Distributed Denial of Service (DDoS) Attacks: The Big Threat in the War on Downtime

An article by CAI Networks, Inc.
(August 13, 2007)

The army has been equipped for battle as the commander scopes out his target. Primed for attack, the troops include ... your mom? The green grocer across town?? A playwright in London??? A veterinarian from Chicago????

The commander is a 15-year-old student from Baltimore with a wallet fatter than most of his high school classmates. With the push of a button, he launches an attack that within seconds will start to cause thousands of casualties.

What's going on here?

Some of today's most feared and hard to defend against threats are "botnet" armies: hundreds of thousands of PCs that have been infected with specially-designed malware – all under the control of evil puppet masters, and generally without the knowledge of their unsuspecting owners. These armies of infected "zombie" computers are capable of wreaking havoc, and they are multiplying at a rate so staggering that would make any recruiter's head spin.

The platoons of these zombie armies are comprised of users with Internet-connected PCs with exposed vulnerabilities. You probably know some of them: friends and relatives who call us "IT professionals" when their PC is "acting strange", starts rebooting itself, or the occasional nuisance pop-up turns into a chronic flurry. But unlike malware designed to display ads or crash PCs, well-behaved botnet malware lives quietly in the background – waiting for the attack signal from its commander.

You run a decent company: it cares about quality and service, looks after its customers, supports the community – but you're preventing an unscrupulous competitor from gaining the market share he wants. He wants: your customers. And one way to try to get them is to make their online experience with your website a frustrating experience – so frustrating that they will abandon your company and do business with his.

Welcome to the age of the contract electronic "hit", where for a fee you have a website crippled.

Too bad to be true?

If this all sounds like a bad fantasy, we need to look back no further than the year 2000, when several large websites – including Amazon.com, CNN.com, eBay and E-Trade – were victims of some of the earliest DDoS attacks. These companies were unprepared, and the resulting disruption of operations made headlines and cost millions.

Most will remember the MyDoom virus, which in 2004 spread via email and infected an estimated one million computers that launched a DDoS attack against the software company SCO. SCO moved to a new domain and removed its old one from the DNS and managed to avoid major damage. A variant of MyDoom targeted Microsoft, but did

not spread widely and so Microsoft's pre-emptive alternate domain shift was not required and its website remained up. A later MyDoom variant attacked Google, AltaVista, and Lycos, completely disabling Google for more than half a day and slowing the others down for hours.

At the time of the writing of this article, a worm named Storm had infected an estimated 1.7 million computers over a three month period, creating a massive zombie army capable of not only taking down the websites and flooding the bandwidth of any company but many countries. Unlike MyDoom, which had its target IP addresses hard-coded, giving its intended victims time to prepare, Storm is capable of receiving its attack coordinates dynamically, so any site can be a put into its crosshairs at any time.

The Denial of Service attack

So-called "denial of service", or "DoS attacks" are electronic assaults intended to make a company or organizations outward-facing computing resources unavailable to users. They generally strive to prevent a website from functioning efficiently or at all, effectively disabling its owner's web presence. DoS attacks may try to deprive service temporarily or even indefinitely.

DoS attacks typically do harm by causing a target system to reset or saturating it with external communication requests so fast and so furious that it cannot respond to legitimate traffic, or that it responds so slowly that users time out or give up in frustration. DDoS attacks can gobble up all available bandwidth so that networks are unable to carry any traffic other than that imposed by the attack, or cause network devices to lock up or crash.

The user experience during a denial of service attack is familiar to anyone who has tried to access a very popular website at a very popular moment: you either cannot get through or it's infuriatingly slow ... and then maybe it times out. In fact, it is not unusual for website operators to create their own inadvertent denial-of-service events by failing to provide sufficient capacity to handle peak user activity. (The foreword-thinking companies and organizations that anticipate and plan for such situations is the reason we're in business.)

To be clear, a DoS attack is not intended to compromise security, steal data or money, or cause the kinds of problems that other kinds of assaults can – but the consequences are often the same: loss of business, revenue, customers, market confidence, etc. And like other types of assaults, there is the potential of the attacker holding a gun to your head asking to be compensated to not do it or make it stop, and positing that the cost of calling off an attack is small compared with the potential damage it could cause.

The Distributed Denial of Service attack

Denial-of-service attacks are nothing new: they have been around for years, and effective methods have been developed to deal with them. What's new is the increased effectiveness of such attacks, brought about by the ease of coordinating multiple attackers in a simultaneous assault. Such "distributed denial of service" or "DDoS" attacks can be vastly more effective not only because the volume of the attack can be so much greater but because the mechanisms generally used to detect and stop such attacks is quite a bit more complicated.

