INFSCI 1600 Security and Privacy

Fall Semester 2021

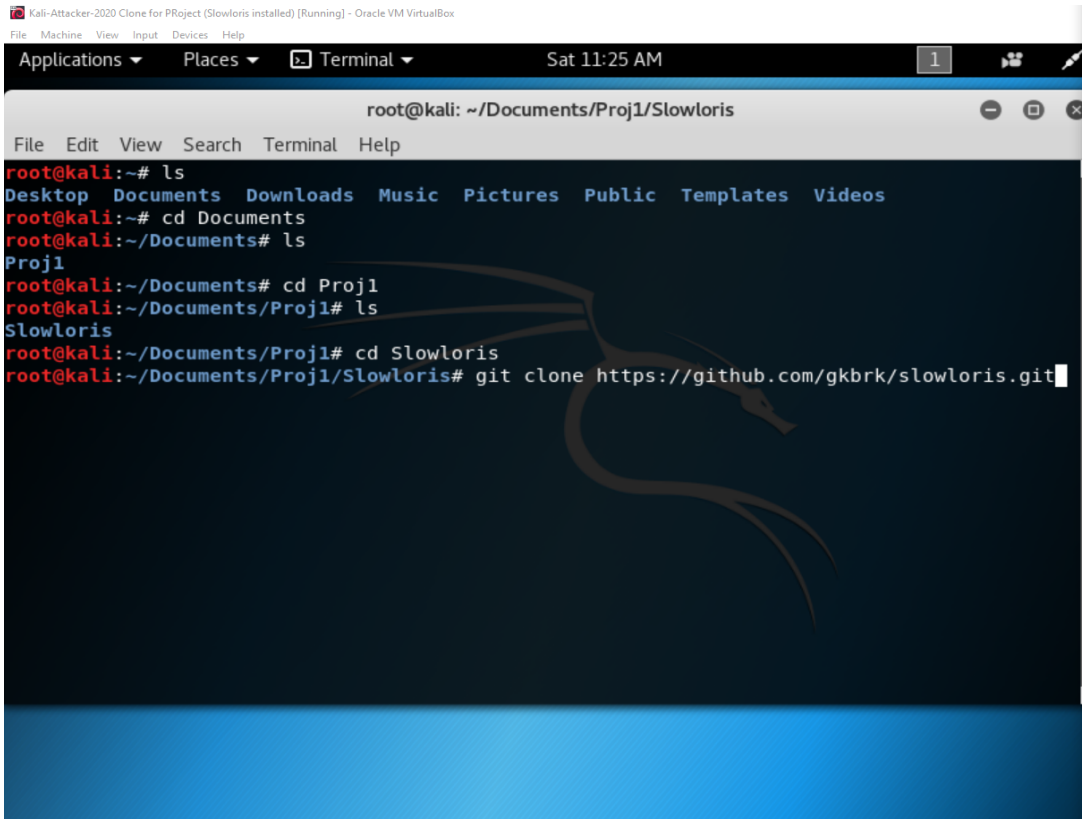
Jonathan Azcona, jka27@pitt.edu

5 October 2021

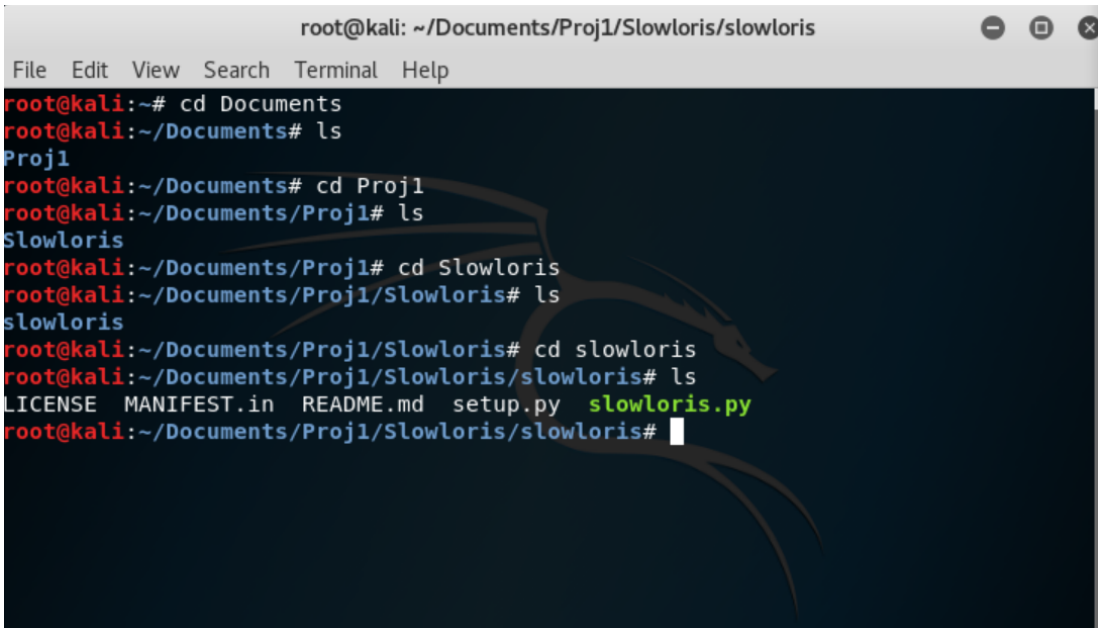
Project 1 - Distributed Denial of Service (DDoS)

At the start of the project, I decided to look up types of DDoS attacks and I fell onto Cloudflare's web page of DDoS attacks. One of the attacks listed is Slowloris, which is a common DDoS attack tool to launch a low and slow attack on a targeted server. Slowloris is classified as a low and slow attack tool under Cloudflare's category of types of DDoS/DoS attacks. I decided to choose slowloris as our DDoS attack tool on Initech for this project. In figure 1, I decided to make some directories and install the slowloris program. Next, I made the directories of Proj1 and Slowloris. Within the Slowloris directory, I installed the slowloris program from github using the command git clone with the link to the github repository. Figure 2 displays the contents of the slowloris github download.

(Figure 1: Directories made for Slowloris)

