

Looking at the IP trace, it seems that all of the malicious connections are coming from various different sources. Each IP is repeatedly sending HTTP GET requests for various online banking pages, and this action is hogging all of mainapp's resources making the online banking pages slow for legitimate users.

With some idea of what is going on, I decide on a short-term plan of action and call an emergency team meeting.

8:30 a.m.

The situation has not gotten any better. The pace of the attack has been constant, but now mainapp hardly responds to any kind of request. The customer support manager at my office is upset, as all of his staff is being overwhelmed by support calls. Customers are unhappy and angry, but what can he instruct them to say? I tell him that I think we are under attack by one or more hackers, that we should not expect to regain normal service soon, and that we may release a formal statement in the near future regarding our downtime.

Meanwhile, I contact our ISP for help, sending them our server logs. Although our bandwidth is not completely saturated yet, I want them to know what's going on and that they should be prepared to provide us with support if necessary.

9:00 a.m.

The situation has now become catastrophic. Word has spread, and the entire staff is in a state of panic. The emergency meeting I called convenes; it consists of the CIO, CTO, network administrators, security manager, application manager, and system administrators (including me). We are tense, but understand that we have to issue an official message to the customers and decide on a plan of action to deal with the attack.

I show everyone the logs, and after a few minutes the security manager notices that some of the malicious requests are coming from Russia. Quickly, I define a rule on the mainapp web server to reject all requests originating from Russia thinking it may slow down the attack. Unfortunately, it doesn't help. After activating my new filter, I see no decrease in the amount of malicious traffic. After a brief period with no new connections, additional connections begin to originate from a dozen different countries, including ours!

9:30 a.m.

The server is still under heavy load; obviously, blocking IPs based on geographic region did not help, so we have to look for another solution. Understanding that we were not prepared to handle such an attack, it has become necessary to gain further understanding of how to prevent and mitigate a denial-of-service attack.

10:00 a.m.

The mainapp Web server is completely flooded, and the online banking site is offline. Upon this news, the CEO decides to get involved. She emphasizes how bad it is for the bank's reputation to announce such an attack, and wonders how much it will cost the bank in revenue loss and customer dissatisfaction. She is worried that if the details of this attack leak to the press it could cause panic among the bank's customers. She reiterates that the attack must be mitigated quickly, by whatever means necessary and vaguely threatens the jobs of the IT staff.

10:15 a.m.

We need expert assistance in mitigating DDoS attacks.

Top Expert Lessons for Surviving a DDoS Attack

You can't be carefree and foolish when it comes to protecting your online business from DDoS attacks. But don't despair: organizations can take back control by following some simple measures. Add these to your need-to-know list:

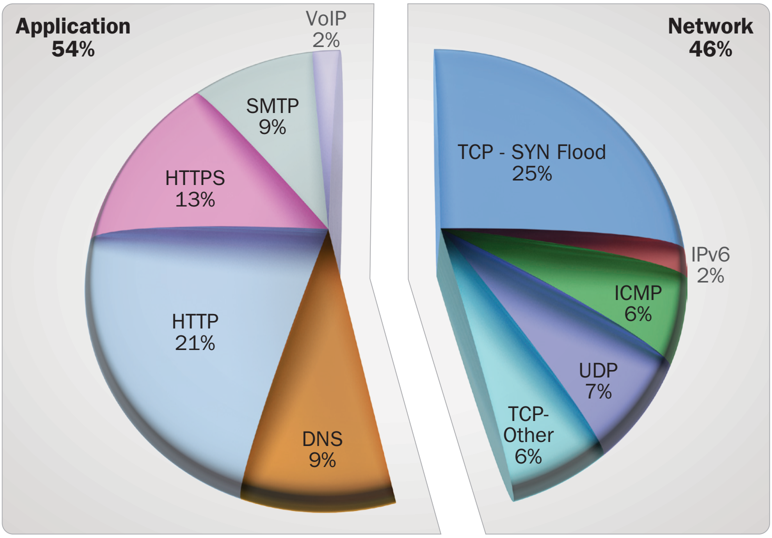
1. No organization is ever safe, only safer.
2. Be prepared for DDoS attacks. Organize a defense strategy ***before*** you're attacked
3. Make sure you're honest about the state of your security readiness. Identify potential security holes, have the right tools and people in place, and be wary of 'free' or 'bolt-on' tools.
4. Perform business risk analysis to determine the right budget to allocate.
5. Induct everyone in the security team. Responsibility for security is no longer the sole province of the security group.
6. The attack may be gone, but the threat lives on. Collect information about attacks such as type, size and frequency. Use the correct measures ***per attack type***.
7. Test your DDoS mitigation systems and make sure they are capable of detecting and mitigating today's threats.
8. Simulate a DDoS attack on your organization and make sure that each staff member knows their role during an attack.
9. You don't actually have to take it sitting down. You can defend yourself while taking an offensive position that can neutralize your attacker. Study the rhythm and intent of the attacker so you can apply an effective counter-technique.

6 Attack Types and Their Effects

Attacks Type Evolution

As mentioned in previous sections, DDoS attacks have evolved considerably over the years. Their democratization is largely due to the ease with which one can launch an attack today, as well as generally poor preparation by most organizations against even some of the most basic DDoS attack types. Tutorials instructing inexperienced users how to carry out such attacks are widely available across the Internet, and one can even rent a botnet through a pay-for-hire DDoS service to increase an attack's power.

Attackers do not take the risk of "missing" their targets once they have committed; they will often change their attack vectors in order to attempt to circumvent defense measures that are in place. Many modern attacks typically use multiple vectors in a single attack campaign, targeting multiple components of an organization's network infrastructure and its applications. In 2011, 56% of cyber attacks were targeted at applications; 46 % at the network. Attacks now include at least 5 different attack vectors in a single campaign.[9] And they're working longer – ensuring the acronym APT (advanced persistent threat) remains a dominant part of our lexicon.



[9] 2011 Global Application and Network Security Report

Attacks will not only attempt to consume network resources, but in some cases server (and other stateful device) or application resources as well.

Classifying the different types of DoS and DDoS attacks by using only one dimension is exceptionally difficult. Each type of attack has different characteristics that may suggest it belongs to multiple categories. Generally speaking, types of attacks include those that target network resources, those that target server resources, and those that target application resources. The following is a list of some the most common attacks and their technical underpinnings.

Operation Payback was a series of cyber attacks initiated by the hacker group Anonymous, in retaliation for the United States government's crackdown on WikiLeaks for having exposed confidential government documents and communications. During Operation Payback, Anonymous targeted sites such as Visa, MasterCard, and PayPal, as they had all stopped accepting donations for WikiLeaks. The main purpose of these attacks was to protest perceived injustice by disrupting the target companies' services, causing them both financial losses and public humiliation. What made the attack especially unique was that Anonymous, for the first time on such a large scale, recruited inexperienced volunteers to download a special DDoS tool that allowed them to participate in the attacks alongside the more experienced hackers using botnets.

