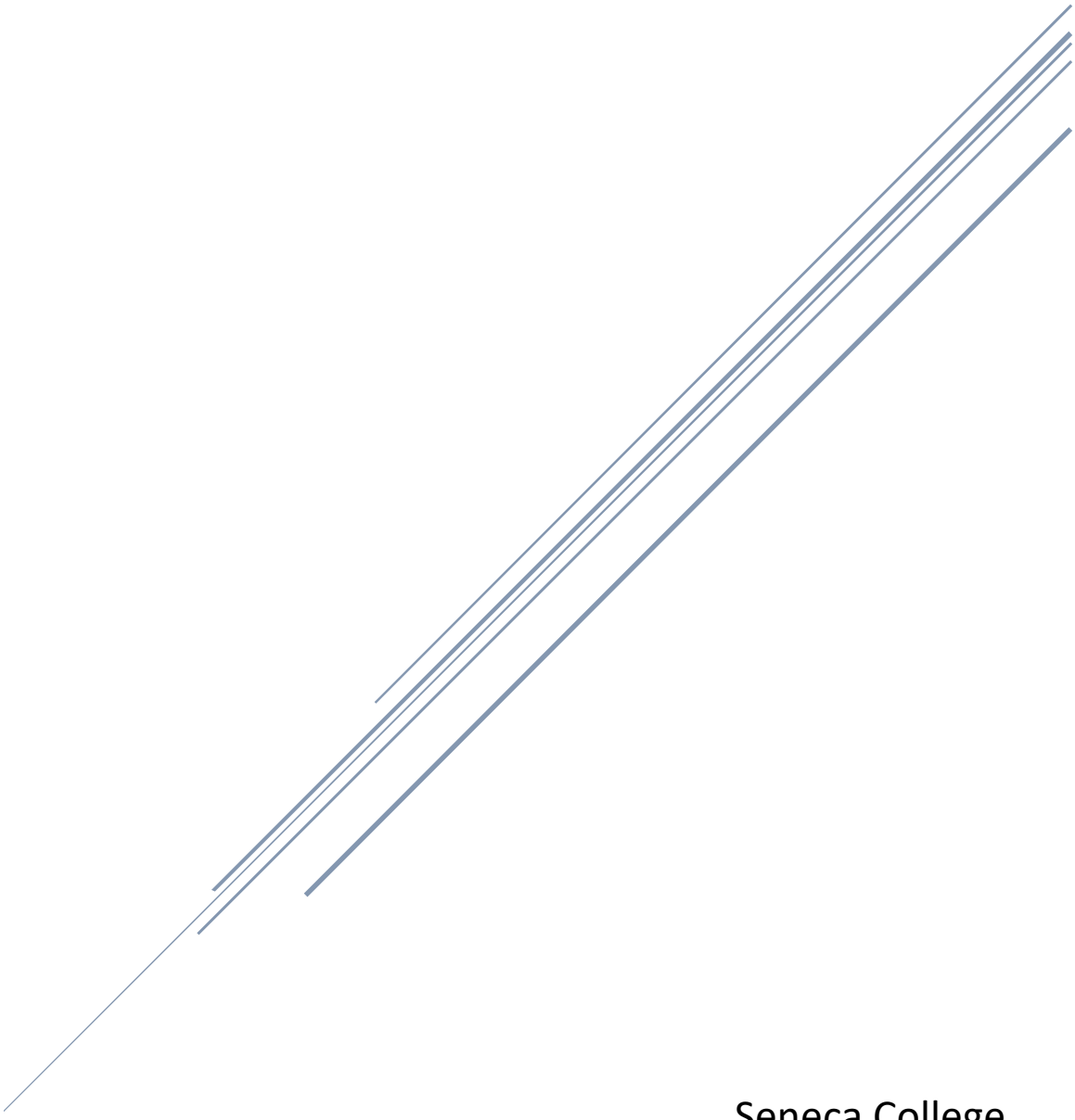
SRT411 PHASE 3

Generating Visualizations



Seneca College
SRT 411

Group Members Associated with Configuring Project Phase 3

Professor Asma Paracha

Members

Ali Abdulkarim: 150706166

Murad Okasa: 108741208

Buwaneka Hettiarachchi: 104376165

College: Seneca College

Table of Contents

Introduction

This is the final phase of the three phase SRT 411 Project, the main functionality of this project was to create an environment and perform log collections and then generate visualizations out of the collected logs. Phase 1 focused on the creatin of the Environment, while phase 2 focused on the log collection and Log generation. This Phase is focussing on the Visual aspect of the generated logs from the phase 2 of the project. The main tasks that our group was assigned in this portion was to compile everything, do data aggregation and processing, Perform indexing and generating queries, create visual representation of the data collected in phase 2, identify the topmost attacks and show how the data is detecting them, and Identify the Normal and Abnormal activity in the system. Additionally, we also performed a Nessus scan of the Attack network to identify the most vulnerable ports and give a broad description about them.

Nessus Scan of the Network

Nessus is used as proprietary Vulnerability Scanner; The main functionality of Nessus is to show the most vulnerable ports that are present in the scan network (172.20.21.1, 172.20.21.2, and 172.20.21.3). Nessus is similar to the OpenVAS Greenbone Vulnerability Scanner; we performed the Nessus scanner in this segment is because we wanted to compare from the scan that we performed in the last phase.

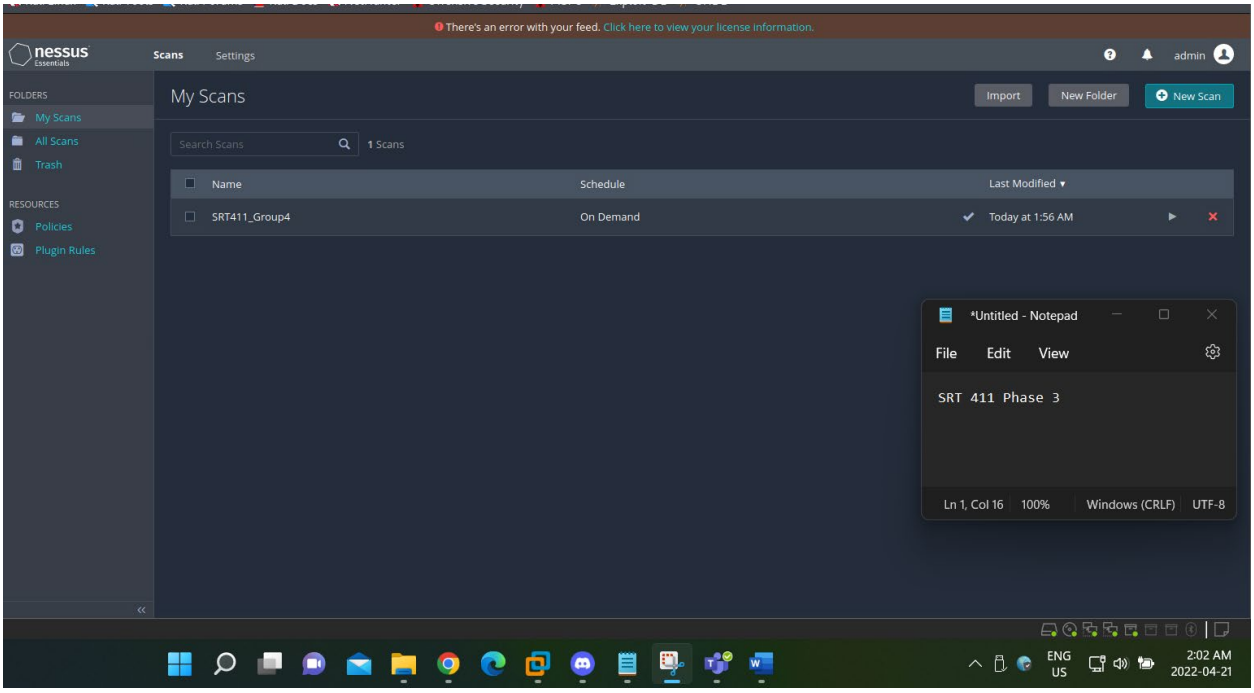


Figure 1: Nessus Running without any issues

For This project, we will be running Nessus in the offline version, this means it wouldn't have all the necessary tools like Nessus Professional. For the Scan we named it (SRT411_Group4 Scan). And the Image Below will show the results of the scan.

