|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name: Nishchal Sreevathsa |  |  |  |  |

Performing a Denial-of-Service Attack from the WAN

Ethical Hacking & Lab 7

Table of Contents

1. [Executive Summary 2](#_Toc409378035)

[Highlights](#_Toc409378036)

[Objectives](#_Toc409378037)

1. [Lab Description Details 3](#_Toc409378038)

[Include Steps Taken, Notes, & Screen Shots demonstrating completion of lab objectives](#_Toc409378039)

1. [Supporting Evidence 15](#_Toc409378040)
2. [Conclusion & Wrap-Up 17](#_Toc409378041)

[Summary with observations, Success & Failures, Challenges](#_Toc409378042)

# Executive Summary

## Highlights

|  |  |
| --- | --- |
|  | This is a hands-on lab where you may practice running and analyzing fundamental denial-of-service attacks from a wide area network. The ideas, resources, and technique offered are beneficial for security education and penetration testing.   * The Low Orbit Ion Cannon (LOIC) tool is used in the lab to demonstrate TCP flood, UDP flood, and HTTP flood denial-of-service attacks. * On the Linux sniffer system, the attack traffic is captured using Tcpdump. * The capinfos command displays the number of packets captured statistics. * SYN packets are sent during a TCP flood to saturate the victim with open connections. * Random ports are flooded with UDP datagrams during the UDP flood. * The web server is overloaded with GET and POST requests due to the HTTP flood. * When "Wait for reply" is disabled in LOIC, the HTTP flood is more effective. * The assaults' main goal is to exhaust server resources and deny authorized users access to services. * For ethical hackers, traffic analysis and network sniffing are crucial skills. * Common denial-of-service attack tools and strategies are practiced in the lab. |

## Objectives

|  |  |
| --- | --- |
|  | This lab aims to practice different denial-of-service (DoS) attacks over a wide area network (WAN). The lab specifically addresses:   * Launching a DoS attack by use of a TCP flood. * Performing a DoS attack by means of a UDP flood. * Launching a DoS attack using an HTTP flood. |

# Lab Description Details

## Include Steps Taken, Notes, & Screen Shots demonstrating completion of lab objectives

**Perform TCP Flood**

**Step 1:** Launch the Internal Linux Sniffer machine which has no IP Address and after machine boots, enter the Username as **root** and Password as **toor** and Sign In into the machine.

**Step 2:** Launch the Linux terminal and find the IP Address of the machine using **the ifconfig** command. Using the command: **ifconfig > ip1.txt**, you can save the IP Address configuration. Copy the IP settings to a file, then use the **cat ip1.txt** command to open the file.

A screenshot of a computer

Description automatically generated

Note: Found the **Sample Flag:999818** for **ip2.txt** file. Discovered and captured the **flag: 123457** for **ip3.txt** file using the command: **cat ip3.txt**.

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

**Step 3:** Type **# ifconfig eth0 0.0.0.0 up** to prevent the system from having an IP address. Then cross verify for eth0 no IPV4 address should be listed using command: **ifconfig**.

A computer screen shot of a black screen

Description automatically generated

**Step 4:** Use the command: **tcpdump --help** to view all of tcpdump's available options. Start the tcpdump sniffing on the eth0 interface using the command: **tcpdump –i eth0 -nntttt -s 0 -w TCPcapture.cap**

**A screenshot of a computer

Description automatically generated**

**Step 5:** Launch the Windows 8.1 Attack Machine button with external IP Address 175.45.176.200 and then start the packet traffic utilizing the Low Orbit Cannon.

**Step 6:** Enter **203.0.113.100** for the Target IP and hit on Lock on button and the Target IP address appears on Selected target field, choose **TCP** from the Method dropdown menu, and then hit the **IMMA CHARGIN MAH LAZER** button. Count on **30 seconds.** then press the button marked **Stop flooding**.

**A screenshot of a computer

Description automatically generated**

**Step 7**: Return to the internal Linux sniffer and hit **Control+C** on the terminal to end the capture.

Using the command: **capinfos TCPcapture.cap,** we can determine how many packets are in total in the TCP capture file.