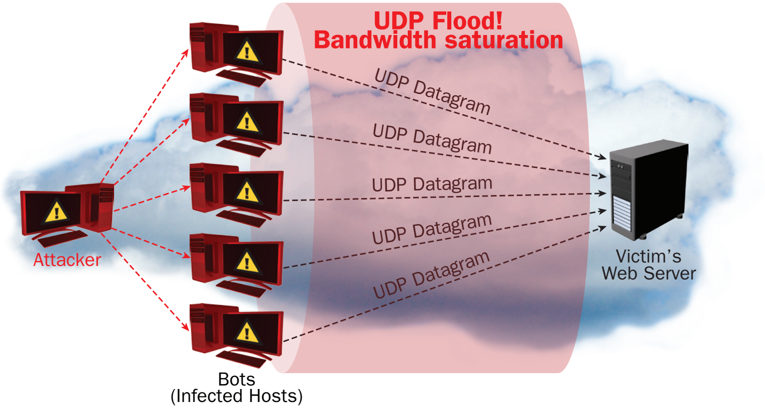
Operation Sony was a series of cyber attacks on the Sony PlayStation Network that both damaged Sony's reputation and hurt it financially. It was a classic case in which hackers used a DDoS attack to distract their target from their true objective – data theft. The DDoS attack was well-planned and well-executed; it allowed for the hackers to steal the account information of over 77 million users of Sony's PlayStation Network. Because Sony was so busy dealing with the DDoS attack, it was unaware for a long time that any information had been stolen.

Attacks Targeting Network Resources

Attacks that target network resources attempt to consume all of a victim's network bandwidth by using a large volume of illegitimate traffic to saturate the company's Internet pipe. Attacks of this manner, called network floods, are simple yet effective. In a typical flooding attack, the offense is distributed among an army of thousands of volunteered or compromised computers – a botnet – that simply sends a huge amount of traffic to the targeted site, overwhelming its network. While requests of this manner may seem legitimate in small numbers; in large numbers they can be significantly harmful. A legitimate user trying to access a victim's site under a flooding attack will find the attacked site incredibly slow or even unresponsive.

Floods

UDP Flood: User Datagram Protocol (UDP) is a connectionless protocol that uses datagrams embedded in IP packets for communication without needing to create a session between two devices (and therefore requiring no handshake process). A UDP Flood attack does not exploit a specific vulnerability, but rather simply abuses normal behavior at a high enough level to cause network congestion for a targeted network. It consists of sending a large number of UDP datagrams from potentially spoofed IP addresses to random ports on a target server; the server receiving this traffic is unable to process every request, and consumes all of its bandwidth attempting to send ICMP "destination unreachable" packet replies to confirm that there was no application listening on the targeted ports. As a volumetric attack, a UDP flood is measured in Mbps (bandwidth) and PPS (packets per second).



ICMP Flood: Internet Control Message Protocol (ICMP) is another connectionless protocol used for IP operations, diagnostics, and errors. Just as with a UDP flood, an ICMP flood (or Ping Flood) is a non-vulnerability based attack; that is, it does not rely on any specific vulnerability to achieve denial-of-service. An ICMP Flood can involve any type of ICMP message of echo request; once enough ICMP traffic is sent to a target server, it becomes overwhelmed from attempting to process every request, resulting in a denial-of-service condition. An ICMP Flood is also a volumetric attack, measured in Mbps (bandwidth) and PPS (packets per second).

IGMP Flood: Internet Group Management Protocol (IGMP) is yet another connectionless protocol, used by IP hosts (computers and routers) to report or leave their multicast group memberships for adjacent routers. An IGMP Flood is non-vulnerability based, as IGMP allows multicast by design. Such floods involve a large number of IGMP message reports being sent to a network or router, significantly slowing down and eventually preventing legitimate traffic from being transmitted across the target network.

An **Amplification Attack** is any attack in which an attacker is able to use an amplification factor to multiply the power of an attack. For instance, the attacker could use a router as an amplifier, taking advantage of the router's broadcast IP address feature to send messages to multiple IP addresses which the source IP (return address) is spoofed to the target IP. Famous examples of amplification attacks include Smurf Attacks (ICMP amplification) and Fraggle Attacks (UDP amplification). Another example of a type of amplification attack is DNS amplification, in which an attacker, having previously compromised a recursive DNS name server to cache a large file, sends a query directly or via a botnet to this recursive DNS server, which in turn opens a request asking for the large cached file. The return message (significantly amplified in size from the original request) is then sent to the victim's (spoofed) IP address, causing a denial-of-service condition.

A **connection-oriented** attack is one in which the attacker must first establish a connection prior to launching his or her DDoS attack. The outcome of this attack usually affects the server or application resources. TCP- or HTTP-based attacks are examples of connection-oriented DDoS attacks.

A **connectionless attack**, on the other hand, does not require the attacker to open a complete connection to the victim, and therefore is much easier to launch. The outcome of a connectionless attack affects network resources, causing denial-of-service before the malicious packets can even reach the server. UDP or ICMP floods are examples of connectionless DDoS attacks.

An attack is **reflective** when the attacker makes use of a potentially legitimate third party to send his or her attack traffic, ultimately hiding his or her own identity.

Attacks Targeting Server Resources

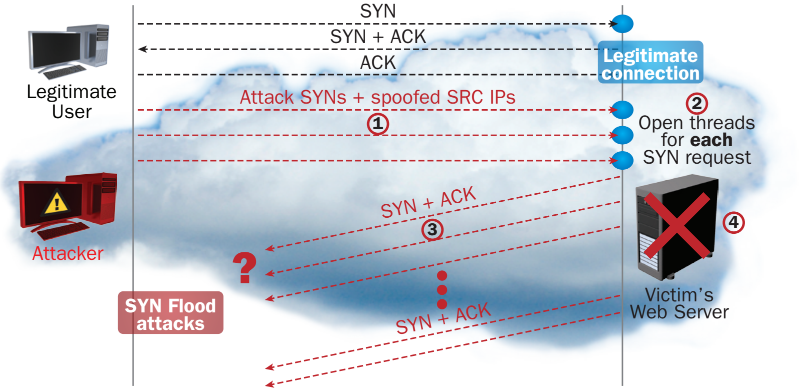
Attacks that target server resources attempt to exhaust a server's processing capabilities or memory, potentially causing a denial-of-service condition. The idea is that an attacker can take advantage of an existing vulnerability on the target server (or a weakness in a communication protocol) in order to cause the target server to become busy handling illegitimate requests so that it no longer has the resources to handle legitimate ones. "Server" most commonly refers to a Website or Web application server, but these types of DDoS attacks can target stateful devices such as firewalls and IPSs as well.

TCP/IP Weaknesses

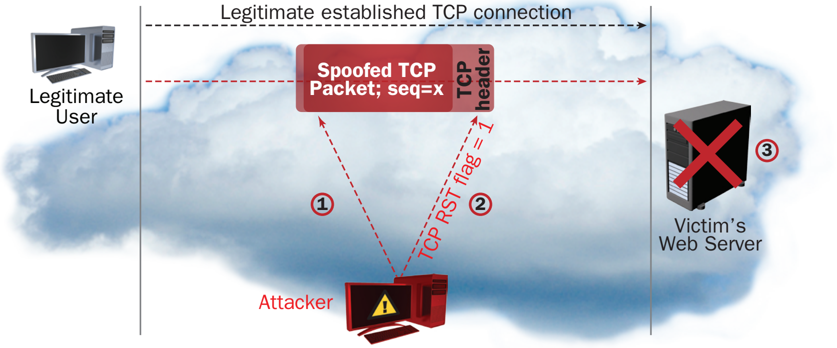
These types of attacks abuse the TCP/IP protocol by taking advantage of some of its design weaknesses. They typically misuse the six control bits (or flags) of the TCP/IP protocol – SYN, ACK, RST, PSH, FIN, and URG – in order disrupt the normal mechanisms of TCP traffic. TCP/IP, unlike UDP and other connectionless protocols, is connection-based, meaning that the packet sender must establish a full connection with his or her intended recipient prior to sending any packets. TCP/IP relies on a three-way handshake mechanism (SYN,

SYN-ACK, ACK) where every request creates a half-open connection (SYN), a request for a reply (SYN-ACK), and then an acknowledgement of the reply (ACK). Any attack that attempts to abuse the TCP/IP protocol will often involve sending TCP packets in the wrong order, causing the target server to run out of computing resources as it attempts to understand such abnormal traffic.

