Ethical Hacking and Cyber Security Report: The GitHub DDoS Attack of 2018

Abstract

GitHub, the code hosting platform, was the target of one of the largest memcached DDoS attacks in history but managed to detect and mitigate the attack. This paper will detail how difficult this was to do, and how memcached attacks are some of the largest growing types of DDoS assault. Lastly, why the attacks are usually so effective will be detailed, and how GitHub is planning to be even more resilient to these types of attacks in the future.

Introduction

GitHub is one of the largest version control-based code repositories available, and they handle millions of push and pull requests every day. In 2017, they repelled one of the largest DDOS attacks in history, resulting in only a few minutes of downtime, compared to several hours for many other DDOS attacks. While GitHub survived this attack, it was only through intensive preparation and dedication to protecting their users.

This paper will first describe the nature of the attack and how it is used to take down websites, including how normal DDoS attacks worked and how these compare to memcached variants. Secondly, a description of how GitHub detected and mitigated the effects of the attack will be shown, including the steps GitHub will be taking in the future to mitigate further attacks. Lastly, a collection of other ways that attacks like these can be prevented or mitigated will be detailed.

What is a Memcached DDOS attack?

Firstly, DDoS stands for Distributed-Denial of Service, and the main purpose of them is to completely take up the maximum bandwidth of a website, in order to take that website down and prevent other users from interacting with the website. Due to this, websites that suffer from DDoS attacks can lose a large amount of revenue, as they can be taken down for several hours or more (IDG, 2019.). Websites can also suffer in other ways, such as a loss of reputation from their website being unusable; many users may switch to a competitor in the hopes that the competitor will be more reliable when needed.

Normal DDoS attacks use botnets to accomplish their goals. A botnet is a vast network of computers that perform the same task at the same time. However, most of the computers in many botnets are unwilling participants, victims of Trojan horse viruses that take up a portion of the computers processing power to perform tasks such as DDoS attacks. The owners of these computers also suffer when a DDoS attack is performed, as much of the computer's processing power is consumed by the botnet, slowing down their computer into an almost unusable state.

The Memcached DDoS attack however, operates very differently from the traditional DDoS attack. Memcached is a type of server introduced in 2003, and by itself is not a threat (Singh, K. and Singh, A., 2018.). Instead of using vast botnets to achieve the required processing power to take up an entire websites bandwidth, they use a technique to massively amplify data requests. Normally, requests to a website go through a UDP (User-Datagram-Protocol) Server that prepares a response, which is then sent to the target website. The website servers then send back a packet, which contains the information that was originally requested.

All of this is normal procedure, and in an average DDoS attack, the only difference is the original machine is instead thousands of machines sending requests all at the same time, in order to overwhelm the target server, with each UDP Server in between doing its job as normal, without much changing. The main difference between this type of attack and a memcached one, is that the memcached attacks forges their original requests to the UDP Server, in such a way that the UDP Server sends much larger packets that the original forged ones to the target server, amplifying the potential attack size by many times. These types of attacks are only permissible on certain UDP servers that haven't filtered this type of attack, so attackers usually use IP Spoofing in order to access those vulnerable UDP Servers. The IP Spoofing make it even harder to later figure out where the attacks originated from.

If owners of UDP servers are aware of this type of attack, they should limit their UDP responses to make sure they are smaller in file size than the original requests. This should outright stop memcached amplification attempts. According to the GitHub briefing about the attack, there are over 100,000 servers in the world that are vulnerable to this type of attack. A very concerning facet of this type of assault is the fact that memcached attacks are far easier to manage than botnet traditional DDoS attacks for the criminals.

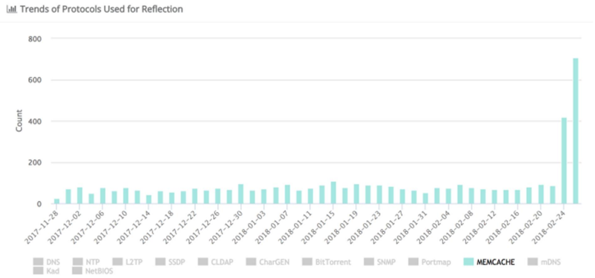

Figure 1: Graph showing frequency of memcached attacks between November 2017 and March 2018 (Cloudflare, 2018)

One of the reasons for this is that there is no need to maintain and manage a gigantic botnet of computers in order to perform the procedure, whereas many traditional DDoS attempts require vast amounts of time and processing power to establish such a botnet and keep it under control. With memcached attacks, there is no need for this; just a few dedicated computers are required, as the amplification effect allows for much smaller packets to be sent for a much larger reward of attack size. Memcached attacks have been growing very rapidly, shown in Figure 1, and this was proven when GitHub was attacked in 2018.

