The Unstoppable Force: An Analysis on DDOS Attacks

Marco Carvallo, Sean English, Tyler Smith
Florida State University
Team 1

Abstract

Today we use the internet for almost everything. We rely on the services we find online in order to complete the tasks in our daily lives. Because of this information security has really been growing rapidly in the last decade or so. New defence methods are created every day to manage online threats, but there is one threat that to this day causes major problems for even some of the strongest systems in the world. A DDOS attack is an attack simple enough for almost anyone to perform, but it is extremely difficult to defend against. In this paper we will be analyzing this seemingly unstoppable force. We will be looking at how it functions in a safe laboratory environment, how it functions and operates as a threat, as well as how you can prevent it and defend a service from this kind of attack.

Keywords: *DDOS, Information Security, Apache, pfSense, Security Onion*

Introduction

The internet is an ingrained and important part of our daily lives that requires more protection than most realize. The internet can be said to run on a series of networks of varying sizes, yet all of these are constantly under pressure from malicious attacks and threats. One such attack is known as a ddos attack. The ddos attack is a dangerous malicious attack that can cripple any network. It is imperative that we understand it in order to prevent it. The purpose of this research paper is to explore the uses of a ddos attack in a virtual environment against other networks, understanding its inner workings, and finally how to defend against this attack. We will achieve this by first going over the initial lab systems settings and setup to give you an understanding of the environment we're working in. This will entail talking about each feature and service inside of the virtual network and how we are utilizing it. This will be followed by a graphical representation of its topology. Following this section, we will then go over ddos attacks themselves. We will do this by first giving you a basic definition of what a DDOS attack is, the types of this attack, and how easy it is to use and its effectiveness. We will then connect this to our virtual environment and describe how we plan on demonstrating how this attack works. Finally, we will cover how a network can defend against the attack. This includes suggestions for defending a large scale network (or a small scale network. Defending against this attack also entails suggestions for mitigation and prevention. With these ideas and concepts in mind), we hope to fulfill the purpose of this research paper and support our initial statement about the dangers of DDOS attacks. Now, onto some of the pieces of literature and their respective topics that are used in this paper.

Literature Review

Network Protection Mechanisms

An important part of any working business is ensuring that your network has the best protection. While our virtual lab environment didn't have the choice of choosing specific types of servers and firewalls, we were able to do research on the concept to see possible faults our network may encounter. The Security Onion was one of the focal points of our research. The paper "Logging and Monitoring to Detect Network Intrusions and Compliance Violations in the Environment" was a great article that discussed the different roles that Security Onion fills. Security Onion is a intrusion detection system that is also a network security monitor. It contains programs such as Sguil, Snorby, Squert, Snort, argus, and many more. The article goes into detail about how all of these features come into play with a network and provides a topology diagram to show what it is like in a basic environment.

Finally, the article goes into detail on how to set up the features and shows how they work with pictures and walkthroughs (Gupta, 2012).

Firewalls are just as important as monitoring software in a network. While we weren't able to choose the firewall, there were online materials that could help us compare the pfSense firewall to other popular ones on the market. The Firewall journal article went over a variety of firewalls and discussed their features. This allowed us to compare the information of the pfSense firewall with those that are commonly used in the industry to help identify possible vulnerabilities or points of concern. It should be noted that this article is from 1997, so its information wasn't taken as up-to-date(Sobol, 1997).

DDOS prevention, detection, and response

This entry focuses on classifying DDOS attacks, preventing them, and how to detect them. According to the journal there five different classifications for the type of ddos attacks. Network device level, OS level, application level, data flood, and protocol feature attack. Network device level attacks aim to exhaust the hardware using exploits such as a password buffer overflow. OS level attacks involve slowing down the actual system which can be done by sending ping attacks which affect the OS. Application level attacks aim to slow down network applications or involve the attacker using applications on the host to slow the system down. Data flooding attacks are the most common and the aim is to take up as much bandwidth to prevent the network from sending and receiving packets. Protocol feature attacks aim to use an exploit in the protocol, an example is using a spoof attack to hide yourself.