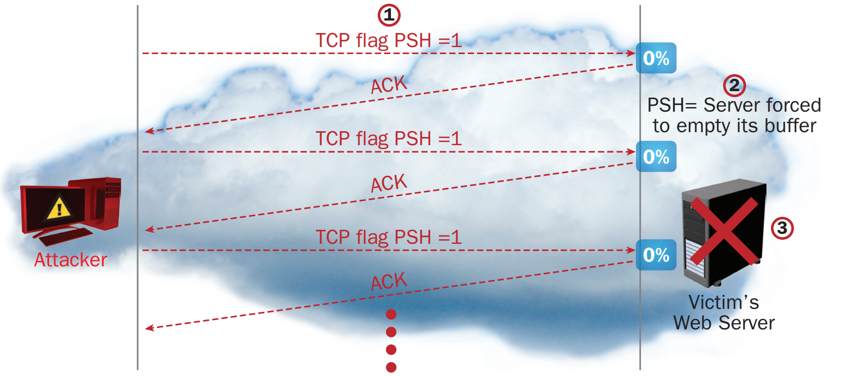
TCP SYN Flood: In the TCP handshake mechanism, there must be an agreement between each party for a connection to be established. If the TCP client does not exist or is a non-requesting client with a spoofed IP, such an agreement is not possible. In a TCP SYN, or simply SYN flood attack, the attacking clients lead the server to believe that they are asking for legitimate connections through a series of TCP requests with TCP flags set to SYN, coming from spoofed IP addresses. To handle each of these SYN requests, the target server opens threads and allocates corresponding buffers to prepare for a connection. It then tries to send a SYN-ACK reply back to the requesting clients to acknowledge their connection requests, but because the clients IP addresses are spoofed or the clients are unable to respond, an acknowledgement (ACK packet) is never sent back to the server. The server is still forced to maintain its open threads and buffers for each one of the original connection requests, attempting to resend its SYN-ACK request acknowledgement packets multiple times before resorting to a request time-out. Because server resources are limited and a SYN flood often involves a massive number of connection requests, a server is unable to time-out its open requests before even more new requests arrive, and this causes a denial-of-service condition.



TCP RST Attack: The TCP RST flag is intended to notify a server that it should immediately reset its corresponding TCP connection. In a TCP RST attack, the attacker interferes with an active TCP connection between two entities by guessing the current sequence number and spoofing a TCP RST packet to use the client's source IP (which is then sent to the server). A botnet is usually used to send thousands of such packets to the server with different sequence numbers, making it fairly easy to guess the correct one. Once this occurs, the server acknowledges the RST packet sent by the attacker, terminating its connection to the client located at the spoofed IP address.



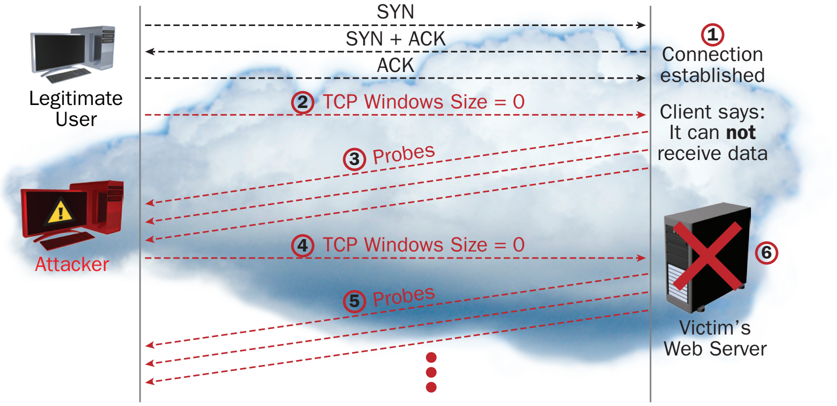
TCP PSH+ACK Flood: When a TCP sender sends a packet with its PUSH flag set to 1, the result is that the TCP data is immediately sent or "pushed" to the TCP receiver. This action actually forces the receiving server to empty its TCP stack buffer and to send an acknowledgement when this action is complete. An attacker, usually using a botnet, can flood a target server with many such requests. This overwhelms the TCP stack buffer on the target server, causing it to be unable to process the requests or even acknowledge them (resulting in a denial-of-service condition).



“Low and Slow” Attacks

Unlike floods, “low and slow” attacks do not require a large amount of traffic. They target specific design flaws or vulnerabilities on a target server with a relatively small amount of malicious traffic, eventually causing it to crash. “Low and slow” attacks mostly target application resources (and sometime server resources), and are very difficult to detect as they involve connections and data transfer that appears to occur at a normal rate.

Sockstress: Sockstress is an attack tool that exploits vulnerabilities in the TCP stack allowing an attacker to create a denial-of-service condition for a target server. In the normal TCP three-way handshake, a client sends a SYN packet to the server, the server responds with a SYN-ACK packet, and the client responds to the SYN-ACK with an ACK, establishing a connection. Attackers using Sockstress establish a normal TCP connection with the target server but they send a “window size 0” packet to the server inside the last ACK, instructing it to set the size of the TCP window to 0 bytes. The TCP Window is a buffer that stores the received data before it uploads it up to the application layer. The Window Size field indicates how much more room is in the buffer in each point of time. Window size set to zero means that there is no more space whatsoever and that the other side should stop sending more data until further notice. In this case the server will send window size probe packets to the client continually to see when it can accept new information, but because the attacker does not change the window size, the connection is kept open indefinitely. By opening many connections of this nature to a server, the attacker consumes all of the space in the server’s TCP connection table (as well as other tables), preventing legitimate users from establishing a connection. Alternately, the attacker can open many connections with a very small (around 4-byte) window size, forcing the server to break up information into a massive number of tiny 4-byte chunks. Many connections of this type will consume a server’s available memory, also causing a denial-of-service.



SSL-Based Attacks

With the rise of Secure Socket Layer (SSL), a method of encryption used by various other network communication protocols, attackers have begun to target it. SSL runs above TCP/IP conceptually, and provides security to users communicating over other protocols by encrypting their communications and authenticating communicating parties. SSL-based DoS attacks take many forms: targeting the SSL handshake mechanism, sending garbage data to the SSL server, or abusing certain functions related to the SSL encryption key negotiation process. SSL-based attacks could also simply mean that the DoS attack is launched over SSL-encrypted traffic, which makes it extremely difficult to identify; such attacks are often considered “asymmetric”, as it takes significantly more server resources to deal with an SSL-based attack than it does to launch one.