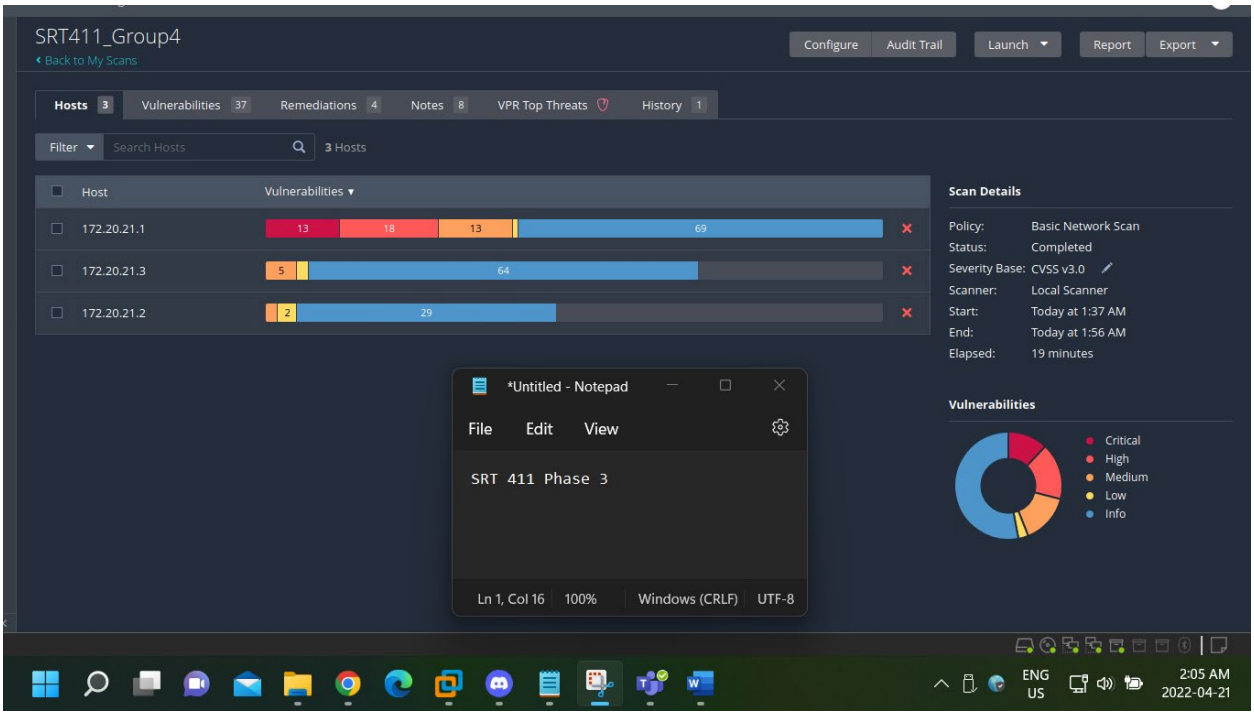


Figure 2: Results of the Nessus Vulnerability scanner

As shown in the above figure, we are witnessing the Vulnerability scan report on the Nessus scanner. The Report states that there are three hosts being scanned and a total of 37 Vulnerabilities were discovered within the scan. Machine (172.20.21.1) seems to have the greatest number of vulnerabilities because the scan shows that there are 13 Critical open ports while there are 18 ports on high alert, 13 on medium alert and the 70 scanned ports are either Information about them or ports with low threats. In the Other hand, machine (172.20.21.2 and 172.20.21.3) has only 6 ports with medium threat levels, indicating that they are less likely to get attacked in comparison to the other machine.

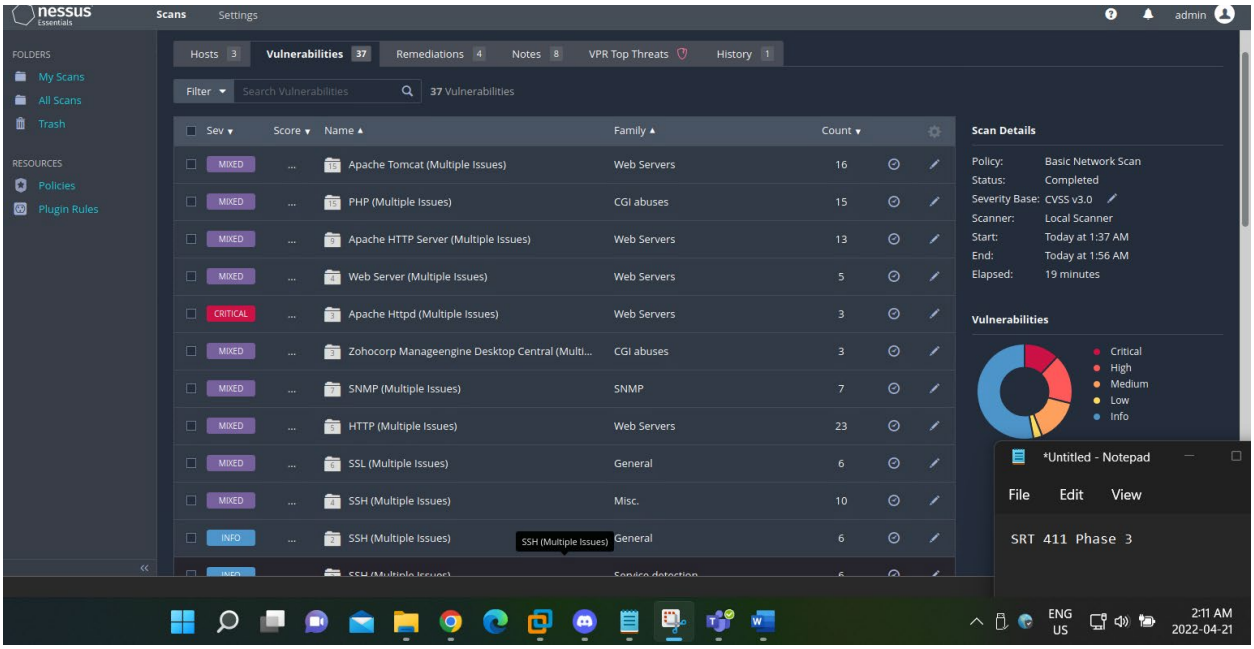


Figure 3: Information on the Vulnerabilities

Most of the Information about the vulnerabilities show their names and the Family of Vulnerabilities that they undertake. Some of the Notable Vulnerabilities include the Apache Tomcat, PHP, Apache HTTP Server issues.