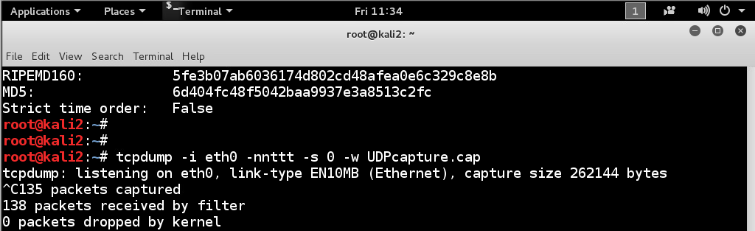
**A screenshot of a computer

Description automatically generated**

**Note:** In TCP Flooding, a total of **4890 k** packets were found.

**Perform UDP Flood**

**Step 8:** Apply the UDP technique to the attack now. Start the tcpdump sniffing on the eth0 interface using the command: **tcpdump –i eth0 -nntttt -s 0 -w UDPcapture.cap**

****

**Step 9:** Launch the Windows 8.1 Attack Machine button with external IP Address 175.45.176.200 and then start the packet traffic utilizing the Low Orbit Cannon.

**Step 10:** Enter **203.0.113.100** for the Target IP and hit on Lock on button and the Target IP address appears on Selected target field, choose **UDP** from the Method dropdown menu, and then hit the **IMMA CHARGIN MAH LAZER** button. Count on **30 seconds.** then press the button marked **Stop flooding**.

A screenshot of a computer

Description automatically generated

**Step 11**: Return to the internal Linux sniffer and hit **Control+C** on the terminal to end the capture.

Using the command: **capinfos UDPcapture.cap,** we can determine how many packets are in total in the UDP capture file.

A screenshot of a computer

Description automatically generated

**Note:** In UDP Flooding, a total of **135** packets were found.

**Perform HTTP Flood**

**Step 12:** Apply the HTTP technique to the attack now. Start the tcpdump sniffing on the eth0 interface using the command: **tcpdump –i eth0 -nntttt -s 0 -w HTTPcapture.cap**

**A screenshot of a computer

Description automatically generated**

**Step 13:** Launch the Windows 8.1 Attack Machine button with external IP Address 175.45.176.200 and then start the packet traffic utilizing the Low Orbit Cannon.

**Step 14:** Enter **203.0.113.100** for the Target IP and hit on Lock-on button and the Target IP address appears on the Selected target field, choose **HTTP** from the Method dropdown menu, and then hit the **IMMA CHARGIN MAH LAZER** button. Count on **30 seconds.** then press the button marked **Stop flooding**.

**A screenshot of a computer

Description automatically generated**

**Step 15**: Return to the internal Linux sniffer and hit **Control+C** on the terminal to end the capture.

Using the command: **capinfos HTTPcapture.cap,** we can determine how many packets are in total in the HTTP capture file.

A screenshot of a computer

Description automatically generated

**Note:** In HTTP Flooding, a total of **2101** packets were found.

**Step 16:** Start the tcpdump sniffing on the eth0 interface using the command: **tcpdump –i eth0 -nntttt -s 0 -w HTTP2capture.cap**

**A screenshot of a computer

Description automatically generated**

**Step 17:** Launch the Windows 8.1 Attack Machine button with external IP Address 175.45.176.200 and then start the packet traffic utilizing the Low Orbit Cannon.

**Step 18:** Enter **203.0.113.100** for the Target IP and hit on Lock-on button and the Target IP address appears on the Selected target field, choose **HTTP** from the Method dropdown menu, Unchecked the Wait for reply button next to HTTP, then hit the **IMMA CHARGIN MAH LAZER** button. Count on **30 seconds.** then press the button marked **Stop flooding**.