Encrypted-based HTTP attacks (HTTPS floods): Many online businesses utilize SSL/TLS (Transport Layer Security) increasingly in their applications to encrypt their traffic and secure end-to-end transit of data. DoS attacks on encrypted traffic are on the rise and mitigating them is not as obvious as might be expected. Most DoS mitigation technologies do not actually inspect SSL traffic, as it requires decrypting the encrypted traffic. HTTPS Floods – which are floods of encrypted HTTP traffic (HTTP Floods are explained below) – are now frequently participating in multi-vulnerability attack campaigns. On top of the “normal” HTTP Floods impact, encrypted HTTP attacks add several other challenges such as the burden of encryption and decryption mechanisms.

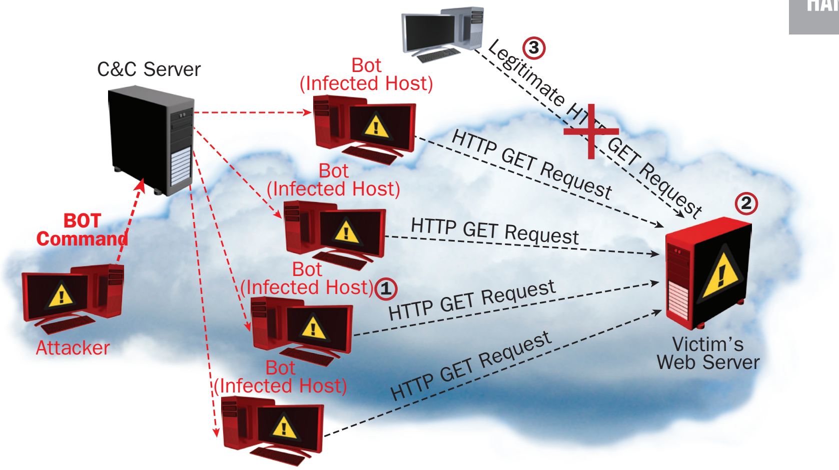
THC-SSL-DOS: This tool was developed by a hacking group called The Hacker's Choice (THC) as a proof-of-concept to encourage vendors to patch their SSL vulnerabilities. THC-SSL-DOS, as with other "low and slow" attacks, requires only a small number of packets to cause denial-of-service for even a fairly large server. It works by initiating a regular SSL handshake, and then immediately requesting for the renegotiation of the encryption key, constantly repeating this renegotiation request again and again until all server resources have been exhausted. Attackers love to launch attacks that use SSL, because each SSL session handshake consumes fifteen times more resources from the server side than from the client side. In fact, a single standard home PC can take down an entire SSL based web server and several computers can take down a complete farm of large secured online services.

Attacks Targeting Application Resources

Instances of DoS attacks that target application resources have grown recently and are widely used by attackers today. They target not only the well-known Hypertext Transfer Protocol (HTTP), but also HTTPS, DNS, SMTP, FTP, VOIP, and other application protocols that possess exploitable weaknesses allowing for DoS attacks. Just as attacks that target network resources, there are different types of attacks that target application resources, including both floods and "low and slow" attacks. The latter are particularly prominent, mostly targeting weaknesses in the HTTP protocol. HTTP, as the most widely used application protocol on the Internet, is an attractive target for attackers.

HTTP Flood

An HTTP flood is the most common application-resource-targeting DDoS attack. It consists of what seem to be legitimate, session-based sets of HTTP GET or POST requests sent to a victim's Web server, making it hard to detect. HTTP flood attacks are typically launched simultaneously from multiple computers (volunteered machines or bots), that continually and repeatedly request to download the target site's pages (HTTP GET flood), exhausting application resources and resulting in a denial-of-service condition. Modern DDoS attack tools such as High Orbit Ion Cannon (HOIC) offer an easy-to-use means of performing multi-threaded HTTP flood attacks.



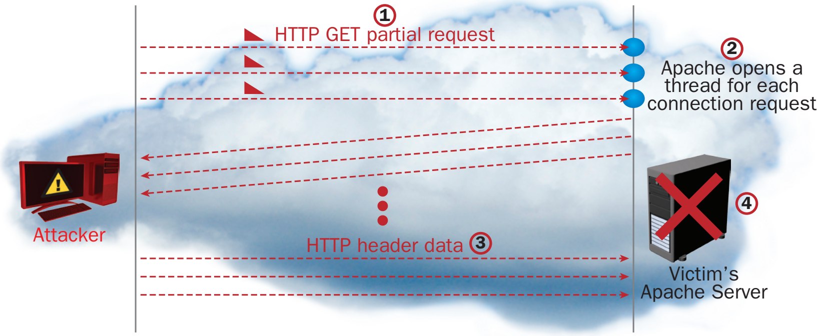
DNS Flood

A DNS flood is easy to launch, yet difficult to detect. Based on the same idea as other flooding attacks, a DNS flood targets the DNS application protocol by sending a high volume of DNS requests. Domain Name System (DNS) is the protocol used to resolve domain names into IP addresses; its underlying protocol is UDP, taking advantage of fast request and response times without the overhead of having to establish connections (as TCP requires). In a DNS flood, the attacker sends multiple DNS requests to the victim's DNS server directly or via a botnet. The DNS server, overwhelmed and unable to process all of its incoming requests, eventually crashes.

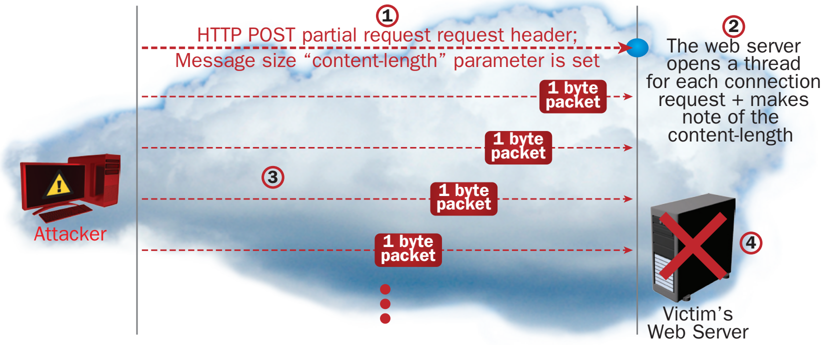
"Low and Slow" Attacks

The characteristics of the "low and slow" attacks in this section relate more particularly to application resources (whereas the previous "low and slow" attacks targeted server resources). These "low and slow" attacks target specific application vulnerabilities, allowing an attacker to stealthily cause denial-of-service. Not volumetric in nature, such attacks can often be launched with only a single machine; additionally, because these attacks occur on the application layer, a TCP handshake is already established, successfully making the malicious traffic look like normal traffic traveling over a legitimate connection.

Slow HTTP GET Request: The idea behind a slow HTTP GET request is to dominate all or most of an application's resources through the use of many open connections, preventing it from providing service to users wishing to open legitimate connections. In this attack, the attacker generates and sends incomplete HTTP GET requests to the server, which opens a separate thread for each of these connection requests and waits for the rest of the data to be sent. The attacker continues to send HTTP header data at (slow) set intervals to make sure the connection stays open and does not time out. Because the rest of the required data arrives so slowly, the server perpetually waits, exhausting the limited space in its connection table and thereby causing a denial-of-service condition.