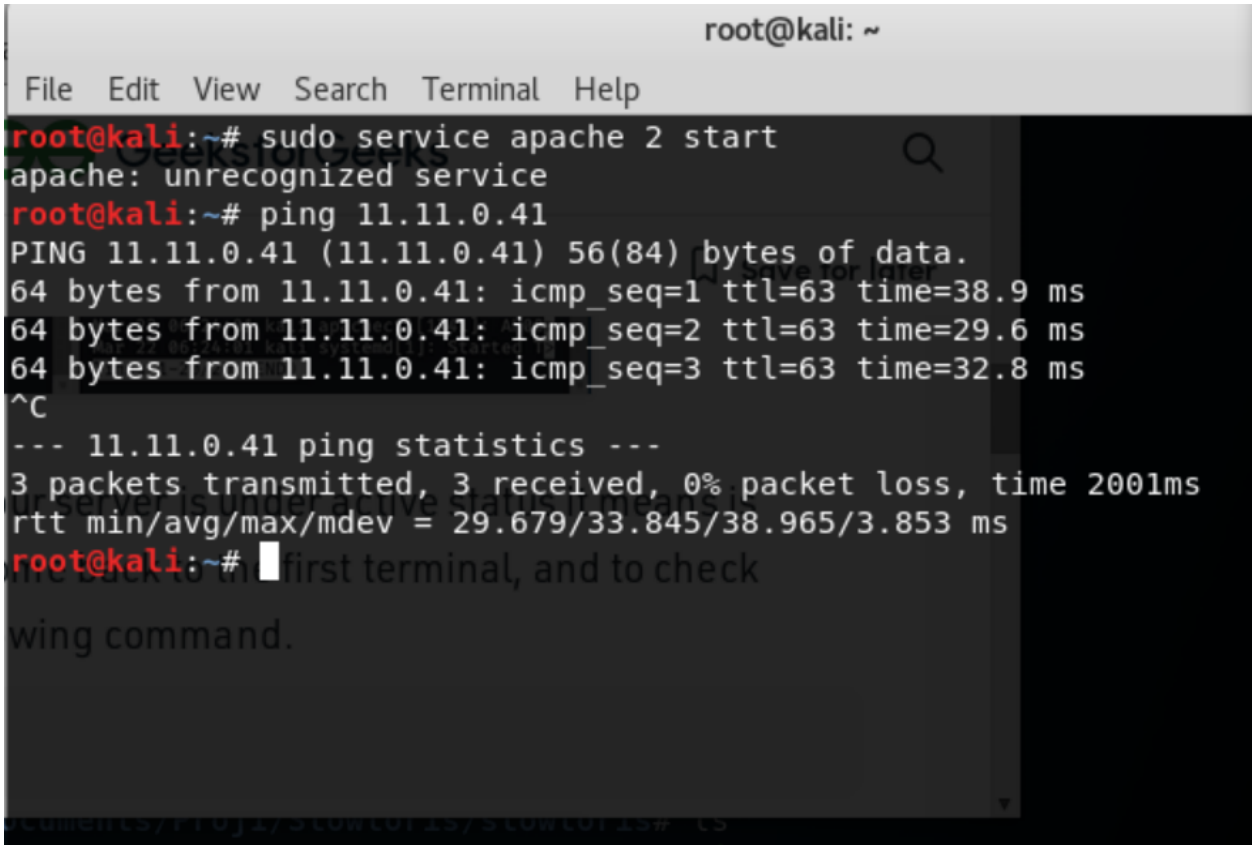


(Figure 2: Slowloris files downloaded from Github)



Following the steps given to me by geeksforgeeks for using Slowloris, I found out that it tells me that I have an apache server up and running to test the DDoS program. I looked into it further and read the README.md file from slowloris along with the Github page for slowloris. I figured out that you don't necessarily need the apache server to test our code since I was