According to the journal, the purpose of the DDOS attack is to temporarily disable the system instead of taking it out completely. Because internet resources are limited, if the attacker has more bandwidth than the victim, the victim is definitely going to be taken out. DDOS attacks have four elements: the attacker, handler of attackers, zombie hosts, and the target. The attacker can also be a handler which uses the zombie hosts to attack the target (Douligeris & Mitrokotsa, 2004). The steps taken in a DDOS attack are as followed: 1. Selection of zombie hosts, 2. Zombie hosts get compromised, 3. Attacker communicates with the handler to notify them of zombie hosts acquired, 4. the attack commences.

When it comes to DDOS defense there are four categories: intrusion prevention, intrusion detection, intrusion tolerance and mitigation, and intrusion response. There are three locations that defenses can be deployed: victim network, intermediate network, and the source network. In order to prevent DDOS attacks you need to prepare your network by setting up several compliances such as : using globally coordinated filters(ingress filtering, egress filtering, route-based distributed packet filtering, history-based ip filtering, and secure overlay services), disabling unused services, applying security patches, changing ip addresses, disabling ip broadcasts, load balancing, and the use of honeypot. Anomaly detection and misuse detection are used for intrusion prevention to check if the user is a genuine user. There's not much that can be done to respond to an attack besides looking at logs and using IP trackback to find the source of the attack. According to the journal, "research on intrusion tolerance accepts that it is impossible to prevent or stop DDOS completely and focuses on minimizing the attack impact on maximizing the quality of its services"

At the victim network level there is little that can be done since most of it's resources are being attacked. The only thing the victim network can do is recognize that it is getting attacked. The intermediate network level's purpose is to trace back and identify the attacker. The source network is responsible for stopping incoming traffic during the attack but you are limiting all traffic which compromises availability. There is little that can be done to stop a DDOS attack other than limit your own network (Douligeris & Mitrokotsa, 2004).

A Review of DDOS Attacks and its Countermeasures in TCP Based Networks

This section covers various methods of executing a DDOS attack and how they different types of attacks function. Over all this article covers SYN flood attacks, TCP Rest Attacks, ICMP attacks, UDP storm attacks, DNS request attacks, CGI request attack, Mail bomb attacks, ARP storm attacks, Algorithmic complexity attacks, and Spam attacks. Some of these attacks like SYN flood attack, and UDP storm function by overburdening a system's

resources.while others like the TCP reset attack function by interfering with the transmission of data to and from the service.

From there the article goes on to explain tools that can be used to issue these attacks. Namely they used Good Bye v3 with a TOR plugin. using this combination of tools they are not only able to launch a DDOS attack but they can also spoof the ip of the attacking machine. Next the article explains possible defence mechanisms that can be implemented. To simplify, they go over the importance of being able to identify an attack and the use of a bloom filter. Prevention is you best defence since stopping an active attack is very difficult. Through careful monitoring and auditing attacks can be avoided and attackers can even be traced back. Finding the attacker and stopping the attack at the source holds the highest chance of success. If that is not possible a bloom filter will help protect your system by managing transmission in a more organized manner.

Finally the author goes into a Individual Component Analysis. while going into the exact explanations offered by the article would take up the better side of two pages, a general explanation of this section is a more detailed way to analyze the patterns of traffic.

Lab Systems Settings and Description

We performed our experiments in a laboratory environment inside of a virtual machine network. This virtual machine network was comprised of a Apache 2.2 server with the Ubuntu distribution, a Windows 7 machine running HoneyBOT, a Windows 7 machine running a firewall called Comodo, a pfSense firewall, a defensive VM running Security Onion 12.04, and finally a offensive VM running Kali Linux. The Security Onion and Kali Linux machines would not be the targets of the attacks in our exercise. Our main priorities in terms of protection was the Windows 7 machine that was also running HoneyBOT and the Apache 2.2 server. We will now briefly explain the basic function and setup of each of those parts.