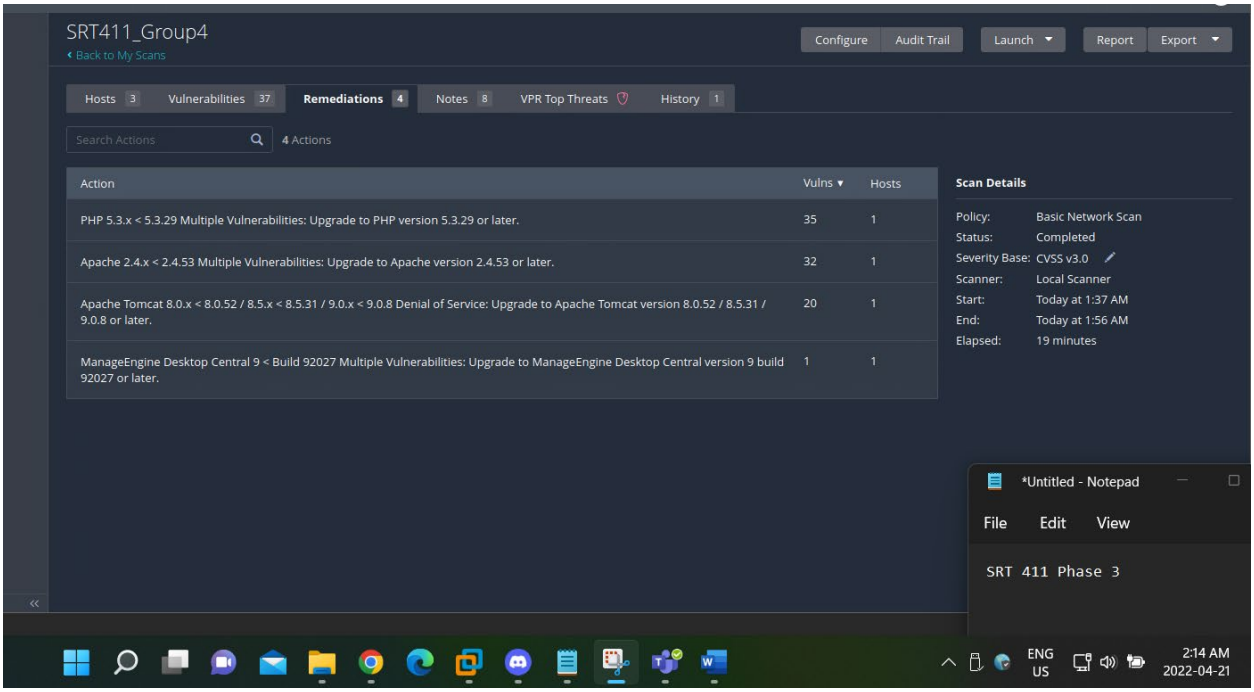


Figure 4: Showing indication of what remedies to be used

The above figure shows what are the remedies or patches that are required for some of the known highly critical vulnerabilities. In most cases, these services are vulnerable because they

are using Service that are outdated and Nessus gives them recommendation to Upgrade the services to never version.

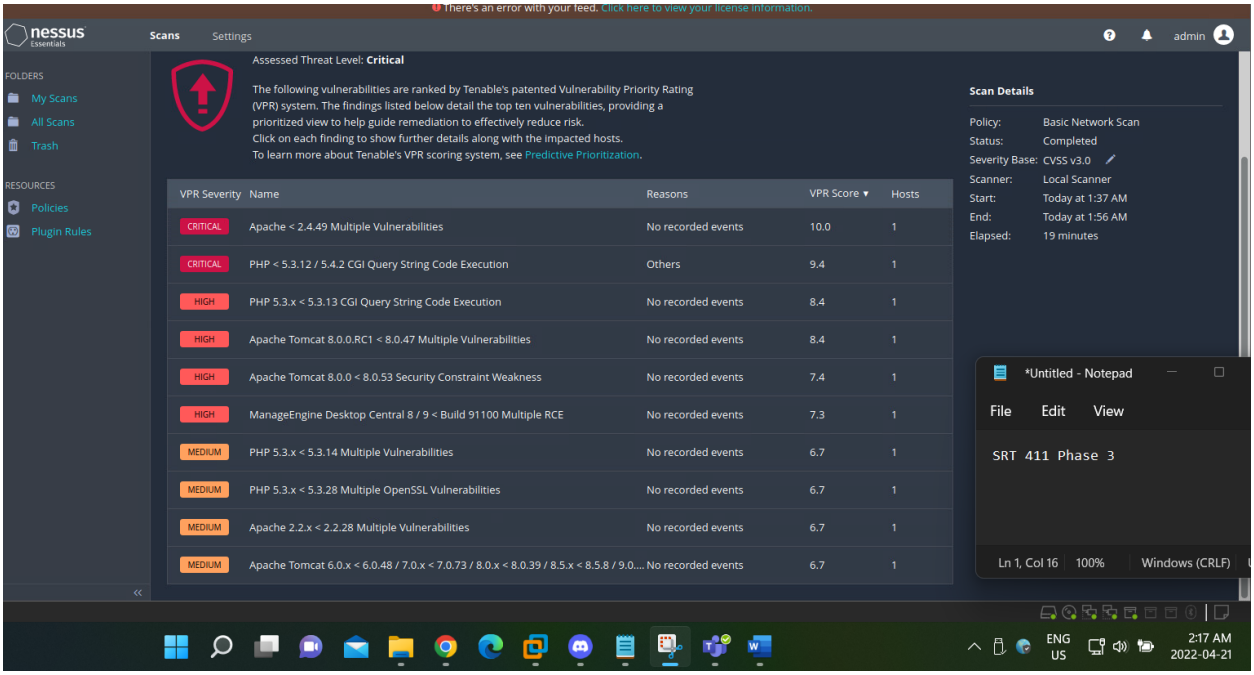


Figure 5: VPR threat level for the Vulnerabilities

So, this Segment shows the severity of each Vulnerability and as you can see, Apache Multiple Vulnerabilities and PHP CGI Query String Code Execution are the known culprits for the critical Vulnerabilities. This mainly concludes the Vulnerability Report that was generated from the Nessus Scan, this scan was mainly used as an offline scanner, this means that the tool offers limited services.

Data Ingestion (Compile, data Aggregation and Processing)

In this segment we will be Ingesting the generated data through the use of Logstash and then we will send the Logstash Configuration into the Elasticsearch, Kibana, Logstash Platform. The Collected Data log is in the form of a CSV, this CSV file contains all the information of the collected normal traffic and the suspicious traffic. We will Ingest this log into one single Index in the ELK stack. By doing this we are able to find the difference between the Normal and Anomaly traffic. For the Ingestion of the data, we have several different methods to ingest them, some of these methods include Filebeats, Winlogbeats, and Logstash. For this Purpose, we will be using a Logstash script to add in the needed segments.

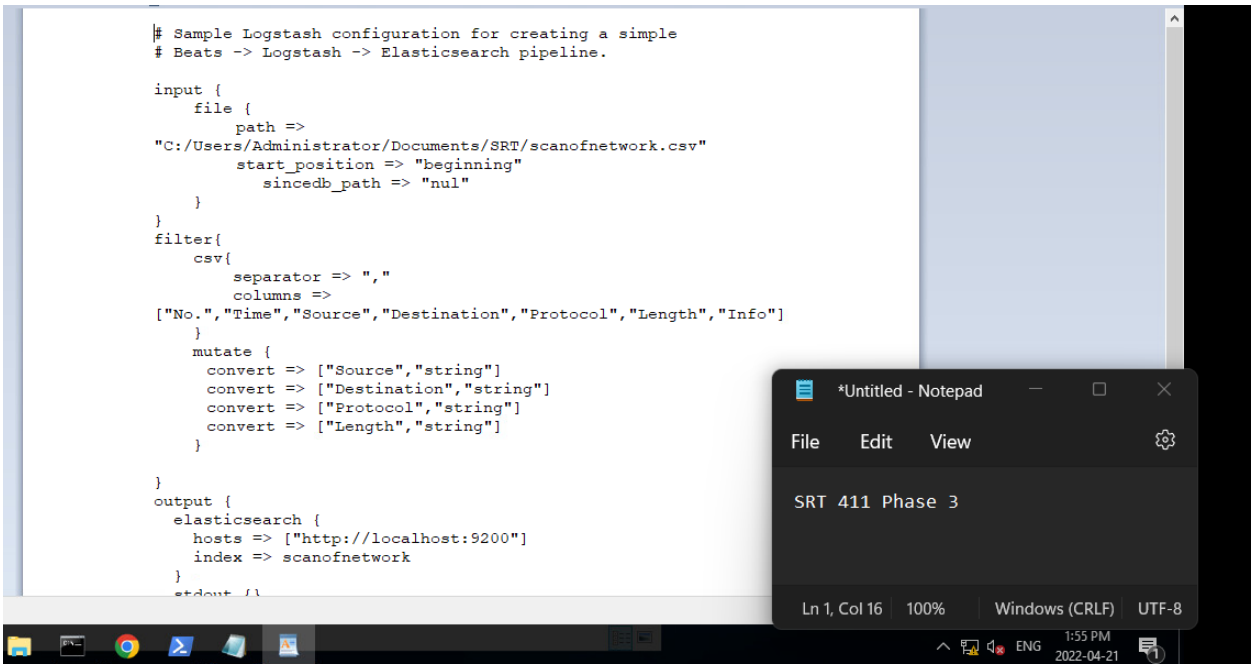


Figure 6: Example of the Logstash script that we used for the ingestion

A Successful Logstash ingestion should create an index within the Elasticsearch Environment. If you do not see any indexes, this means there needs to be some changes to be configured in the Logstash Script.