trying to DDoS the initech website (11.11.0.41). I pinged the website and found out that it is up and running and our connection to that website results in zero packet loss. In Steps 7,8, and 9, in the Geeksforgeeks article, it told me about the apache server so I decided to skip those steps since our website, initech, is our target website. Figure 3 displays the ping to the website.

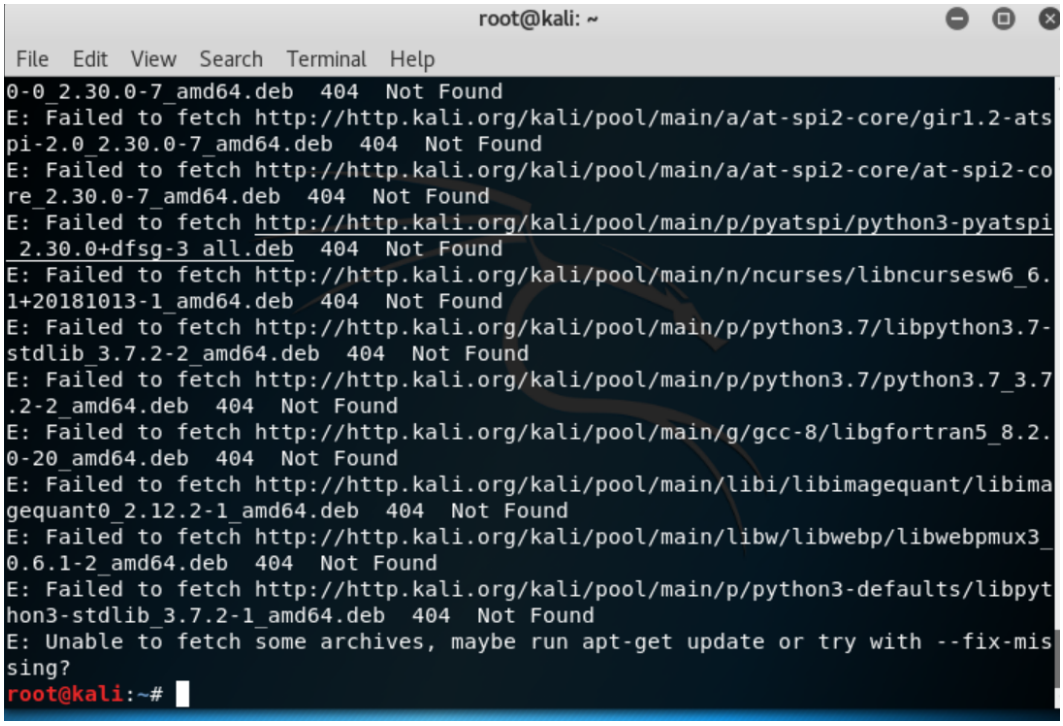
(Fig. 3: Ping from VM to website)