Security Onion

Security Onion's main purpose is network security. It has three main features that it uses to give its users this. These are intrusion detection, network security monitoring, and log management. In order for Security Onion to perform, it combines the use of a few applications. In this lab environment the main applications that will be used are known as Sguil, Squert, Snorby, and finally ELSA. Sguil's main purpose is recording and capturing network traffic in the form of events, information from sessions, and then packet captures. This a visual representation of the data and information flowing through a network, which leds to proper decision making. The next application is known as Squert. Where Sguil captures the data, Squert helps present it. Basically Squert utilizes the data collected and morphs into something that is easier to read. It also allows the user to better extract the data from Sguil through queries. Then we have Snorby. Snorby, like Squert, helps extract and display information captured through other systems (such as Sguil). However, the main difference is that Snorby is a front end web application and it can also interact with systems other then Sguil, like Snort. The final application we have is called ELSA, which stands for Enterprise Log Search and Archive. When monitoring a network, the amount of data being stored can become cumbersome to search through, especially when it can reach such large quantities. That is where ELSA comes in. It helps its users search through logs so that they can much more easily find the information they need in a timely fashion. (REF: https://www.sans.org/reading-room/whitepapers/detection/logging-monitoring-detect-network-intrusions-compliance-violations-environment-33985)

Kali Linux

Where Security Onion is our eyes, Kali Linux is our sword. Kali Linux is a linux distribution that is formed with penetration testing in mind. It sports a wide variety of tools that can be used in the offensive aspect. These can include tools such as the exploit search/usage of Armitage, the reconnaissance capabilities of Maltego, and many more. With regards to our laboratory exercise, we will(INDICATE if this has been done) mainly use Kali Linux's

reconnaissance to map out our potential targets and it will be our forerunner in launching offensive attacks against other teams (which we will explain in a further section).

pfSense

The next part of our setup is the shield. The pfSense application is the main foundation for our shield. It sports a wide variety of capabilities that can be used, including dynamic DNS, wireless access points, IPv4/6 support, NAT, inbound load balance, and many others. However, its most important feature is its firewall capabilities. This will be the major feature we will be taking advantage of in our lab environment. pfSense will serve as the frontline defense against the outside world to protect our information assets. We will configure the assets and the firewall in such a way to ensure that no unauthorized actions can be taken against the network from outside the firewall or from inside the firewall.