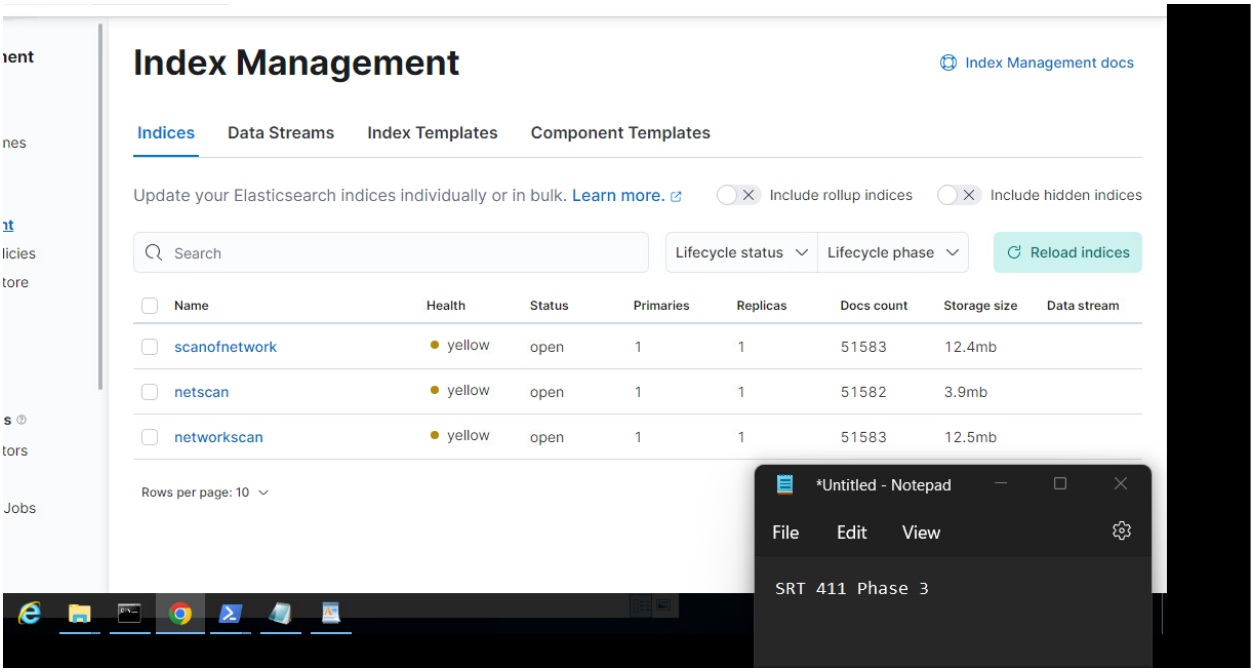


Figure 7: Indexes being created in ELK index management

The Above figure is an example of when a successful Ingestion has occurred, as you can see our group created three Indexes, but all the three of them are the same dataset, the reason why we did three different indexes is for testing purposes of our ELK environment.

Generating Queries and Indexing

We mainly generate queries to get an understanding of the data log file, mainly we need to look into variables such as filtering any anomalies in the traffic and with referring that query we will be able generate some interesting Visualizations. First we will show an example of a test query that we generated, in this query, we will generate all the ICMP packets that was captured within the network Capture, We do this solely Through searching the protocol at the search icon at the Elasticsearch Discovery section. Initially the index has around 51000 hits, but when we only search the queries with ICMP traffic, we were able to filter out the hits to only 145 hits.

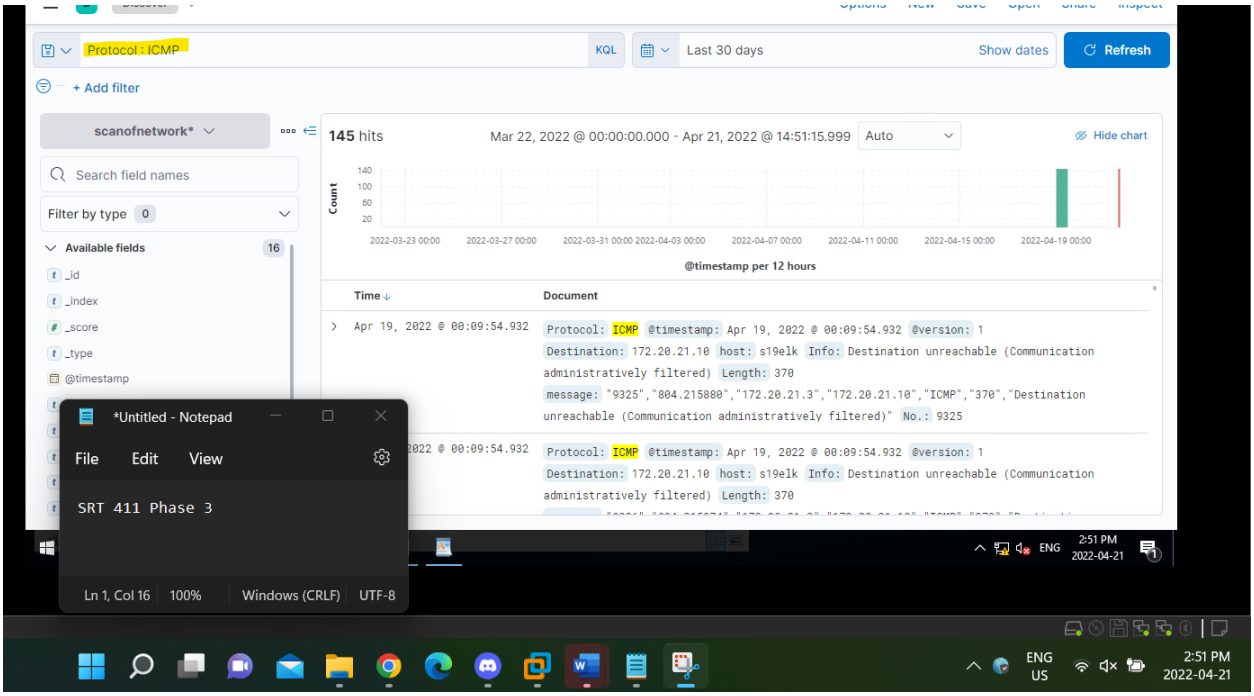


Figure 8: Example on how to generate Queries

As the above figure shows you how we would perform queries and indexing through the discovery section. In we won't to work on figuring Indexing and generating Queries, we need to use the Dev Tools option from Elasticsearch. For this segment, we would do Index Mappings and other important Queries that would give us some Valuable information about the Index That we generated.