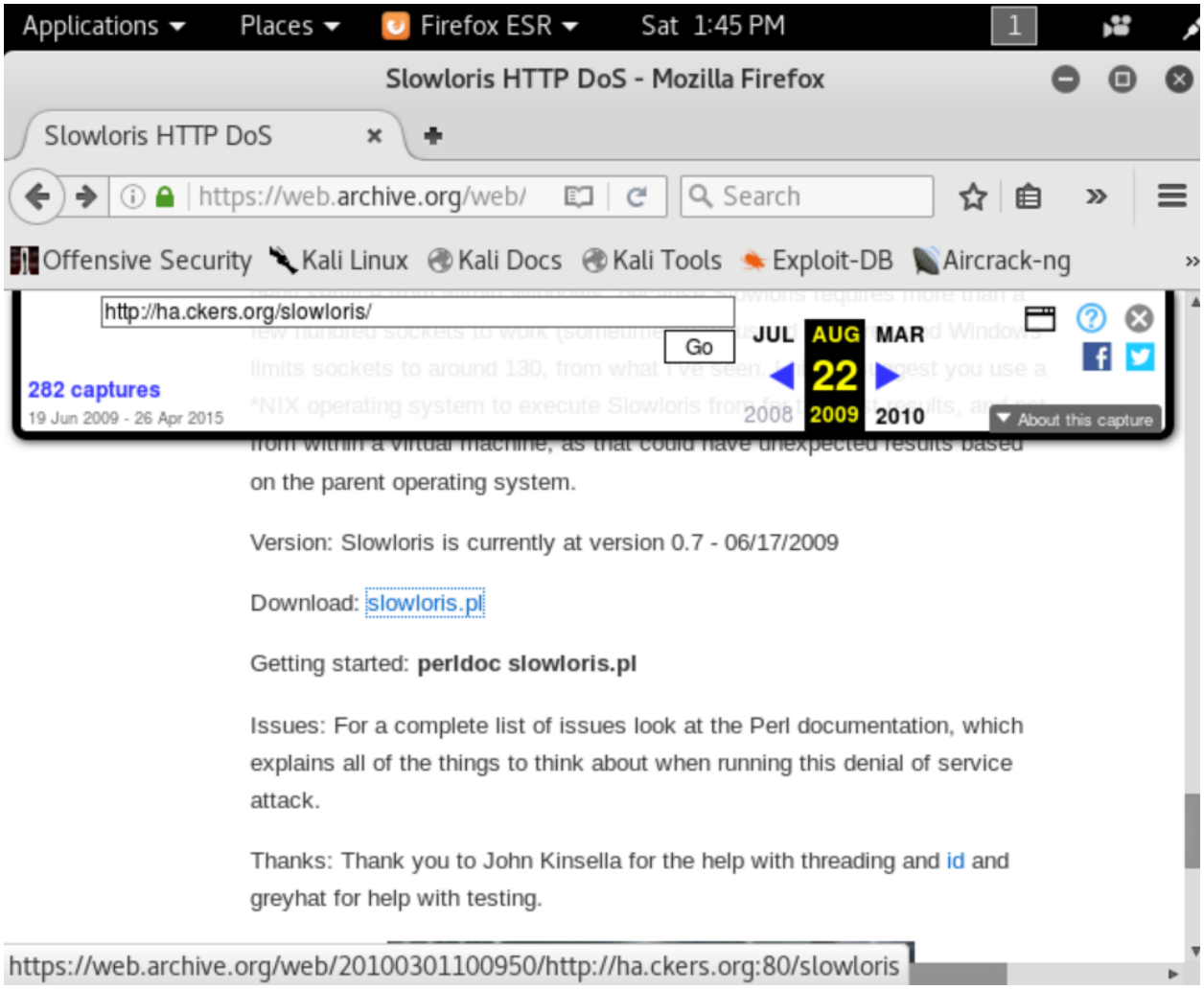
I couldn't get the update to work with python3 so I decided to look for another way to use solaris. It kept saying that I do have python3 downloaded but also when I tried updating it would