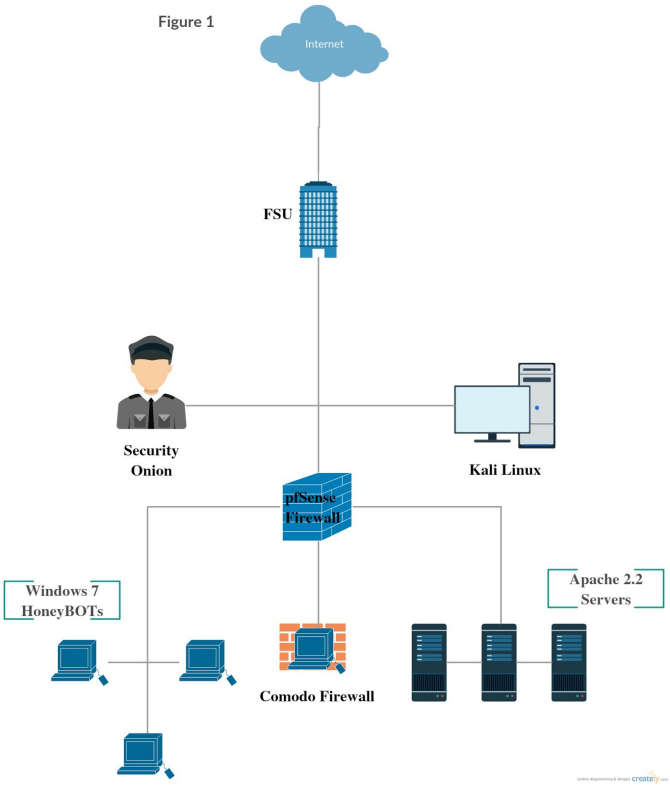
Comodo

Comodo is another part of the shield of our network. It is firewall that is focused on filtering out client/workstation based activities. It can do things such as whitelisting trusted publishers of files, controlling which applications can run, personalized prevention rules, behavior analysis, and even malware/virus protection. We will be taking advantage of all of these features to ensure our Windows 7 information asset is protected to the best of our abilities. Comodo has come under fire recently for various security issues. The two most recent ones are the GeekBuddy VNC server vulnerability and the Web Security Vulnerability. The GeekBuddy vulnerability that was brought to light a week ago automatically installs a VNC server and enables it (tav..., 703, 2016), Because of this we have chosen to remove GeekBuddy from the computer. The Web Security vulnerability that was found about a month ago was primarily dealing with the browser that Comodo provides, so as a precaution we didn't utilize the browser (https://code.google.com/p/google-security-research/issues/detail?id=704). For precautionary measures, we will be ensuring that neither of these services are installed/used for the remainder of the exercise.

Apache

Apache is one of our main assets. It represents our web services and is the server for them to run on. In a business setting, it would be imperative to keep this operational and maintained. In the laboratory environment, each group member had an apache server which they ran and maintained. This means we have 3 total Apache servers to protect. These Apache servers are of configuration 2.2 and run on Ubuntu linux distribution.

Figure 1



Windows 7/HoneyBOT

Finally, we have information assets in the form of Windows 7 work stations. Similar to the Apache setup, each group member has been assigned a Windows 7 machine. This means we have a total of 6 information assets we need to protect. Three Windows 7 machines and three Apache servers. Part of the defensive measures to protect the network is setting up HoneyBOT. HoneyBOT is a honeypot which acts as a distraction for any attackers, making them think they're attacking the real asset when in fact they're attacking a fake one. It is sort of like bait on a fishing line.

Topology

Now that we have explained the features of our network, let us discuss the topology. First, the pfSense firewall will be the main gateway into our network kingdom. It is from here that all inbound and outbound connections will pass

through. As a result of this choke point, it is one of the major security checkpoints to ensure our kingdom stays safe. As a result, we have put all of our main assets behind this file and set it as the default gateway. So a total of 3 Apache 2.2 servers, 3 Windows 7 HoneyBOTs, and our Windows 7 Comodo are behind this firewall. Outside of our firewall we have our remaining two services, the Security Onion and Kali Linux. The reason that these services are not behind the firewall is because they will not the the targets of attacks. Figure 1 demonstrates a graphical topology of our network.

DDOS Attacks

DDOS (distributed denial of service) attacks are one of the few attacks that can't be stopped. Even today companies such as Microsoft have had problems in the past with DDOS attacks. A DDOS attack uses zombie (which are usually infected with Trojans) hosts to attack a single target to disrupt a service. A group of these "zombies" are called a botnet. The most common DDOS attacks are bandwidth attacks and application attacks. Bandwidth attacks are used to disrupt network bandwidth by flooding the system with packets (TCP, UDP, and ICMP). Application attacks involve using HTTP errors to overload the system.

Threat type

To really understand a DDOS (distributed denial of service) attack we must first cover what a DOS (Denial of service) attack is. A DOS attack is defined on Wikipedia as "an attack that attempts to make a machine or network resource unavailable to its intended users, such as to temporarily or indefinitely interrupt or suspend services of a host connected to the Internet." While this covers the base concept of this attack it really does not explain it fully. DOS attacks are used to stop the operating function of a webpage, server, or other computing resource by interfering with the transmission of information to or from the service. A very simple metaphor to help understand most of these attacks is to imagine a horse that can only carry 100 pounds down a road. It takes this horse one hour to do this. If you were to come along and load that horse with 200 pounds, it would walk much slower. With all the extra weight it would take the horse two hours now to reach the end of the road. If we were to then put a huge amount of weight on the horse, say 1000 pounds, the horse would be too encumbered and would not be able to walk at all no matter how much time was allotted. A DOS attack follows essentially the same logic. By sending a huge amount of requests or transmissions we can encumber a service to the point that it does not function, or functions so slowly that it is of no use to its users.