Slow HTTP POST Request: In order to carry out a slow HTTP POST request attack, the attacker detects forms on the target Website and sends HTTP POST requests to the Web server through these forms. The POST requests, rather than being sent normally, are sent byte-by-byte. As with a slow HTTP GET request, the attacker ensures that his or her malicious connection remains open by regularly sending each new byte of POST information slowly at regular intervals. The server, aware of the content-length of the HTTP POST request, has no choice but to wait for the full POST request to be received (this behavior mimics legitimate users with slow Internet connection). The attacker repeats this behavior many times in parallel, never close an open connection, and after several hundred open connections, the target server is unable to handle new requests, hence achieving a denial-of-service condition.


1
HTTP POST partial request request header;
Message size "content-length" parameter is set
2
The web server opens a thread for each connection request + makes note of the content-length
1 byte packet
1 byte packet
3
1 byte packet
1 byte packet
Attacker
4
Victim's Web Server

Regular Expression DoS attacks: A special case of "low and slow" attacks is RegEx DoS (or ReDos) attacks. In this scenario, the attacker sends a specially crafted message (sometimes called evil RegExes) that leverages a weakness in a library deployed in the server, in this case, a regular expression software library. This causes the server to consume large amounts of resources while trying to compute a regular expression over the user-provided input, or to execute a complex and resource-hungry regular expression processing dictated by the attacker.

Hash Collisions DoS attacks: This kind of attack targets common security vulnerabilities in Web application frameworks. In short, most application servers create hash tables to index POST session parameters and are sometimes required to manage hash collisions when similar hash values are returned. Collision resolutions are resource intensive, as they require an additional amount of CPU to process the requests. In a Hash Collision DoS attack scenario, the attacker sends a specially crafted POST message with a multitude of parameters. The parameters are built in a way that causes hash collisions on the server side, slowing down the response processing dramatically. Hash Collisions DoS attacks are very effective and could be launched from a single attacker computer, slowly exhausting the application server's resources.

7 Attack Tools

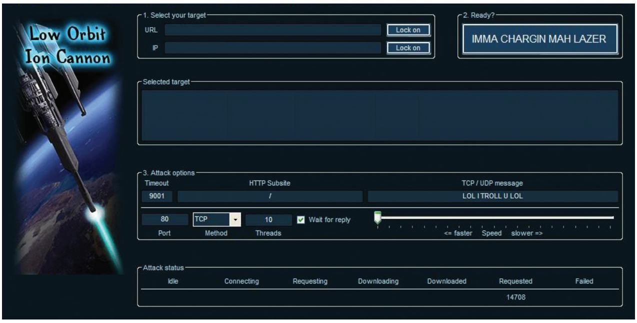
The previous chapters discussed various types of DDoS attacks occurring on both the network and application layers. While it is possible to execute many of these attacks manually, specialized attack tools have been developed for the purpose of executing attacks more easily and efficiently. The first DDoS tools – examples of which include Trinoo and Stacheldraht – were widely used around the turn of the century, but were somewhat complex and only ran on the Linux and Solaris operating systems.

In more recent years, DDoS tools have become much more straightforward to use and cross-platform, rendering DDoS attacks much easier to carry out for attackers and more dangerous for targets. Some of these newer DDoS tools, such as Low Orbit Ion Cannon (LOIC), were originally developed as network stress testing tools and later modified and used for malicious purposes, while others such as Slowloris were developed by "gray hat" hackers – those aiming to draw the public's attention to a particular software weakness by releasing such tools publicly so the makers of the vulnerable software would be forced to patch it in order to avoid large-scale attacks. Additionally, just as the network security and hacking world is constantly evolving, so are the attack tools used to carry out DDoS attacks. New attack tools are becoming smaller in size, more effective at causing a denial-of-service condition, and more stealthy.

Low Orbit Ion Cannon (LOIC)

"Hacktivist" group Anonymous's original tool of choice – Low Orbit Ion Cannon (LOIC) – is a simple flooding tool, able to generate massive amounts of TCP, UDP, or HTTP traffic in order to subject a server to a heavy network load. While LOIC's original developers, Praetox Technologies, intended the tool to be used by developers who wanted to subject their own servers to such a heavy network traffic load for testing purposes, Anonymous picked up the open-source tool and began using it to launch coordinated DDoS attacks.

Soon afterwards, LOIC was modified and given its "Hivemind" feature, allowing any LOIC user to point his or her copy of LOIC at an IRC server, transferring control of it to a master user who can then send commands over IRC to every connected LOIC client simultaneously. In this configuration, users are able to launch much more effective DDoS attacks than those of a group of less-coordinated LOIC users not operating simultaneously. In late 2011, however, Anonymous began to step away from LOIC as their DDoS tool of choice, as LOIC makes no effort to obscure its users' IP addresses. This lack of anonymity resulted in the arrest of various users around the world for participating in LOIC attacks, and Anonymous broadcasting a clear message across all of its IRC channels: "Do NOT use LOIC."



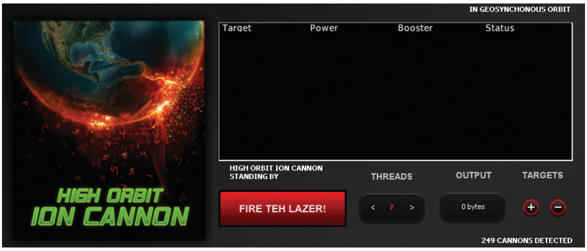
Timeout: 9001
HTTP Subsite: /
TCP / UDP message: LOL I TROLL U LOL
Port: 80
Method: TCP
Threads: 10
☑ Wait for reply

Idle	Connecting	Requesting	Downloading	Downloaded	Requested	Failed
					14708	

High Orbit Ion Cannon (HOIC)

After Anonymous "officially" dropped LOIC as its tool of choice, LOIC's "successor", "High Orbit Ion Cannon (HOIC), quickly took the spotlight when it was used to target the United States Department of Justice in response to its decision to take down Megaupload.com. While HOIC is also a simple application at its core – a cross-platform Basic script for sending HTTP POST and GET requests wrapped in an easy-to-use GUI – its effectiveness stems from its add-on "booster" scripts, or additional text files that contain additional Basic code interpreted by the main application upon a user's launch of an attack.

Even though HOIC does not directly employ any anonymity techniques, the use of booster scripts allows a user to specify lists of target URLs and identifying information for HOIC to cycle through as it generates its attack traffic, making HOIC attacks slightly harder to block. HOIC continues to be used by Anonymous all over the world to launch DDoS attacks, although Anonymous attacks are not limited to those involving HOIC.



hping

In addition to LOIC and HOIC, Anonymous and other hacking groups and individuals have employed various other tools to launch DDoS attacks, especially due to the Ion Cannons' lack of anonymity. One such tool, hping, is a fairly basic command line utility similar to the ping utility; however, it has more functionality than the sending of a simple ICMP echo request that is the traditional use of ping. hping can be used to send large volumes of TCP traffic at a target while spoofing the source IP address, making it appear random or even originating from a specific user-defined source. As a powerful, well-rounded tool (possessing some spoofing capabilities), hping remains on Anonymous's list of tools of choice.

Slowloris

Besides straightforward brute-force flood attacks, many of the more intricate "low and slow" attack types have been wrapped up into easy-to-use tools, making for denial-of-service attacks that are much harder to detect. Slowloris, a tool developed by a gray hat hacker who goes by the handle "RSnake", is able to create a denial-of-service condition for a server by using a very slow HTTP request. By sending HTTP headers to the target site in tiny chunks as slow as possible (waiting to send the next tiny chunk until just before the server would time out the request), the server is forced to continue to wait for the headers to arrive. If enough connections are opened to the server in this fashion, it is quickly unable to handle legitimate requests.