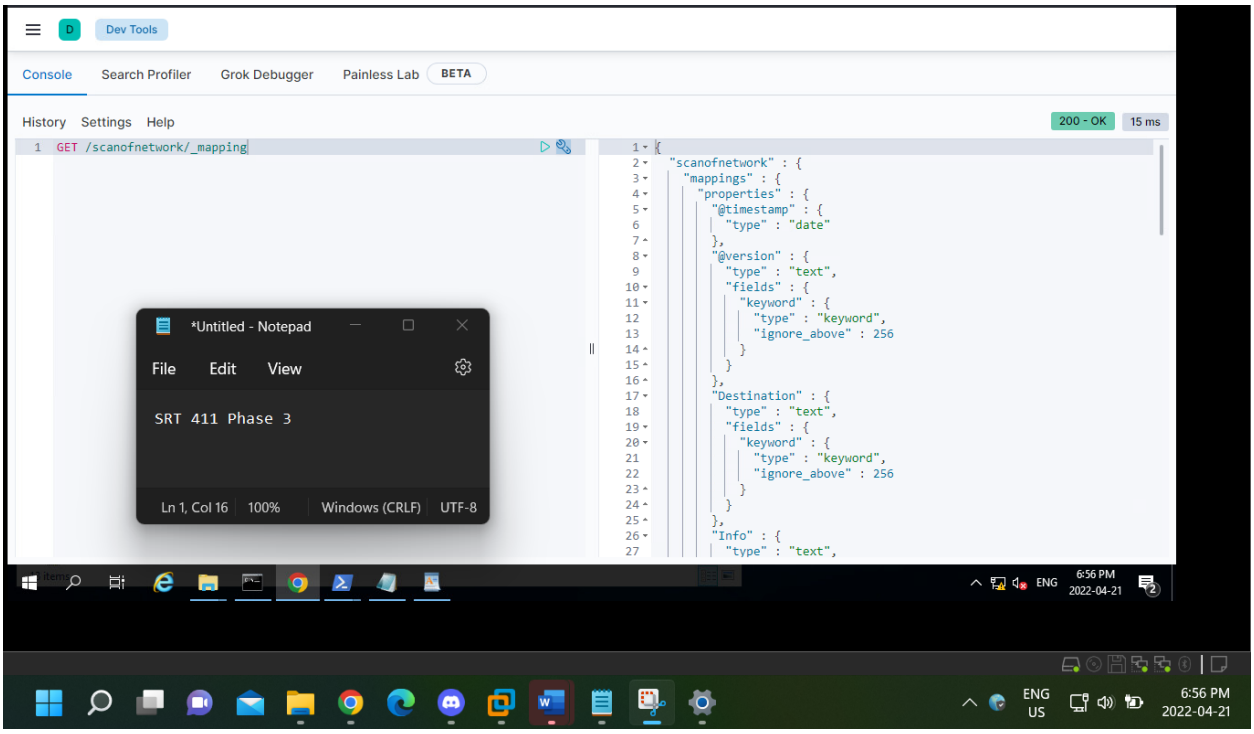


Figure 9: Information showing the Mapping of the Generated Index

As you can see the image above represents the index mapping of the generated index, this mainly maps all the queries into their representative position at the Index itself.

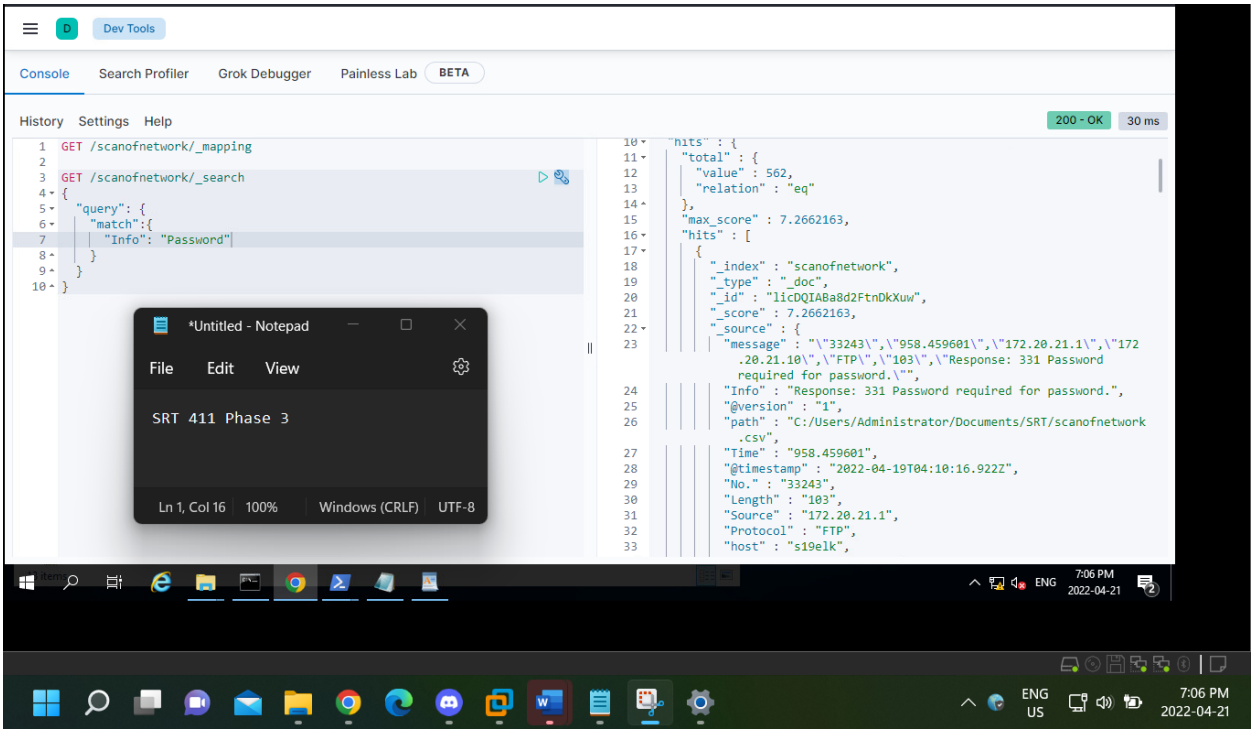


Figure 10: Query trying to get information on any instance with the term Password

The above Query shows Information on the term Password, I put this because in the Info segment in the Log file, it represents the number of tries it took the Password to get the right answer. From the Query we can see that we are able to get information such as the Source IP address, Destination IP address, and the most importantly, what protocol performed this action. As you can see the FTP protocol has performed this action. In total, this Query has 562 hits.

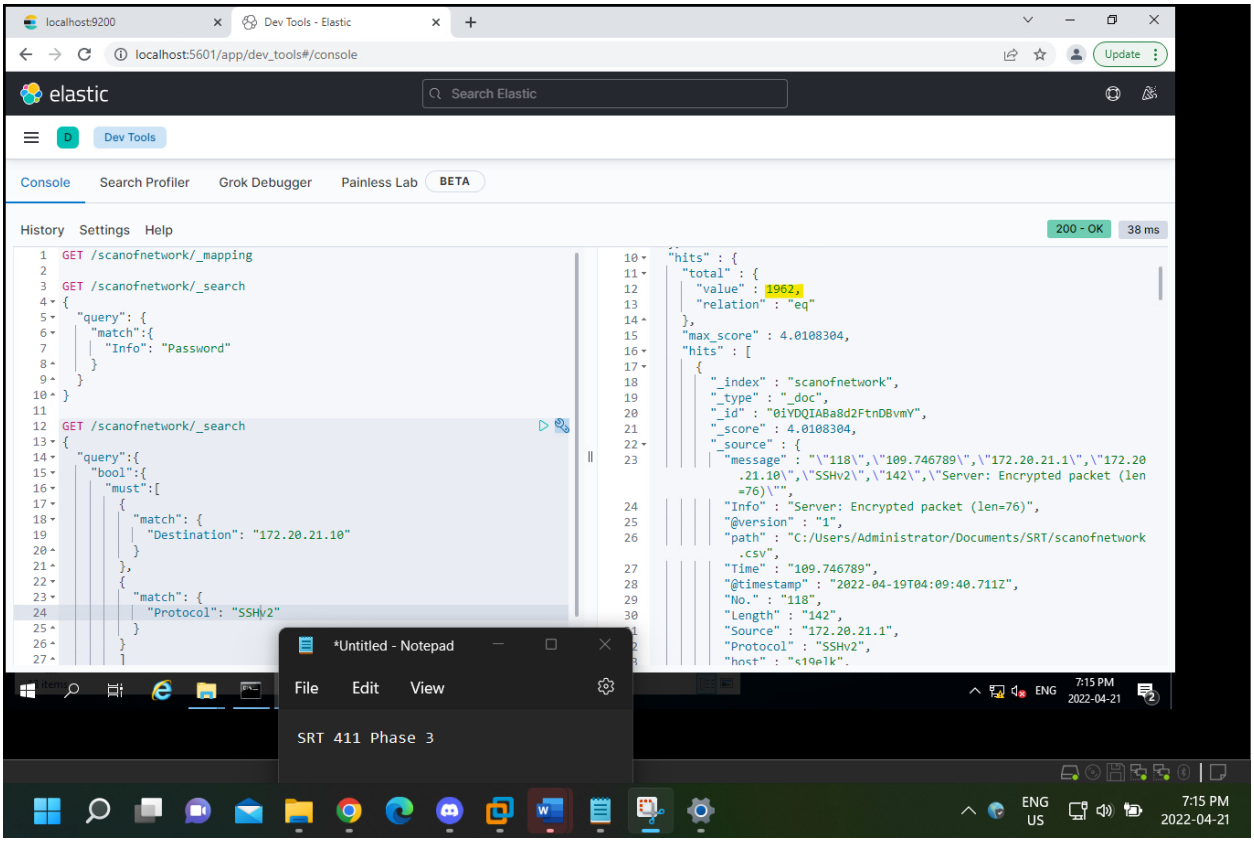


Figure 11: Query Showing Information of the Protocol and its relativity to a Single IP

In this segment we will be looking at the SSHv2 protocol and as you can see, we were able to witness that there have been a total of 1962 hits within the Environment. Under Information we can see that it is stating itself as Server Encrypted Packets. If we further scroll through the System we are able to see information on key exchanges.