The issue with a DOS attack is that if you for some reason cannot encumber the target, the attack will fail. Sticking with our horse metaphor, if someone has a really strong horse that can carry 2000 pounds, and you only have 1000 pounds to weigh it down with, the horse will still be able to move around just fine. Stronger entities like large companies (Google, Microsoft, etc.) will have stronger systems thusly making these attacks ineffective. One person on one computer will never be able to successfully launch a DOS attack on Google.com. This is where DDOS comes into play. While a single person on a single computer can't do much, a single person controlling 10, 20, 100, even 1000 machines can do a lot of damage. Referring back to our super strong horse, even though it can carry 2000 pounds and one person can only load 1000 pounds onto it at a time, with a group of less than five people you could entirely encumber it. Attackers use this logic to effectively take down larger, more powerful targets. According to Akash Mittal (2011) "The current attacks on trendy web sites like Amazon, Yahoo, e-Bay and Microsoft and their resultant disruption of services have uncovered the weakness of the Internet to Distributed Denial of Service (DDoS) attacks."By using malicious software, such as Trojans, attackers are able to gain control over various computers depending on how they choose to infect their victims. Once a machine is infected an attacker uses this system (usually against the owners will) to help attack the target. Each of these infected machines is known as a Zombie and the collection of these infected machines is known as a botnet. With a large enough botnet it is theoretically possible to take down any target because you simply need to reach the threshold of what that system can handle.

Aside from increasing the computing power, controlling this many computers can help in other ways as well. One of the most common ways to stop a DDOS attack is to simply block all requests and transmissions coming from the source that is attacking you. This can be identified by a few different means but most common way is by looking at the attackers IP address. With only one source this is a very effective way to weaken the attack, but with say twenty

or more machines this becomes much more difficult. Large scale attacks have been known to take over hundreds, even thousands of computers making systematically blocking each one a long and difficult process. This also makes tracking the attacker much more difficult. To keep themselves hidden, attackers have been known to attack with their botnet but not with their own machine. This is roughly equivalent to not leaving your figure prints at a crime scene. By not directly involving the main machine, it makes the attack untraceable unless they traced back through one of their zombies. This is also a bit of a double edged sword. Legally someone cannot be arrested or charged with any offence in the United States unless their attack affects more than 10 machines or causes more than $5000 dollars in damages. So while a larger botnet may make an attacker more powerful, the price for failure is much higher the larger you grow.

DDOS also offers a special quality that other attacks often don't have. It is commonly very hard to tell on more active or popular web pages if you are being attacked, or if you are simply receiving a large amount of traffic. Because most DDOS attacks simulate a service being overloaded by its users, it is easy to mistake high network traffic for an attack and vice-versa. This poses an issue because, as we mentioned before, one of the main ways to stop an attack like this is to block the source from transmitting requests to the service. The danger is that you do not want to block real, authentic, users from accessing the service. In turn, this disguise causes it to be extremely tricky to stop this attack without accidently affecting the legitimate users negatively.

There are two attacks to worry about when defending your system. A volumetric attack and a application level attack. A volumetric attack is very hard to defend against as you're dealing with a botnet but an application level attack uses some sort of open port to take down a service. The most common is a volumetric attack because a host isn't always running services that can be exploited.Volumetric attacks might also use a form of a reflective attack that would spoof their IP, but there are safeguards that can be put up to decrease the likelihood of a successful DDoS attack.

With our Kali Linux's wide range of scanning and exploit tools, it is the perfect platform for what we have in mind. More specifically we are going to be using the tools nmap and Ettercap. Nmap is a network mapping tool that we will be able to use in order to scan the network for our opposing teams. In the scan we will be given information like other systems IP addresses, what ports they have open and other important networking information that we will need in order to successfully crash their system. Once we have found a main point of weakness we will begin our attack with Ettercap. Ettercap is a network security tool that is capable of doing a number of nasty attacks, the most noted is a MITM (Man in the middle) attack. This attack is where an attacker inserts themselves into the information stream of a service, effectively acting as the router as far as the computer sees. The computer sends information to the attacker now instead of the router, and the attacker relays that information to the router after stealing the data they need from it. Using this line of code, "ettercap -T -q -i eth0 -M arp /victim ip///" with victims IP replaced with the Ip address of the target system, we are able to launch a special type of DDOS attack. To simplify the code it in essence tells Ettercap to begin a MITM attack but instead of passing the packets along to the router, it drops them. This in effect denies service to the user that is under attack.