A distributed-denial-of-service attack is rather the same as a lot of users trying to access a website at the same time: like trying to get onto CNN.com or bbcnews.com after a huge news event, like 9-11. This was not an intentional attack, but the conditions and the consequences were the same: too much traffic for the website to handle, resulting in poor performance or a crash.

Detecting and stopping denial-of-service attacks

In this article, we will not go into detail about how DoS and DDoS attacks actually perpetrate their evils. It is not important for this discussion, and we do not want to give clues to hackers (although such information can easily be readily found on the Internet). It is sufficient to know the following:

- DoS attacks are normally characterized by a single automated attacker sending transactions much faster than any human user could, and often trying to exploit a host-side vulnerability that would cause the impact of the bombardment to be heightened.
- Detecting and stopping a DoS attack involves determining the IP address that is sending all the traffic and blocking it (and reporting it to authorities: DoS attacks are illegal).
- DDoS attacks are similar to DoS attacks except because there are multiple attackers they do not necessarily need to exploit vulnerabilities: the sheer volume of normal transactional traffic that can be generated could be sufficient to bring down a website or flood a network that lacks the capacity to handle it.
- DDoS attacks do not need to generate very substantial transaction volumes from any client, since the same – and much more – volume of traffic can be generated by all the attacking computers in concert.
- Detecting and stopping DDoS attacks cannot simply pinpoint and block a single IP address since there are a multitude of IP addresses involved in the attack, and few if any of them may be generating an inordinate amount of traffic.
- Since DDoS attacks are done from a multitude of computers, it is very difficult if not impossible to identify the Commander in Chief of the attack. (The MyDoom author, believed to be a Russian programmer, is today still at large despite rewards in the hundreds of thousands offered for his unmasking.)

For both DoS and DDoS attacks, time is of the essence in determining that an attack is underway and stopping it. If too late, depending on the nature of the attack, it could be difficult if not impossible to get sufficient system control to slow down or stop the attack without a system shutdown.

Preventing denial-of-service attacks

The best protection against denial-of-service attacks of any flavor is a strong defense. This first of all means closing any holes an attacker could exploit. This, of course, cannot include closing off the channel to the Internet, since without that there is no service to be denied; rather, it means following best practices in setup and configuration and correcting any known vulnerabilities. But even with no vulnerabilities available for an attacker to exploit, there still remains the fact that a large enough volume of traffic –

as would be the norm for a very busy website – could bring a website to its knees, or worse.

A DDoS attack in particular poses the greatest threat, since via a substantial botnet with each zombie sending traffic to the website, the multiplying effect will be much more pronounced than a DoS attack. This means that a rapid response is often not enough – the response must be immediate.

As stated, early detection and corrective action when an attack hits are imperative. This means that detection needs to be done by an intelligent mechanism and, ideally, the same mechanism immediately takes corrective action – or is in close communication with the correcting agent.

Without detailing the nature of the mechanisms that detect and correct for DoS and DDoS attacks, it is obvious that all traffic to a website needs to be monitored to identify attack traffic, since legitimate and malicious traffic may look the same. This means that a mechanism able to monitor all incoming traffic to a website is required, and that it operates at high speed. This mechanism should also be capable of blocking malicious traffic while allowing legitimate traffic to pass through, ideally unimpeded.

WebMux's DoS and DDoS protection

It was when one of our customers fell victim to a DDoS attack that we implemented DDoS protection in its installed WebMux device, to augment WebMux's existing DoS and other types of protection and security capabilities. This company was already mindful of the need for sufficient capacity to handle spikes in user traffic, which is why it deployed multiple web servers with a WebMux load balancer in front of them to distribute the traffic.

We were all a bit skeptical – us, the customer, their ISP, and their security consultants – that they were actually being attacked: after all, they were not the type of organization that one would think would warrant such an assault. But the data did not lie, and the WebMux DDoS protection solution was successful in defending against the attack. (The attack did not stop, but its negative impact on the company's website was stopped.)

WebMux is a standalone network appliance that connects to the network, in between the external and internal firewalls, or using its own firewalling capability. All incoming traffic passes through the WebMux, where it is automatically routed to the appropriate web server, either agnostically or intelligently based on content (URL, MIME header, cookie, etc.). So WebMux is a natural place to put DoS and DDoS protection since it is the gatekeeper for all inbound traffic.

With a built-in detection and correction mechanism to handle DDoS attacks in addition to its existing DoS attack protection, WebMux is the only server load balancer in its class to offer this critical functionality.

WebMux users with DDoS protection (available as a free firmware upgrade for incumbent customers under support) can add this need-to-have safeguard to the long list of high availability and high performance capabilities WebMux provides.

The Storm is on the horizon, the armies are massing, and the clock is ticking.

The checkbooks are open, and so are the wallets.

WebMux is guarding the border.

Are you protected?