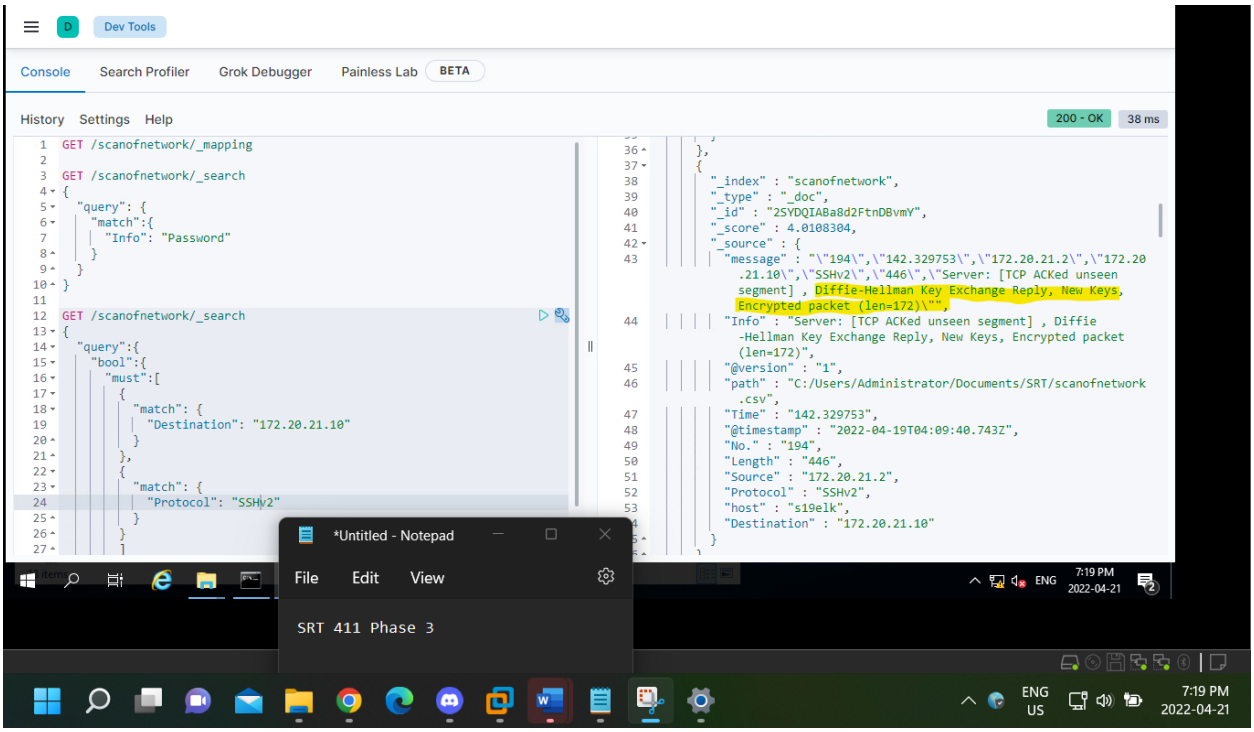


Figure 12: Instance of a Diffie Helman key exchange

Mostly this is an indication of a possible key exchange, and this actually gave the Attacker machine access to possibly perform Brute Force attack on the System, as you can see SSHv2 is mostly used to encrypt the authentication layer, transport layer, and connection layer. This is an indication of a violation provided by the Kali Linux machine.

Generating Visualizations

In this segment we will work on creating multiple visualizations from the collected dataset from the phase 2 of the project. Most of the visualization have various different information, some ranging from just normal Visualizations such as the Number of Source or destination IP addresses and to more detailed visualizations such as detecting Normal and Anomaly of the System. First let us start by creating some normal Visualizations such as Bar graphs representing the Instances like the most used protocols, Source and Destination Addresses.

Top Source Ports within the dataset

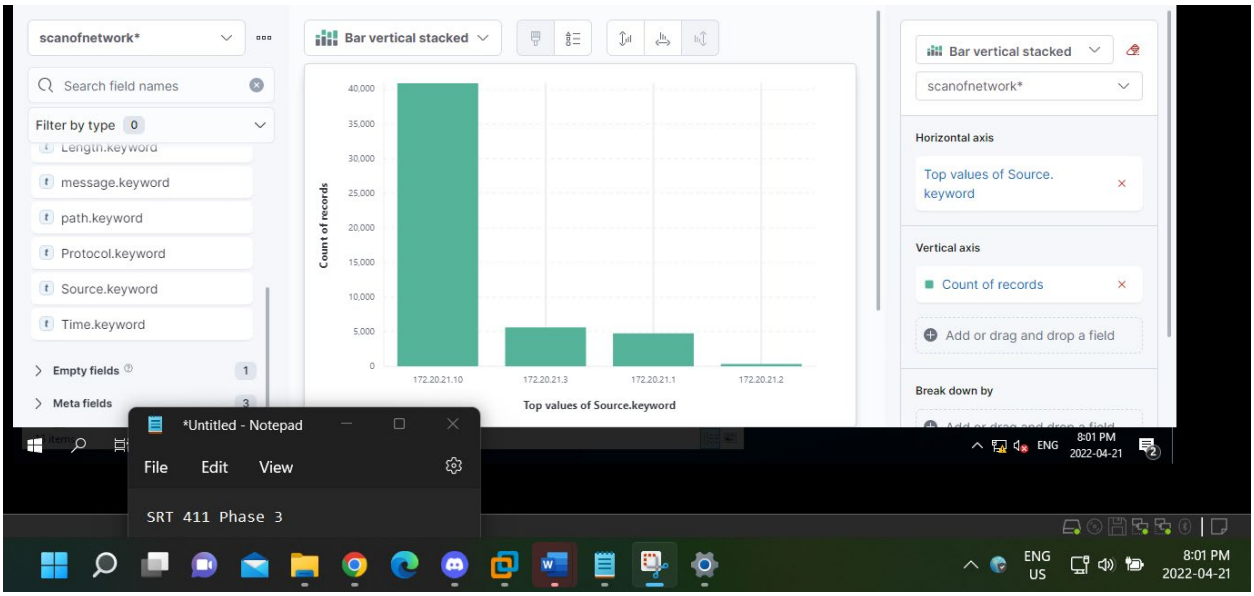


Figure 13: The Top value of the Source Addresses used in the dataset

As you can see in the above diagram, the highest used Ip address as the Source address is that the Kali Linux IP address, this is obvious because we used the Kali Linux machine as the machine that performed all the action such as performing normal scans of the network such as Nmap scans, performing various different attack on the network and generating Normal Traffic within the Network. The we have the DVWA and WordPress machine as the second highest source address, and then followed by the Metasploitable machine and the LAMP server machine with the least collected records. Our team chose the bar graph to represent this Visualization, mainly because it represents the flow of the IP addresses in a clear on concise way.