R U Dead Yet? (R.U.D.Y.)

Another slow-rate denial-of-service tool similar to Slowloris is R U Dead Yet? (R.U.D.Y.). Named after the Children of Bodom album "Are You Dead Yet?" R.U.D.Y. achieves denial of service by using long form field HTTP POST submissions rather than HTTP headers, as Slowloris does. By injecting one byte of information into an application POST field at a time and then waiting, R.U.D.Y. causes application threads to await the end of never-ending posts in order to perform processing (this behavior is necessary in order to allow Webservers to support users with slower connections). Since R.U.D.Y. causes the target Webserver to hang while waiting for the rest of an HTTP POST request, a user is able to create many simultaneous connections to the server with R.U.D.Y., ultimately exhausting the server's connection table and causing a denial-of-service condition.

#RefRef

While all the aforementioned tools are non-vulnerability-based, #RefRef, another tool in Anonymous's arsenal, is based on vulnerability in the widely used SQL database software allowing for an injection attack. Using an SQL injection, #RefRef allows an attacker to cause a denial-of-service condition for a target server by forcing it to use a special SQL function (which allows for the repeated execution of any other SQL expression). This constant execution of a few lines of code consumes the target servers' resources, resulting in denial-of-service. Unlike LOIC or HOIC, #RefRef does not require a vast number of machines in order to take down a server due to the nature of its attack vector. If the server's backend uses SQL and is vulnerable, only a few machines are needed to cause a significant outage. While developing the tool, Anonymous tested it on various sites, easily causing outages for minutes at a time, and requiring only 10-20 seconds of a single machine running #RefRef. In one such attack (on Pastebin), a 17-second attack from a single machine was able to take the site offline for 42 minutes.

The Botnet as a DDoS Tool

Regardless of the attack tool used, however, the ability to launch an attack from multiple computers – whether it is hundreds, thousands, or millions – significantly amplifies the potential of an attack to cause denial-of-service. Attackers often have "botnets" at their disposal – large collections of compromised computers, often referred to as "zombies", infected with malware that allows an attacker to control them. Botnet owners, or "herders", are able to control the machines in their botnet by means of a covert channel such as IRC (Internet Relay Chat), issuing commands to perform malicious activities such as distributed denial-of-service (DDoS) attacks, the sending of spam mail, and information theft.

As of 2006, the average size of the average botnet around the world was around 20,000 machines (as botnet owners attempted to scale down their networks to avoid detection), although some larger more advanced botnets, such as BredoLab, Conficker, TDL-4, and Zeus have been estimated to contain millions of machines. Large botnets can often be rented out by anyone willing to pay as little as $100 per day to use them (one particular online forum ad offered the use of a botnet containing 80,000-120,000 infected hosts for $200 per day), allowing almost anyone with only moderate technical knowledge and the right tools to launch a devastating attack. With this in mind, it is important to be aware of all recent attack tools, maintain up-to-date software on all servers and other network devices, and use some kind of in-house DDoS mitigation solution to protect against attacks as they continue to evolve.

8 Protecting Your Organization from DDoS Attacks

Even though DoS and DDoS attacks have been around for several years, many organizations continue to ignore the potential impact of such threats. The rise of hacktivism perpetrated by groups such as Anonymous in the form of DDoS attacks has brought more focus to DDoS attacks in the eye of corporations. Even though DoS threats managed to get the attention of CSOs, many organizations have not yet defined their anti-DoS strategies. In a recent survey conducted by research firm Neustar, it was found that only 3%, of surveyed organizations had a dedicated anti-DoS solution.[10] The vast majority of organizations hope that their existing network security products such as firewalls and IPSs (or even switches and routers) will block DoS attacks. This is a dangerous mindset to have.

Why Your Firewall Cannot Block DDoS Attacks

At the beginning of 2012, Radware's ERT released its annual security report[11] based on dozens of DoS and DDoS attacks that the team handled during 2011. The ERT checked which network devices were bottlenecks during these DoS attacks, and found that in 32% of the cases the target organization's firewall and IPS devices were the main bottlenecks. As high as this number sounds, it should not surprise security experts who understand the nature of DoS and DDoS attacks and how firewalls are designed.

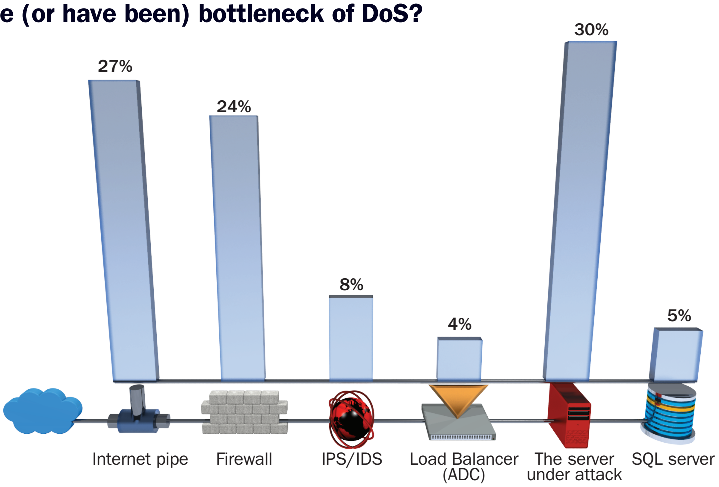
Firewalls are stateful devices, meaning they keep track of the status of all network connections that they inspect. All such connections are stored in a connection table, and every packet is matched against that connection table to verify that it is being transmitted over an established legitimate connection. The connection table of a standard enterprise-class firewall can store tens of thousands of active connections, and this is sufficient for normal network activity. However, during a DDoS attack, an attacker will send thousands of packets per second to the target's network.

In the absence of a dedicated anti-DoS device to shield the firewall from such a high volume of traffic, the firewall itself is usually the first

[10] Neustar Insight – DDoS Survey Q1 2012
[11] Radware 2011 Global Application and Network Security Report

device in an organization's network to handle the force of the DDoS attack. Because of the way a firewall is designed, it will open a new connection in its connection table for each malicious packet, resulting in the exhaustion of the connection table in a very short period of time. Once a firewall's connection table has reached its maximum capacity, it will not allow additional connections to be opened, ultimately blocking legitimate users from establishing connections, and subsequently preventing such users from accessing the online services hosted by the target network's server or servers. Not so strangely – in this scenario – a denial of condition was still achieved despite the presence of a firewall.

Radware Security Survey: Which services or network elements are (or have been) bottleneck of DoS?



Challenges in DDoS Attack Mitigation

There are several reasons why DDoS attacks are often hard to detect and mitigate. In many of the possible attack scenarios, each individual "malicious" packet is by itself a legitimate transaction – not something that would cause any harm to the online service or organization's network infrastructure. Legitimate transactions as simple as requesting a Web page can be abused by performing them so frequently that the server runs out of resources in an attempt to satisfy every one of the potentially thousands of requests per second per machine. Additionally, because each computer in a DDoS attack often possesses a unique IP address and attempts to make each of its thousands of requests using a different forged IP address and different header information, it can be difficult to identify and block a single attack source.