give me. Figure 4 provides a screenshot of Kali Linux telling me that they couldn't update or download the files. I then tried to flood the address with syn packets through hping3. I tried to flood the website using the command in figure 5 but it resulted in a 100% packet loss with the website still up. I stumbled upon the wiki page with a web link to a slowloris information page. If you scroll down the wiki page, you find the link titled, "Slowloris HTTP DoS." I clicked that and led us into a web archive of the Slowloris HTTP DoS. From there, there is a download link to the slowloris program as shown in figure 6.

(Fig. 4)



(Fig. 5: Slowloris.pl)



After that step, the link brought up a page of code and I looked at it. The webpage provided us with a lot of information and also an example of how to run the code. In figure 6, it shows an example of the given code, how to run the slowloris command, but not the way to make slowloris actually executable. It was weird because at this point, I had no idea what to do and searched up what a pl file is and how to make a file executable. I found out that I should copy the needed code and paste it onto a file in my VM files under the name of slowloris.pl.

(Fig. 6)

```
=head3 Testing Example:

./slowloris.pl -dns www.example.com -port 80 -test

This won't give you a perfect number, but it should give you a pretty good guess as to where t

=head2 HTTP DoS

Once you find a timeout window, you can tune Slowloris to use certain timeout windows.  For in

=head3 HTTP DoS Example:

./slowloris.pl -dns www.example.com -port 80 -timeout 2000 -num 500 -tcpto 5
```

Since it is just a regular file, it cannot be executed. I went on several websites including askubuntu, medium, and fosslinux, to search up a way to make that particular file work. I compared the differences between the websites' commands for making the executable file and decided to go with "chmod +x **filename**" because it was shown in all websites.The filename in this case is slowloris.pl. It is shown in figure 8. I then execute the file shown in the example in figure 9. In figure 9, -port 22 means that I am performing that slowloris.pl file on port 22 with a number of 200 packets.

(Figure 7: Left is found on medium.com, Right is found on Fosslinux.com)

```
> chmod +x hello
> ./hello
hi
```

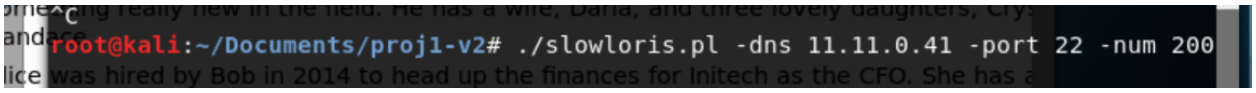
To make the file executable, we will use the "chmod" command as demonstrated below.

```
$ chmod +x greetings
```

(Fig. 8)