A screenshot of a computer

Description automatically generated

**Step 19:** Return to the internal Linux sniffer and hit **Control+C** on the terminal to end the capture.

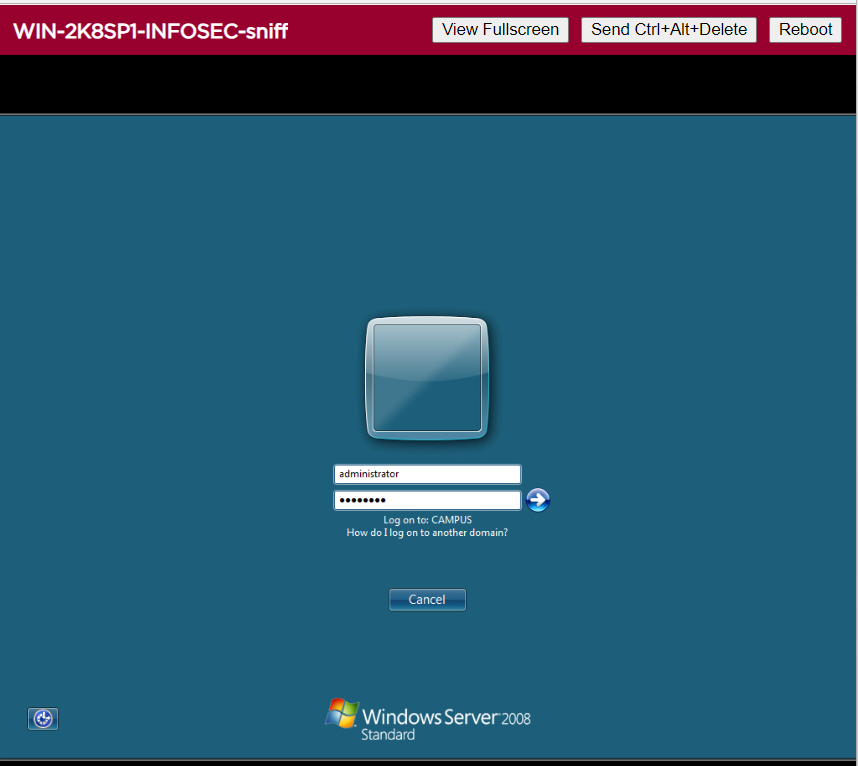
Using the command: **capinfos HTTP2capture.cap,** we can determine how many packets are in total in the HTTP2 capture file.