While this attack is effective and meets all technical definitions of a DOS attack, it does really fit our "overburdening" model. Service is still being denied but instead of the service being overloaded, the information being sent and received is simply lost in transition. While this does burnen the system with fail logs, the overall denial of service is due to the loss of data. Because of this, in order to see how different variation of this attack affect a system we will also be executing standard flooding DOS attack through Ettercap. This attack will use the Ettercap Graphical tool plugin to send a burdening amount of requests to a service. Logically though, because our opponents are using systems with equivalent computing power, we will have to use multiple machines (our three Kali Linux machines) in order to encumber and in the end crash a service. Between these two attacks, launched from our three machines we should be able to target an IP address and crash all of its services at will.

Defense

DDOS attacks are one of the easiest attacks to use and one of the hardest attacks to defend against. The issue with DDOS attacks is that if it's being attempted on a system that has a lot of users then the DDOS attack is hard to

defend against because you're not sure if this overwhelming traffic is genuine. When on a home network it is very easy to defend against DDOS attacks because you should be able to monitor your traffic and distinguish a DDOS attack from regular traffic. The main focus is prevention because it is impossible to stop a DDoS attack and very hard to properly respond to such an attack. The only way to respond to a DDoS attack is to check your logs and to find patterns within the ip address to understand who and where the attack came from.

Some common techniques for preventing DDOS attacks (without outside help) include using a firewall which acts as a "chokepoint", route filtering techniques which involves using a remotely triggered black hole and putting all the DDOS traffic into this "dead zone", and unicast reverse path forwarding which determines the reachability of the connection and if it fails it rejects the packets. There are two disabling techniques which involve disabling unused services to prevent attacks from using these ports and disabling IP broadcasts prevents ICMP floods.. Load balancing relies on the ISP to allow you to use more bandwidth during an attack. The last resort is to change your IP address to dissociate yourself from the attack. Although it won't take long for the attacker to notice, a honeypot can be set up to distract an attacker long enough so you can be alerted of a DDOS attack giving you enough time to respond. (Douligeris & Mitrokotsa, 2004)

The problem with monitoring all the traffic is that you can't tell if the connection is genuine or part of a DDOS attack. Another way to distinguish the traffic is to look at the IP location and see if this botnet is operating in a common location and block all IPs from that location, although it can affect the availability of your network. An issue that arises is that many companies don't want to pay somebody to monitor the traffic because DDOS attacks don't happen often. A tool that is good for monitoring packet traffic is Wireshark. It runs on all operating systems and provides a lot of tools that can give you detailed reports and analysis of traffic. This is ideal for home networks because there isn't a lot of genuine traffic to be monitored, making a DDOS attack easy to spot. Theoretically speaking, the best way to defend against a DDOS attack is to keep up with the attack and keep blocking if you're fast enough.

If your business is unable to have an in-house security division then there are two options. Relying on multiple ISPs and using cloud mitigation services. Relying on multiple ISPs allows you to have a backup if one of the services your ISP provided is currently getting attacked which would allow you to mitigate all of this bad traffic to one of their servers dedicated to DDOS mitigation. The problem with this is that the ISPs don't communicate with each other which could be a problem if both networks are being affected by DDOS attacks. One of the problems

According to Nirav Shah(2014)brings up the point that there is no firewall protection against DDOS attacks as it causes a bottleneck in your overall performance. The nature of DDOS attacks involves disguising the traffic as wanted traffic in the ports that have to be open to prevent it from being blocked. Having all the traffic checked causes a major availability issue as it will take longer to receive packets. Malwarebytes is a free scanner tool that also blocks incoming packets if it doesn't recognize the source. When you first start it, it blocks all of your application ports to make sure they aren't malicious and to confirm with you that you want that port open. This is a great tool for the regular user who doesn't know much about IT security and it's also free.