Top destination ports

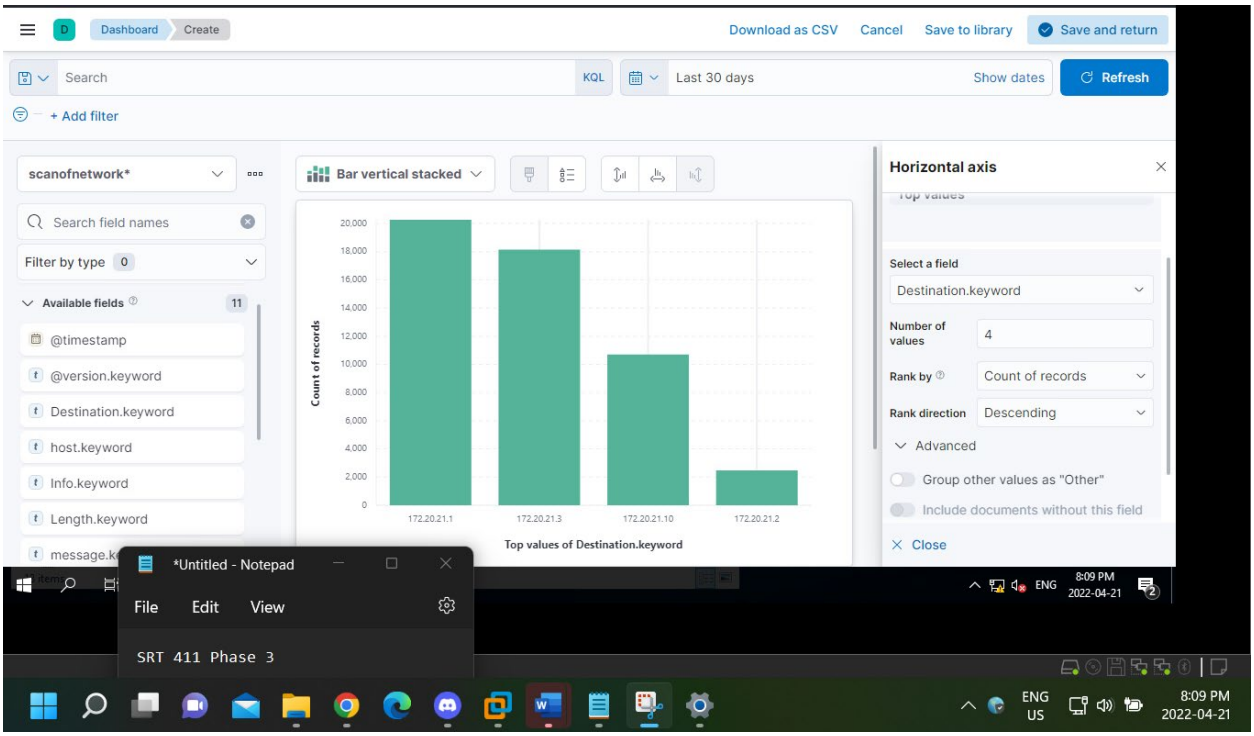


Figure 14: Visualization representing the top destination ports

The above visualization represents the top Destination IP addresses that was recorded in the dataset that was collected in the second phase of the Project. And as you can see the highest recorded destination IP address belongs to the Metasploitable machine. This is mainly because we were performing many attacks in this machine due to the fact that this machine is a vulnerable machine with several different vulnerabilities in the machine. The second machine was the DVWA and WordPress machine, this is mainly because I went through normal web servers to generate normal traffic using both WordPress and DVWA. Th third machine is the Kali Linux machine, this machine is most likely in the destination field because it is trying to communicate with these machines and perform various attacks and get information related to those attacks. Lastly, we are having the LAMP server as the least used Destination port. Again, the best suitable Visualization for this data set is a Bar graph that accurately represents what we are looking for.

Average Length of Protocols over the data collection period

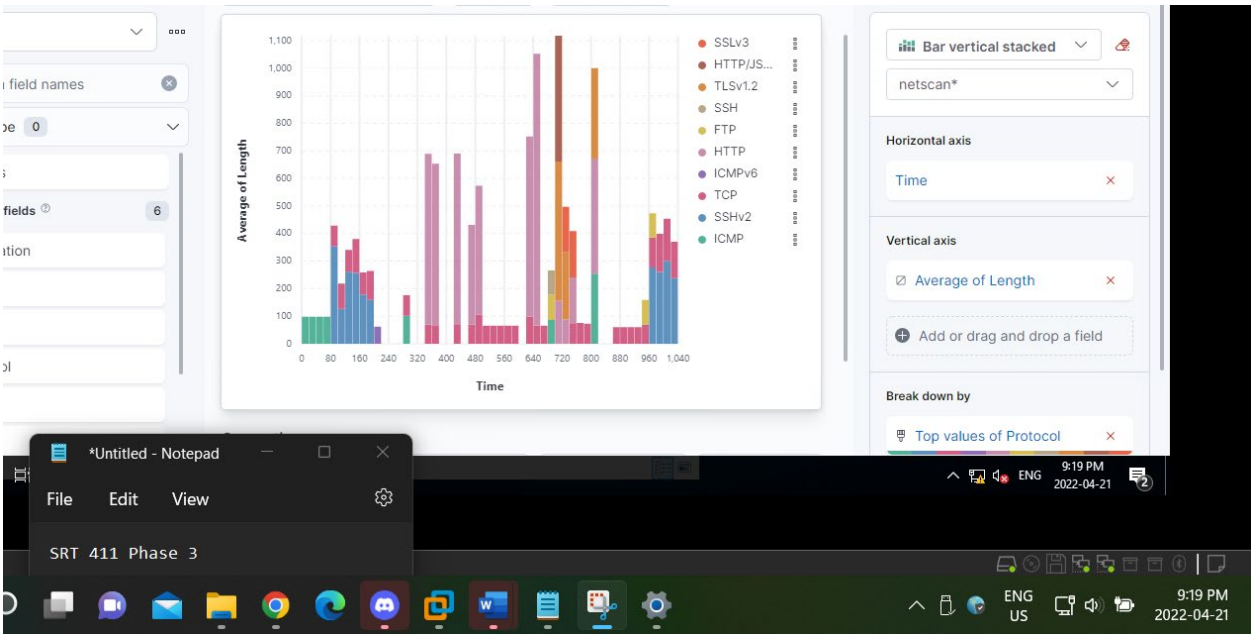


Figure 15: Visualization showing the average length of each protocol during the certain data collection time span

So, the above visualization seems to look really complicating, our group tried other visualization platforms to have a better understanding of the generated visualization but ended up using a bar graph to represent the given visualization. As show in the diagram there are many instances of protocols being present within the dataset, but lets focus on the most important Protocols that we are focusing, If you look in to the TCP protocol, you can see it seems to maintain an average packet size during every period of data capture, between the instances of Time, in between 40 and 240, we can see a continuous motion of TCP packets with the same type byte length, my prediction is that, this was the moment when were performing a DDoS scan in the Metasploitable machine. HTTP packets seems to have a larger size on average, this is true because we are trying to get HTTP requests and each packet will have variety of different sizes. I did start the scan from performing Ping scans on all the machines, and as shown in the Diagram, we are able to ICMP packets being scanned in the first few Time intervals. The last bits of the Graph contains information regarding FTP and SSHv2, from my speculations, this is supposed to be when our team performed the various FTP and SSH Brute force attacks. In conclusion, even though this graph shows valuable information, it is bit unclear on explaining the graph, but it will be further explained with all the details in the phase 3 video record submission.