**A computer screen shot of a black screen

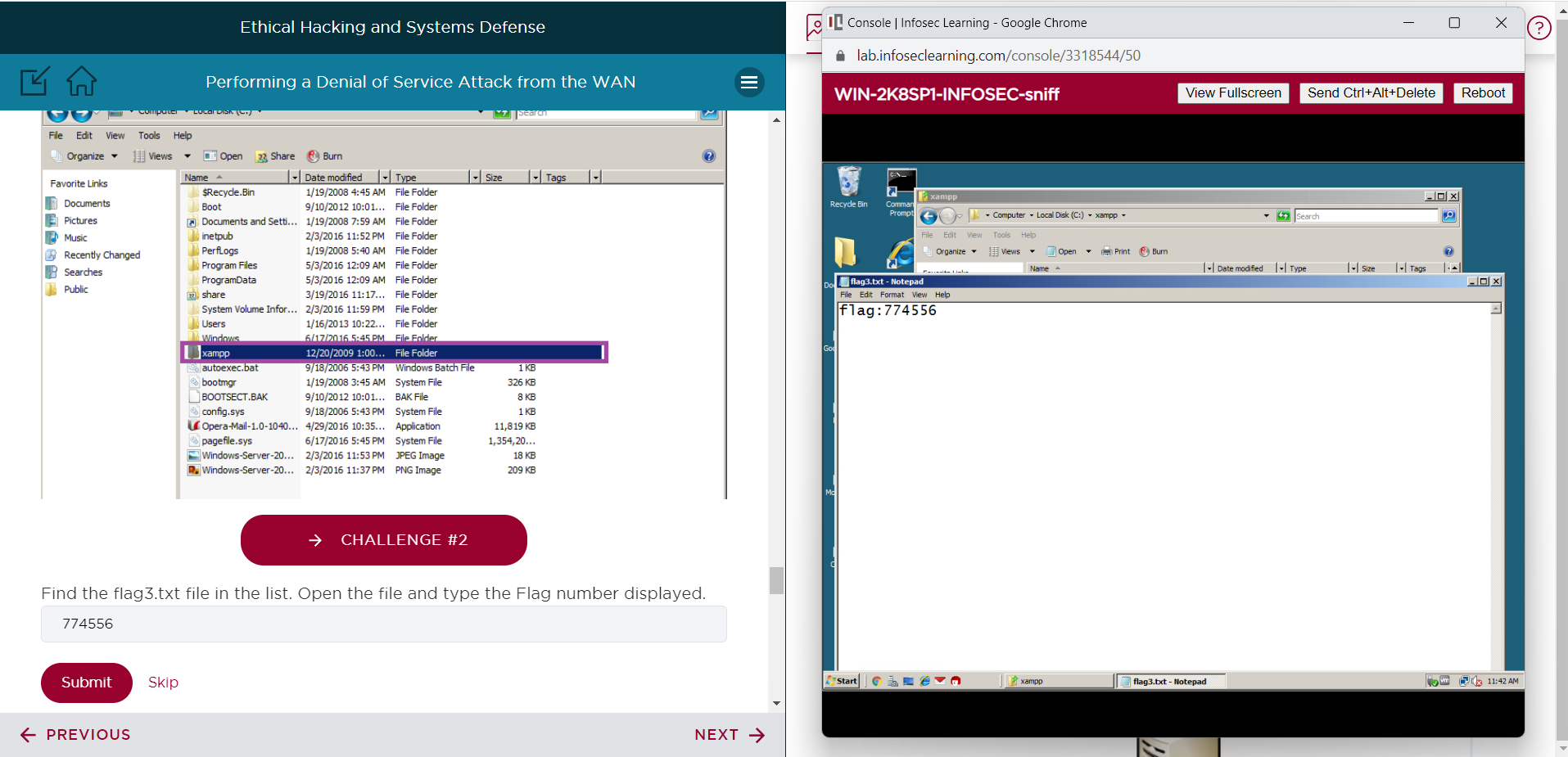
Description automatically generated**

**Note:** In HTTP2 Flooding, a total of **2297** packets were found.

**Step 20:** Launch the Internal Windows Server with IP address **192.168.1.10** and login with administratorcredentials.Username – **administrator** and Password as **P@ssw0rd**

****

**Step 21:** In search, click on computer and navigate to Local Disk (C:) drive and open xampp folder where flag3.txt is found and flag number is **774556.**

****

**Step 22:** Open the apache folder and search for flag4.txt and open theflag4.txt. **Flag Number is: 345678**.

A screenshot of a computer

Description automatically generated

**Step 23:** Navigate to logs folder and search for flag5.txt file and open the file. **Flag number: 818772**

**A screenshot of a computer

Description automatically generated**

**Step 24:** Open the access.log file and search for flag6. **Flag6 number: 445616**

Check out the entries that mention "A cat is fine, too."