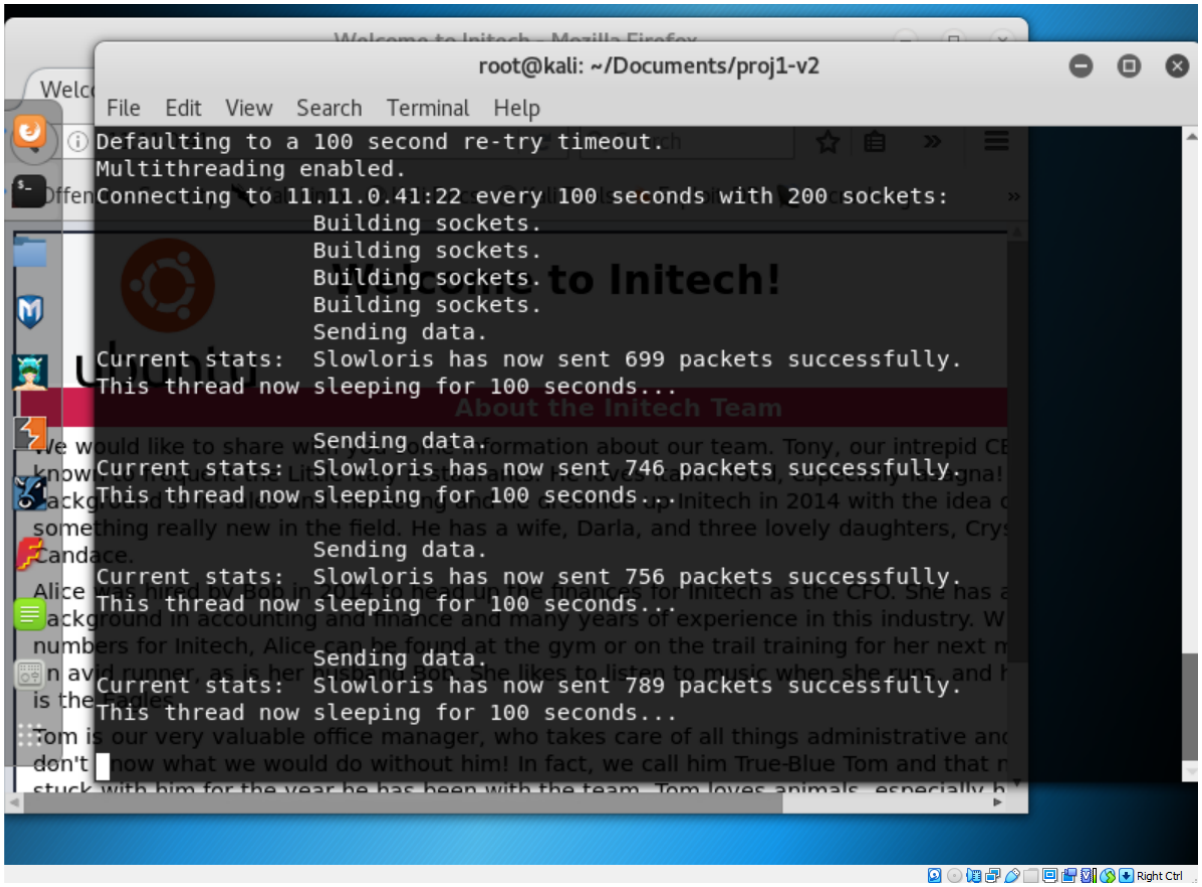
```
root@kali:~/Documents/proj1-v2# ls
slowloris.pl
root@kali:~/Documents/proj1-v2# chmod +x slowloris.pl
root@kali:~/Documents/proj1-v2# ls
slowloris.pl
root@kali:~/Documents/proj1-v2#
```

(Fig.9)