One particularly simple but ineffective technique used to mitigate DDoS attacks is the use of a **rate limit rule**. By setting a limit on the maximum amount of traffic that can flow to a Web server from the Internet (and refusing to accept the rest of the traffic), one introduces the issue of potentially refusing legitimate traffic. If a user attempts to connect to a server that has reached the maximum level of traffic allowed by its rate limit rule, he or she will be refused a connection despite his or her non-malicious intentions. Since rate limit rules do not distinguish between legitimate and illegitimate users, they are usually not very useful for DDoS attack mitigation, especially in the face of the "Slashdot effect" – when a popular Website links to a smaller site, causing a temporary massive increase in traffic or "flash crowd" on the smaller site.

Another strategy that DDoS attackers use to strengthen their attacks is the sending of out-of-state packets – **TCP packets** that are sent out of normal sequential order as defined by the TCP protocol. By sending packets out of order (that is, an ACK packet before a SYN-ACK packet), the attacker forces his or her target's machine to maintain information on this malicious connection in its connection table. As previously described, most devices cannot handle storing an excessively large number of connections in their connection tables without malfunctioning. To compensate for this, more advanced dedicated anti-DDoS solutions utilize sophisticated techniques to identify whether or not a packet is out-of-state, and activate mitigation mechanisms to block traffic based on such abnormal packet flows.

As attackers use not only volumetric attacks but also **"low and slow" attacks**, special mitigation strategies are required to deal with such attacks, as they involve apparently legitimate traffic arriving at a seemingly legitimate, albeit slow, rate. Tools such as Slowloris and R.U.D.Y. produce legitimate packets at a slow rate, allowing attacks carried out using them to pass traditional mitigation strategies undetected. One possible way to detect such an attack is to perform network behavioral analysis on the network during periods of normal operation, and compare such data to that gathered during a time of attack by a "low and slow" tool. For example, if on one particular application it takes on average five minutes and ten HTTP sessions to complete a transaction if a user spends five hours and requires 1,000 HTTP sessions to complete the same transaction they might be an attacker and further security measures are required.

Yet another sophisticated attack method abuses vulnerability in **Secure Socket Layer (SSL)**, a common method of Web encryption used in the HTTPS protocol. By forcing repeated encryption and decryption of data, particularly through the use of SSL's "renegotiation" feature, an attacker can completely occupy a target server's resources so it is not able to satisfy legitimate requests. SSL-based DoS attacks are particularly difficult to detect and mitigate as all traffic to the server is encrypted, and therefore must be decrypted – which is often a time- and resource-intensive process – before it can be determined to be legitimate or malicious and subsequently handled.

How to Deploy a DDoS Defense Strategy

The aforementioned challenges are only some of the many that security solutions providers face today when it comes to mitigating the latest and most complex DoS and DDoS attacks. It is clear that traditional security solutions such as firewalls and IPSs cannot provide an effective solution for DoS and DDoS attacks alone – organizations are urged to search for an attack mitigation system that can provide dedicated and more comprehensive protection from DoS and DDoS attacks.

Organizations have two primary choices when it comes to implementing a DDoS defense strategy: buy an anti-DoS service from a security provider or deploy an on-site attack mitigation system. We believe that organizations should not choose between these two alternatives, but rather adopt both, as they are complementary to one another.

Purchasing an Anti-DoS Service from a Security Provider

With a recent rise in the number of DDoS attacks, many Internet Service Providers (ISPs) and Managed Security Service Providers (MSSPs) have begun to offer anti-DDoS services. Such services protect organizations from network flood attacks by deploying mitigation equipment at the ISP or MSSP, just before their connection point to the organization. Often referred to as "clean pipe", this type of mitigation is guaranteed to block network flood attacks from ever reaching the organization, as attacks are mitigated before they ever reach the connection between the ISP or MSSP and the organization. This renders the organization's "internet pipe" free of malicious traffic.

Organizations that only deploy mitigation equipment on-site, however, can run into problems trying to mitigate the more massive network floods that saturate their entire “internet pipe”, which is why the anti-DDoS services are helpful. On the other hand, anti-DDoS services cannot block application DoS attacks as well as low and slow attacks, since their mitigation equipment is not sensitive enough to detect the intricacies of such attacks. Using both types of protection together can therefore shield your organization more effectively from both volumetric and application level DoS attacks.

Deploying an On-Premises Attack Mitigation System

To successfully detect and mitigate application-layer DDoS attacks such as HTTP and HTTPS floods or low and slow attacks, organizations should consider deploying on-site mitigation systems. Systems that are deployed in an organization’s data center provide perimeter security for the entire network infrastructure within the data center, specifically for any online services provided through servers located within the data center. Mitigation systems deployed in such proximity to the applications they are designed to protect can be fine-tuned to have a greater awareness to changes in network traffic flows in and out of the application servers, and therefore have a greater chance of detecting suspicious traffic on the application layer.

Recommendations

On-site attack mitigation systems can provide comprehensive mitigation for all sorts of application-specific attacks, but will fail to provide adequate protection against massive network floods that completely saturate an organization’s Internet pipe. That is why we recommend that organizations deploy both an on-site attack mitigation system as well as a cloud-based anti-DoS solution. The following table summarizes the different attack types, and where these attacks are more likely to be mitigated.

Attack Type	Cloud Mitigation	On-Site Mitigation
Network Flood blocking the internet pipe	●	
Application Flood		●
Low & Slow Attack		●
SSL Based Attack		●

Table 1: A summary of the mitigation capabilities offered by each defense strategy

Key Requirement Checklist for an DDoS Attack Mitigation System

In order for any attack mitigation system to detect and mitigate various types of DDoS attacks successfully, you should expect it to contain several basic features:

☑ **The Ability to Detect and Mitigate Both Known and Unknown Attack Vectors**

With the rapid introduction of new attack tools and methods, attack mitigation systems should be able to mitigate attacks using both known and emerging attack vectors. Hackers release attack tools employing new attack vectors on a daily basis, and so it is nearly impossible to arm a mitigation system with a database that contains information on every emerging attack tool. It is possible however, for a mitigation system to detect the impact of a new attack vector on normal network activity and generate a real-time signature as an attack using a previously unknown attack vector occurs, effectively blocking it as it happens. The use of both a legacy static signature-based system as well as a newer advanced real-time signature-based system allows for the mitigation of attacks using both known and unknown attack vectors – the most comprehensive solution.

☑ **The Ability to Analyze User Activity and Detect Misbehavior**

As previously discussed, many DoS and DDoS tools generate legitimate-looking network traffic that can still cause a denial-of-service condition when sent repeatedly en masse. For example, if a user attempts to abuse the previously described SSL renegotiation vulnerability, an attack mitigation system should detect that the repeated renegotiation of an SSL key is not normal user behavior. By comparing such suspicious activity with that gathered during network behavioral analysis, an attack mitigation system can block misbehavior, preventing the repeated SSL key renegotiation from consuming the target server's resources and ultimately causing a denial-of-service condition.

√ The Ability to Eliminate False Positives

An advanced attack mitigation system must be able to distinguish between legitimate users and malicious users, never flagging a legitimate user as malicious (false positive), or a malicious user as legitimate (false negative). A false positive situation results in the denial of service for legitimate users, significantly reducing the quality of experience for both an organization and its customers, while a false negative situation may allow a malicious user to perform additional cyber attacks without being detected.