The GitHub Attack

A shown in the section above, memcached DDoS attacks can be some of the most damaging ways to take down a website. However, GitHub was prepared for this type of attack and many other types as well. GitHub is a version control code sharing website and is used by over 65 million developers worldwide (Preston-Werner, T., et al, 2008). The main purpose of GitHub is to allow multiple developers to work on the same project at the same time. This is done by keeping a central codebase online that individual developers can push and pull files to and from. By using GitHub to manage the changes in files and to ensure compatibility between files, GitHub allows millions of software projects to undergo in record time, ensuring that no time is lost when looking for compatibility between changes to codebases.

Due to the nature of GitHub, the website having any form of downtime can cause irreparable damage to many of the software projects that are hosted on the website. Most projects on the website are constantly being altered, and if the server cannot be contacted then all those projects will grind to a halt. Therefore, it is so beneficial for GitHub to prepare for these occurrences, which fortunately was the case in this assault.

In order to mitigate the memcached assault, GitHub first used multiple detection systems that would wait until the incoming traffic was large enough to warrant help. The system would then signal another system, named Akamai, which would help process the requests and provide additional temporary bandwidth. This attempt was a success, as the Akamai system allowed GitHub to process all the requests as normal, resulting in only a few minutes of downtime, which would not likely have even been noticed by much of the GitHub community.

This method of preventing the damage of the attack was mostly made possible by the fact that GitHub was so prepared for large DDoS attacks, but another determining factor is that GitHub is so large of a company that they can provide such a large maximum bandwidth. Most companies encountering this type of assault would certainly fail to withstand it, so the fact that GitHub was attacked instead was fortunate. However, this means that all companies in the future that host online websites should prepare for this type of assault, perhaps utilizing backup bandwidth providers that can help when further processing power is required. More possible techniques will be detailed in the next section.

How do companies prevent further attacks?

The method that GitHub used was largely possible due to the size of GitHub itself, so many other companies cannot rely on just having a larger bandwidth than the attackers can supply. One such method is to prevent the attacks at the source, spreading the word about just how many of these unprotected UDP Servers that there are and educating business owners that do not prevent this sort of attack utilizing their UDP Servers.

There are some very simple steps that developers can take when working with basic memcached servers that can completely prevent memcached DDoS attacks. For example, disabling the UDP port manually will prevent nearly all of these attacks, as UDP is known as an outdated way to deliver web service. UDP ports are automatically enabled on all servers, so manually preventing them from being used may completely prevent these assaults in the future. However, getting this to work for the over 100,000 vulnerable servers out there will be a very difficult task. Furthermore, the internet listening port 11211 is used by many of these assaults, ass it is UDP by default, even though switching to TCP (Transmission-Control-Protocol) will vastly improve defenses for the server.

Conclusion

For many companies in the world, DDoS attacks hold a very significant threat. They come with the possibility of losing customers, revenue, and even reputation. Preventing these threats as much as possible should be a high priority for server managers. GitHub was able to do this by spreading the load to a temporary boosting system that allowed them to carry on working minutes after the initial DDoS attack. Preventing these attacks can be as simple as turning off UDP type ports, to as difficult as finding every vulnerable memcached server and convincing its owner to help prevent these types of attacks. Lastly, every company should do its best at guarding against all types of DDoS attacks, no matter how big, as their customers should be allowed to use their website without interruption.

References

Cloudflare, 2018. Recent memcached attacks against Cloudflare Servers. [image] Available at: <https://blog.cloudflare.com/content/images/2018/02/memcached-ddosmon.png> [Accessed 17 May 2021].

IDG, 2019. IDG DDoS Report: Evolving Strategies for Handling Today’s Complex and Costly Threats. [online] IDG. Available at: <https://www.a10networks.com/marketing-comms/reports/evolving-strategies-for-handling-todays-complex-and-costly-threats/> [Accessed 20 May 2021].

Majkowski, M., 2018. Memcrashed - Major amplification attacks from UDP port 11211. [online] The Cloudflare Blog. Available at: <https://blog.cloudflare.com/memcrashed-major-amplification-attacks-from-port-11211/> [Accessed 14 May 2021].

Preston-Werner, T., et al, 2008. GitHub: Where the world builds software. [online] GitHub. Available at: <https://github.com/> [Accessed 16 May 2021].

Osanaiye, O., 2015. Short Paper: IP spoofing detection for preventing DDoS attack in Cloud Computing. 2015 18th International Conference on Intelligence in Next Generation Networks, [online] Available at: <https://ieeexplore.ieee.org/document/7073820> [Accessed 20 May 2021]

Singh, K. and Singh, A., 2018. Memcached DDoS Exploits: Operations, Vulnerabilities, Preventions and Mitigations. 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS), [online] [Accessed 20 May 2021]