**A screenshot of a computer

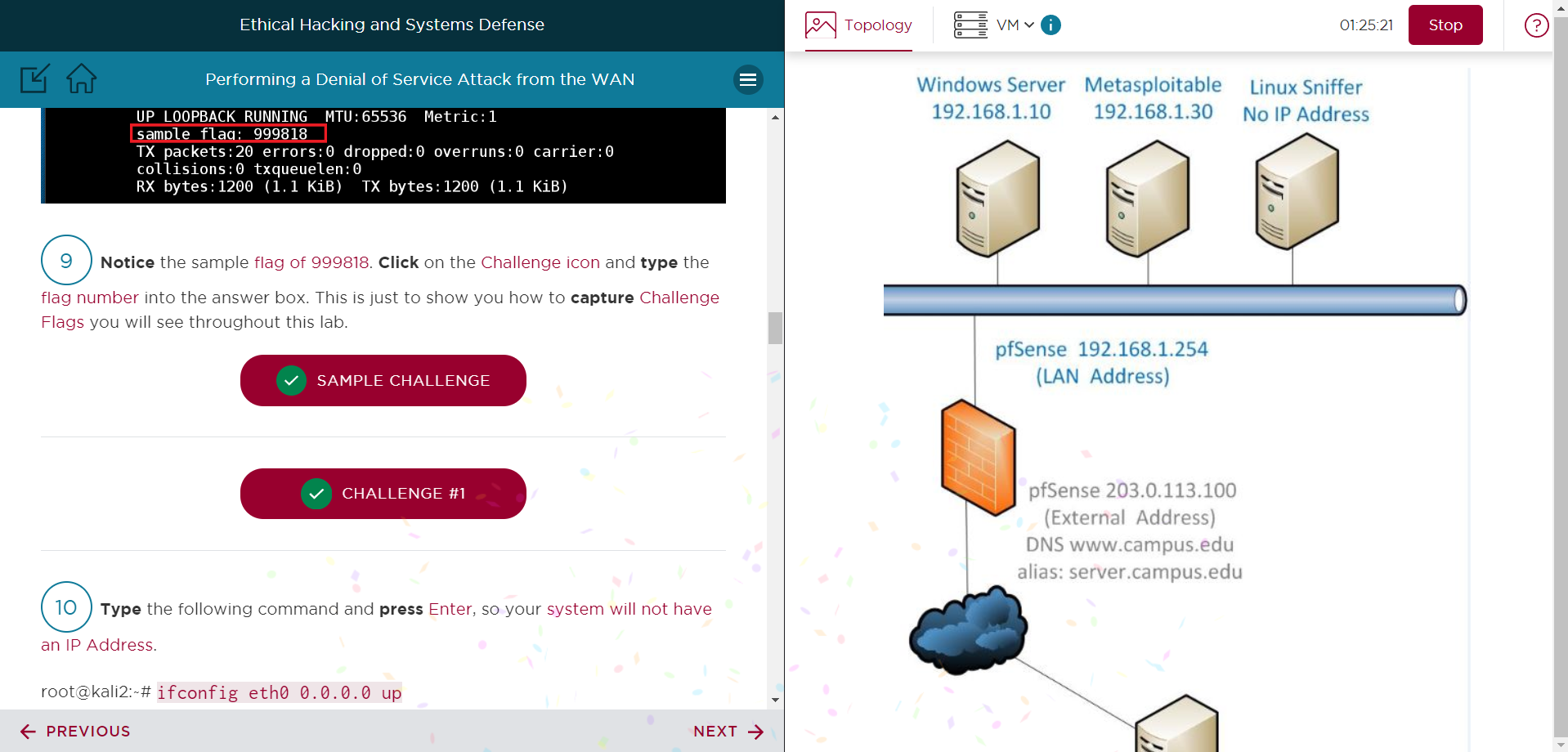
Description automatically generated**

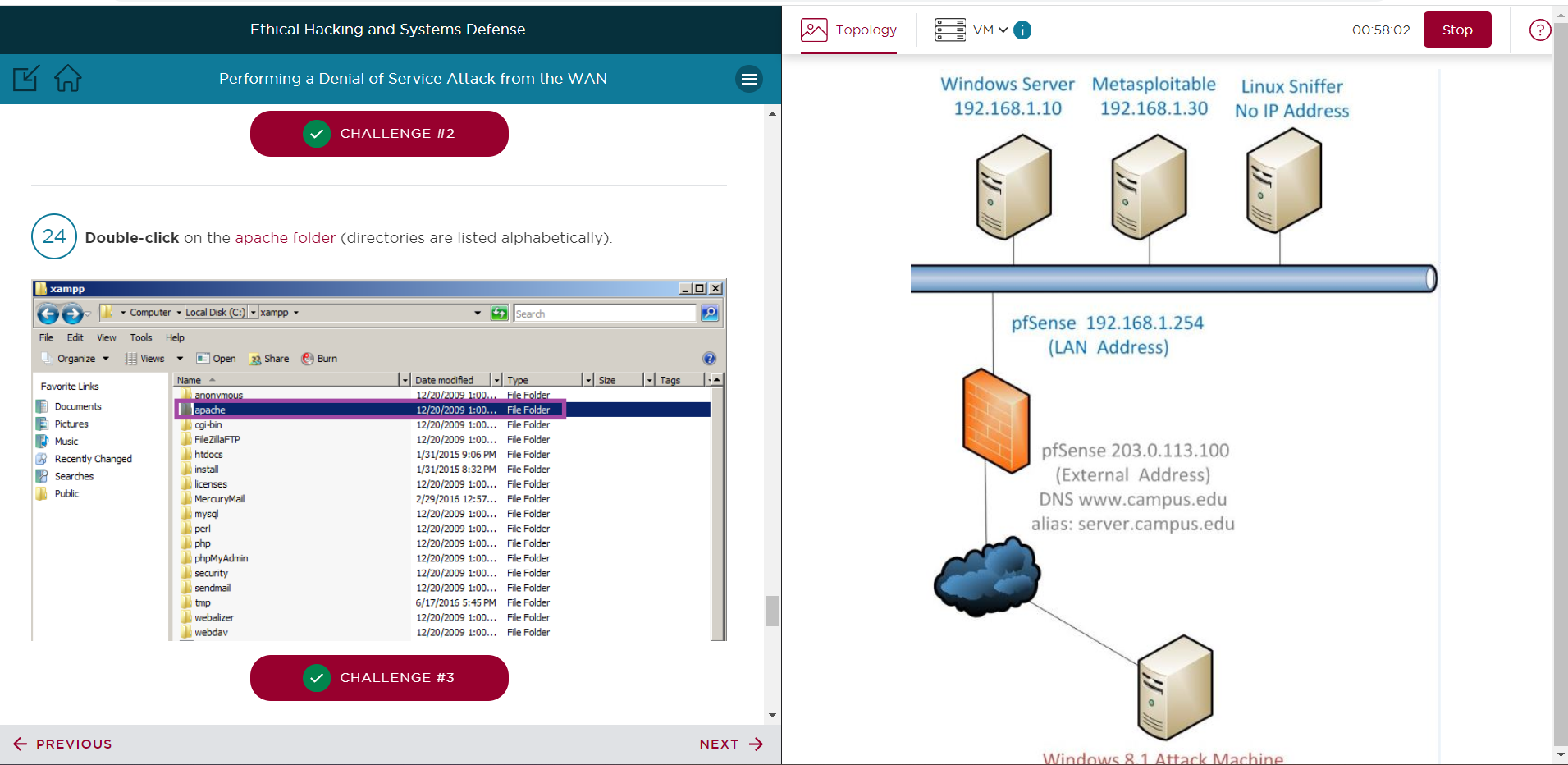
**A screenshot of a computer

Description automatically generated**

# Supporting Evidence

**Screenshots, Research, Etc.**

****

****

**A screenshot of a computer

Description automatically generated**

# Conclusion & Wrap-Up

## Summary with observations, Success & Failures, Challenges

In this lab, we used the Low Orbit Ion Cannon (LOIC) program to carry out denial-of-service assaults using a TCP SYN flood, a UDP flood, and an HTTP flood. You recorded the attack flow using tcpdump and examined the number of packets with capinfos. The assaults consumed resources, overloaded the victim server with requests, and rendered services unusable. This gave users practical experience with popular DoS attack tools and techniques via a WAN.

In conclusion, the lab was successful in achieving its goal of simulating various denial-of-service assaults from a WAN and evaluating their effects using packet captures. The principles and abilities learned can be applied to ethical hacking and penetration testing.

**Successes:**

* Successfully started the WAN-based TCP, UDP, and HTTP floods.
* Tcpdump was correctly configured to record the assault flow.
* Utilized capinfos packet counts to evaluate the effects of the attacks.

**Failures:**

* Until "Wait for reply" was removed in LOIC, the HTTP deluge was less effective.
* Did not validate a service failure as seen by the victim server.

**Challenges:**

* It took some time to comprehend how to use the LOIC interface.
* It took some time to figure out how to interpret the tcpdump output and filters.
* It's challenging to gauge the ideal attack duration that will still result in successful captures without crashing the victim.

**Risks:**

* **Disruption of the network:** DoS attacks may have a detrimental effect on the subnet's other hosts' availability or network performance.
* **Detection/tracing:** It's possible to identify and locate the source of the assault flow.
* **Machine crashes:** Attacks have the potential to destroy data and crash systems if they consume too much resources.
* **Escalation:** A perpetrator may use a denial-of-service assault to divert attention from more serious attacks.

**Remediation:**

* **Limit assault duration:** In order to obtain captures without upsetting hosts, just launch attacks momentarily.
* **Monitoring:** To identify excessive connections or spikes in bandwidth, enable logging and/or IDS.
* **Filtering:** To stop known DoS traffic patterns, use proxies, ACLs, etc.
* **Hardening:** To reduce the attack surface, disable unnecessary services and adhere to best practices.
* **Redundancy:** To lessen the impact of a single host failing, use load balancing and failover.