There are several methods by which an advanced attack mitigation system can accurately identify the traffic of malicious users, including network behavior analysis (described in the previous section) and a challenge-response (C/R) mechanism. C/R mechanisms are designed to check whether a request to an online service has arrived from a real user with a real Web browser and PC, or a malicious user who has attempted to spoof such information with automated requests to make his or her requests seem real. In order to use a C/R mechanism, an attack mitigation system launches a series of queries to the source of a request in question, and according to the subsequent response it receives from the source, decides between two actions: sending a more sophisticated challenge, or flagging the source as a malicious user. C/R mechanisms are automatic processes that require no human intervention on both the attack mitigation system and the source sides, making them convenient and efficient as a defense mechanism. The intelligent usage of a C/R mechanism and network behavioral analysis can almost completely eliminate false positives, guaranteeing an excellent quality of experience for legitimate users.

√ The Ability to Mitigate Floods with Dedicated Hardware

The final important requirement for an attack mitigation system is the use of the proper hardware. Mitigation devices should implement dedicated hardware accelerator cards that can handle massive traffic floods, as it is important that a large amount of malicious traffic does not impact the performance of other mechanisms within the device. This could cause various components within the device to malfunction, ultimately not providing adequate protection against attacks.

DDoS Attack Vulnerability Assessment – 11 Questions to Ask Yourself

Knowledge is the foundation to any company's attack mitigation strategy for defending enterprise networks and applications. When it comes to security, what you don't know can hurt you. This vulnerability assessment is designed provide you with an overview of your organizations' security strengths and weaknesses. It can be a valuable indicator for areas to plan for additional training, continuing education, or professional certification. If you're not sure of the answers to any of these questions, you may be more vulnerable than you think.

- ? Does our business rely on high availability of revenue-generating of web applications?
- ? Would our company's reputation be diminished by negative publicity caused by availability issues?
- ? What's the hourly / daily cost of downtime to my organization?
- ? What is my organizations' defense strategy against DDoS attacks?
- ? How long would it take for DDoS attack detection and notification?
- ? What would I do if my organization experienced a DDoS attack tomorrow?
- ? Do we have an automatic DDoS attack response in place?
- ? How many times have we experienced attacks within the last year?
- ? Which of my infrastructure devices is most likely to fail during an attack on our business' availability?

? What is the best solution to remedy an attack while keeping the organization 100% available?

? What is our organizations’ ability to launch a counter measure against hackers and other cyber criminals?

Looking Forward

Over the next few years, Radware expects DDoS attacks to increase in sophistication, frequency and persistence.

First, powerful DoS and DDoS attacks will increasingly take advantage of the encrypted SSL traffic, targeting firms that depend on secured online transactions such as financial institutions, government agencies, social networking companies and others. Any organization that relies on SSL-based traffic without a proper decryption engine working in sync with an attack mitigation solution is exposing itself to great risk.

Security companies must also strive to develop new techniques to deal with the increase of low and slow attacks. The ease in which these attacks are launched and the destruction they can cause encourages hackers to develop more sophisticated low and slow attack tools to use in these attacks.

We anticipate attackers to become more persistent and more focused on their victims. During the past 12 months, we see a trend that attack campaigns last longer, and that attackers change their attack methods during the campaign in order to penetrate organizations’ security systems and to eliminate the online presence of their targets. Some attacks during 2012 lasted more than 3 weeks with constant attack methods that were changed by the attackers.

Attackers no longer launch random DDoS attacks on various targets; today, and more so in the future, attackers choose their targets carefully, perform preliminary scans to find security holes, choose the most painful timeframe to launch the attack, and keep it persistent for many days.

9 Conclusion

Imagine you woke up one day to hear a national broadcast on all TV channels announcing a hacker team's intention to disrupt the nation's transportation systems and power grids. Many cities' electrical systems have already been disabled, all major stock exchanges have been shut down, and all law enforcement computers and computer networks are malfunctioning.

Does this sound like the apocalypse? Perhaps some form of futuristic cyber warfare? This is, of course, a hypothetical scenario – it describes some of the events that occurred in the 2007 movie "Live Free or Die Hard", in which a series of cyber terrorists attempted to launch a complex multi-part cyber attack on the United States. With the increasing integration of computers and computer networks into everyday devices, the probability of such an attack occurring is not so astronomical any longer, as people's data is stored in more forms and in more places than ever before.

In the famous Confidentiality, Integrity, and Availability "Security Triangle", DDoS attacks target availability, preventing legitimate users from accessing the services provided by a targeted network device. There are numerous motivations for such attacks, ranging from fun to financial extortion, political protest, and even warfare. Those attempting to carry out attacks are not necessarily highly skilled hackers, as many tools have been developed that allow even the least experienced users to perform complex attacks.

In this handbook, we have tried to demonstrate that any business, large or small, that is dependent on Internet traffic to generate sales, service its customers, or maintain confidentiality is a candidate for stepped up protection against DoS and DDoS attacks for their network systems. No business or industry should consider itself completely safe from such attacks, as a failure to maintain defensive measures can result in severe financial and reputational consequences.

Companies that have deployed security solutions such as firewalls, IPSs and antivirus software may be well-protected against some types of security threats, but such solutions do not provide protection against DDoS attacks. In order to defend itself against DDoS attacks effectively, an organization should be aware of who its enemies are, what motivates them, and what tools they use. They need to deploy DDoS protection on multiple layers – bandwidth protection at their ISP, as well as application protection on-site. A combination of comprehensive knowledge, adequate DDoS protection systems, and a healthy sense of paranoia provide an organization with the best insurance against a DDoS attack.

For More Information

Want to stay ahead in the fight against DDoS attacks? Please visit: www.ddoswarriors.com for additional expert resources and information.

About the Authors

Radware (NASDAQ: RDWR), is a global leader of application delivery and application security solutions for virtual and cloud data centers. Its award-winning solutions portfolio delivers full resilience for business-critical applications, maximum IT efficiency, and complete business agility. Radware's solutions empower more than 10,000 enterprise and carrier customers worldwide to adapt to market challenges quickly, maintain business continuity and achieve maximum productivity while keeping costs down. For more information, please visit www.radware.com.

Radware' Emergency Response Team (ERT) is an emergency service with dedicated specialists that can respond in real time offering proactive, "hands-on" participation by security and product experts to mitigate active threat. Our longstanding relationships and reputation as a trusted advisor and solution partner make this guide possible. Our ERT has extensive experience handling attacks 'in the wild' as they occur.

Radware's ERT gives real-time assistance to customers under DoS/DDoS attacks. They do this by directly accessing the customer's network equipment, capturing the files, analyzing the situation and discussing the situation with the customer. Although the main intention of the service is to stop the attack and help the customer recover, the team also gets a unique view of the attack. Due to their hands-on involvement, they get real-time information regarding what the attack actually looks like. They are able to actually measure the impact caused by the attack. In other words, ERT has an in-depth perspective of what really happens when a website is attacked. Generally, the ERT is only called upon to respond when it is a medium to high grade attack campaign.

Contributors

Ronen Kenig
Director, Security Product Marketing
Radware

Deborah Manor
Security Product Marketing Manager
Radware

Ziv Gadot
SOC/ERT Team Leader
Radware

Daniel Trauner
Security Technical Writer
Radware

radware