The only real defense against DDOS is to prevent it from happening. There are many companies that are making money off this situation by providing products for DDOS attacks such as 3rd party mitigation services and scanners. This option is very expensive but is definitely the best protection. Letting a 3rd party deal with mitigations requires no security professionals to be on site of the network and the traffic gets mitigated as soon as it's detected. Radware has an IPS called the "DefensePro" that has a mitigation device but is very expensive. They also offer the service of attack mitigation that allows them to mitigate traffic once you have notified them of an attack, but it's best to use this paired with DefensePro so you have an IPS on site but this option is ideal if you don't have the capital for a DefensePro but want some sort of intrusion response.

One of the only things a firewall can do is detect and block incoming traffic during a DDOS attack. Unless it's one system commencing a DoS, it will be very hard to detect the zombie bot IPs that are attacking your system. You could block a range of IPs but you might block genuine traffic that will affect the availability of your network.

Conclusion

With all this in mind, we can quickly see that DDOS attacks are becoming a very serious threat in today's cyber world. With our virtual environment laboratory setup, we can see that even a bare minimum, highly protected network is still vulnerable to this attack. By examining the threat type earlier in the article, we found that DDOS attacks specialize in taking down servers and are often hard to counter. Using our Kali Linux machines, we were able to explore using this type of attack against other teams. Of course one of the most important things is learning to counter it, which we covered in the defense section. It was here that we found methods of utilizing firewall routing rules and "black holes" can help counter DDOS attacks. However, the biggest issue with this system is differentiating between genuine traffic and malicious traffic. Having a firewall chokepoint can also negatively impact your organization's ability to offer services to clients, depending on the situation. So with this information in mind, we found that the only viable solution to this attack is attempting to prevent it. This includes having your network be able to handle large amounts of traffic at once and/or having third party mitigations. When we compared this solution to our own virtual environment, we found it would simply be unfeasible. Our network is too small to handle large loads of traffic at once and our resources in the real world are too small to be able to afford third party mitigation. This can translate into real world scenarios where our small network can represent the many small companies in today's business environment and how susceptible they are to DDOS attacks.

Work Cited

Douligeris, C., & Mitrokotsa, A. (2004). DDoS attacks and defense mechanisms: Classification and state-of-the-art. Computer Networks, 44(5), 643-666. doi:http://dx.doi.org/10.1016/j.comnet.2003.10.003

Gupta, S. (2012). *Logging and monitoring to detect network intrusions and compliance violations in the environment.* (Research Paper). SANS Institute Reading Room: The SANS Institute.

Leach, S. (2013). Four ways to defend against DDoS attacks. Retrieved from *http*://www.networkworld.com/article/2170051/tech-primers/tech-primers-four-ways-to-defend-against-ddos-attacks.html

Mittal, Akash, Ajit Kumar Shrivastava, and Manish Manoria., 2016, "A Review of DDOS Attack and Its Countermeasures in TCP Based Networks." *International Journal of Computer Science & Engineering Survey IJCSES* 2.4 (2011): 177-87. http://airccse.org/journal/ijcses/papers/1111ijcses13.pdf

Shah, N. (2014). The DDos myth about the firewall and IPS. Retrieved from https://www.corero.com/blog/609-the-ddos-myth-about-the-firewall-and-the-ips.html

Sobol, M. I. (1997). Firewalls. *Information Systems Security, 6*(1), 20-28. doi:10.1080/10658989709342525

tav... (2016). Issue 703: comodo: Comodo internet security installs and starts a VNC server by default Retrieved from https://code.google.com/p/google-security-research/issues/detail?id=703

tav... (2016). Issue 704: comodo: Comodo "chromodo" browser disables same origin policy, effectively turning off web security. Retrieved from https://code.google.com/p/google-security-research/issues/detail?id=704