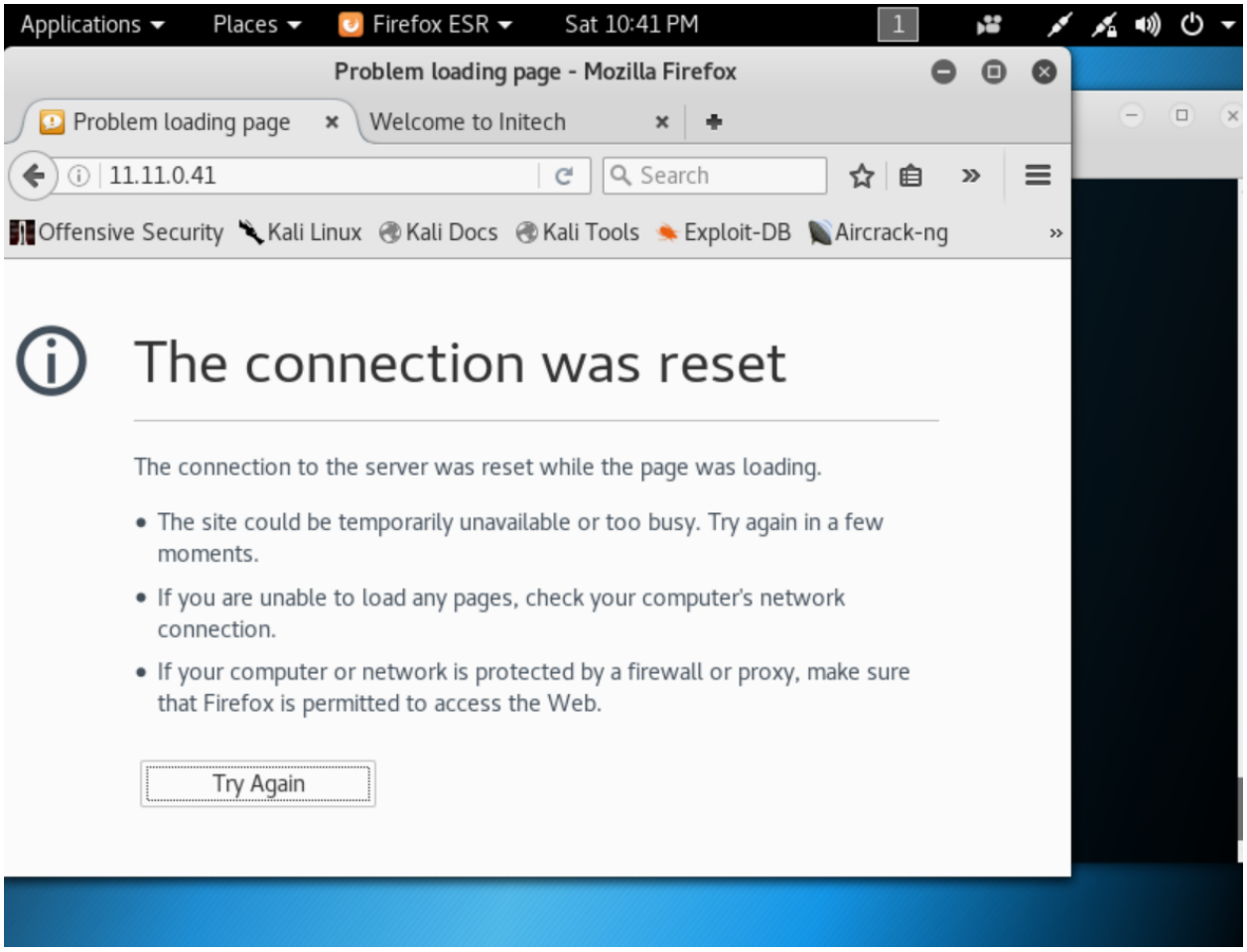


After typing in the program, I tried executing the same program through different ports. At first I tried executing it through port 20, it did say that it was successful in sending the packets as shown in figure 10, but the website did load. It had loaded even when the program was running and when I opened up another tab. Next I executed the same program but on a different port, port 20. It was then successful in attacking the website as shown in figure 10 and figure 11.

(Fig.10: Slowloris Response)



(Fig. 11: Slowloris Attack on Port 22)



Overall, the project was a mainly difficult task. At first, I looked up types of DDoS/DoS attacks online and found a couple of them. I then decided to pick slowloris because it was the first one I saw at the moment. I went ahead and tried it using the Github repository then applied it to my vm. At first, it seemed too easy to be true. After I got it installed, I decided to execute it and then here comes the difficult part of the project. It did not work because of the python version installed in the vm. The slowloris code required python3 but it was already pre-installed with Kali Linux. It still didn't work. I went through a variety of websites, forums, and command lines to try and get it to work. I contacted the developer on the Github repository that the syntax error is still consistent, and I went ahead and tried different types of attacks. As documented

above, I caught a break for our project through a link through a wiki page of slowloris. The experience overall was pretty good with a lot of hardships. In the end, it was satisfying to do it successfully and to see it finally run on the website.