Pie Chart Describing the most used Protocols in the data set

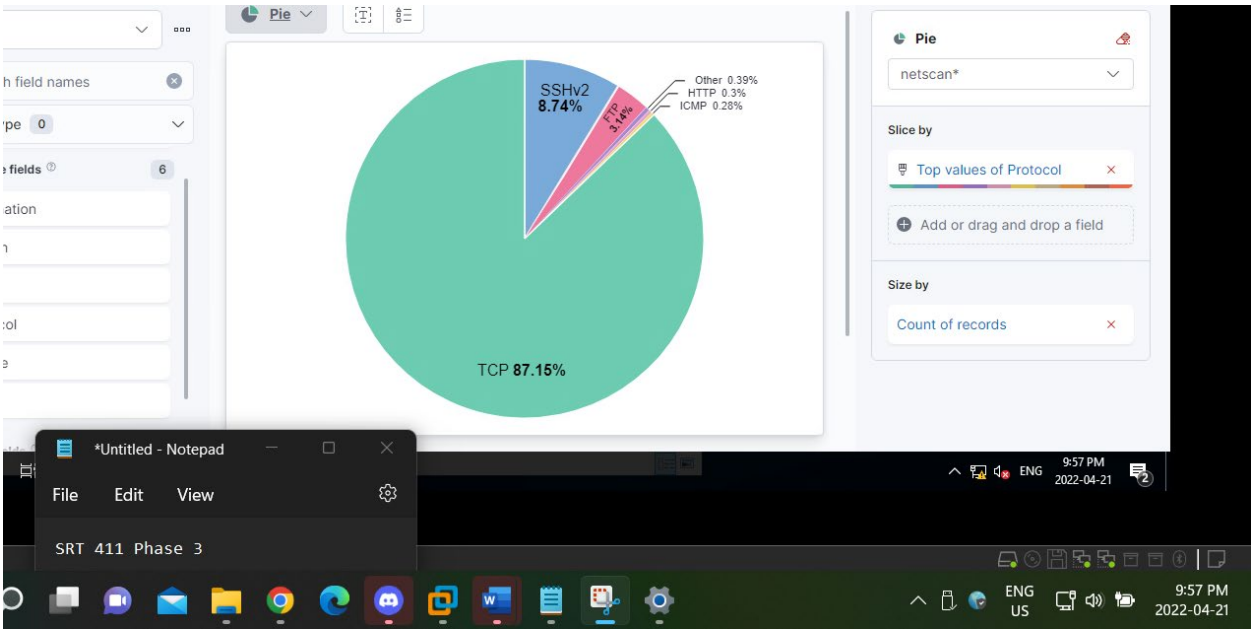


Figure 16: Pie chart showing the results

As you can see the above Pie chart, you can see the distribution of the packets within the data collection. And As shown from the visualization, we can come to a conclusion that the most of the packets in the Visualization are TCP packets, and they are followed by SSHv2, FTP, ICMP, and other Protocols that are scanned with the data set.

Average Length Anomaly Traffic in regards to PCAP time

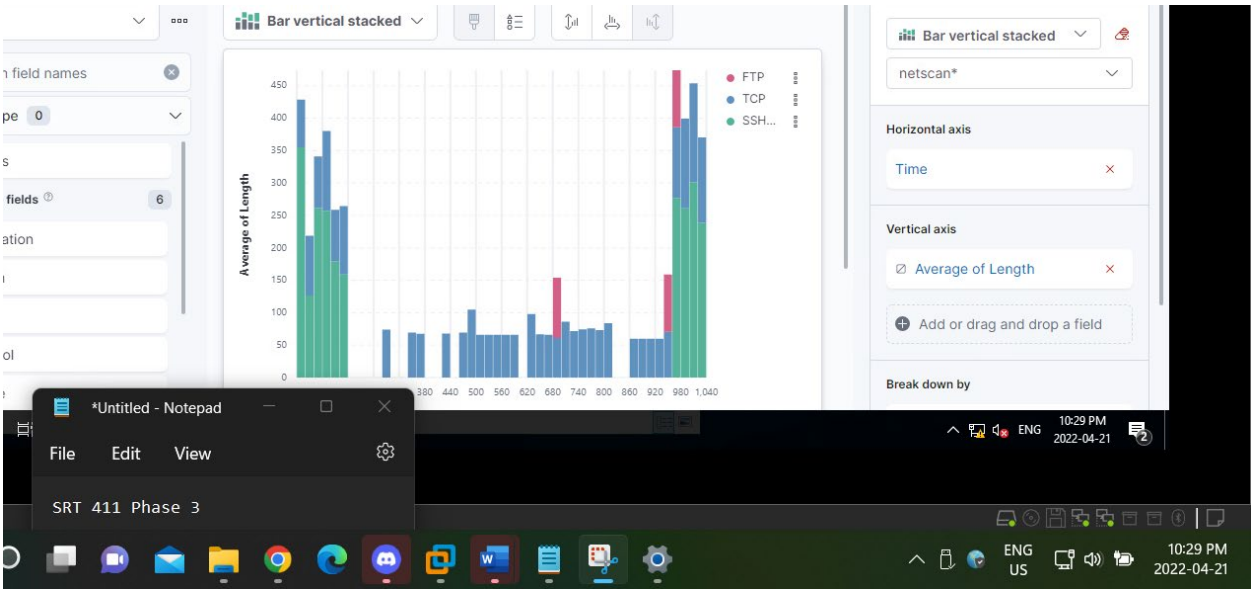


Figure 17: Figure of the Anomaly Packets, which is show with significant spike

The above Figure shows the Anomaly traffic of the Pcap Environment occurred, as you can see most of the Anomaly occurs during the time period when the average length is at a significant spike. As shown in the middle of the packet is actually just normal TCP traffic, But when ever TCP is involved with SSH or FTP, this means that there is a certain attack taking place within the environment, Like in the beginning, this is a clear indication of a certain DDoS attack, while TCP is communicating with both FTP and SSHv2, we have a clue that this is regards to the certain Brute force attacks done in both FTP and SSH versions. Again, the Data, best describe this function is A bar chart because it is capable of showing more accurate results of any Anomalies present within the Network.

Average Length of the normal traffic protocols within the Pcap time frame

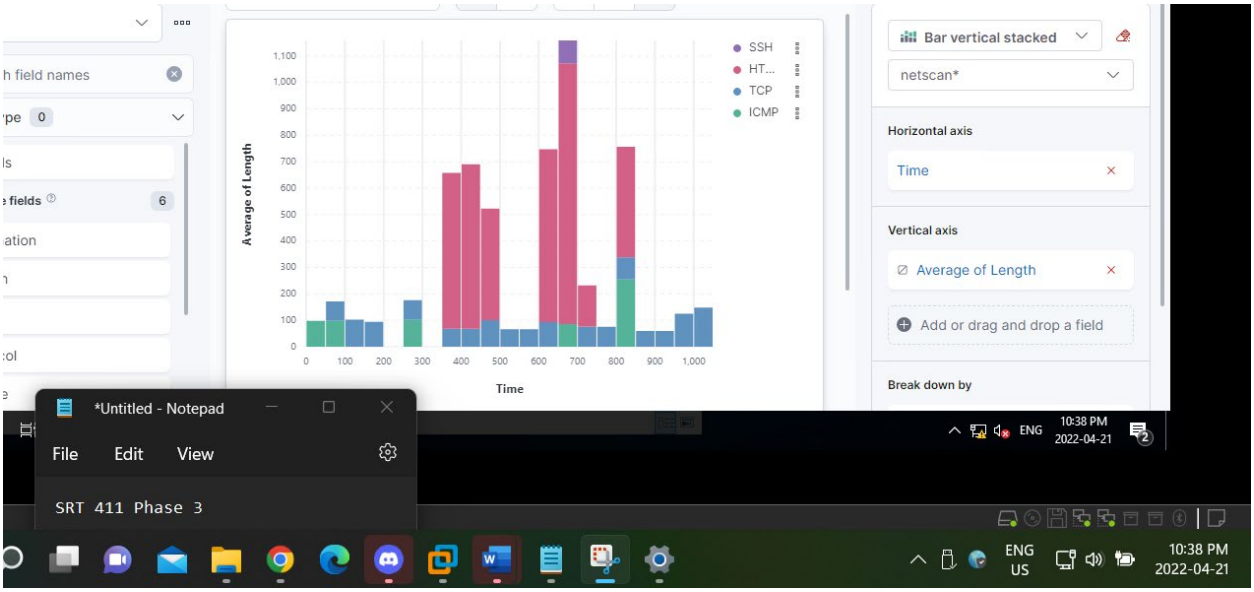


Figure 18: Graph showing the Normal Traffic

The above graph shows the normal traffic of the data capture and as you can see, Most of the Normal Traffic comes from TCP, HTTP, ICMP, and normal SSH traffic. These were manly captured from visiting Web pages such as WordPress, DVWA, and accessing Apache web servers hosted by the machines. SSH traffic is also generated mainly through the use of performing SSH connections from 1 host to the other. ICMP is mainly available, again, these are through the use of performing ping requests to other machines and generating Nmap results. This is done by a Percentage Bar graph to show the distribution of each protocol length through the certain time period.

Defense Mechanisms to the created Network

Defense is an important aspect for a network, in this scenario our main defense strategies include performing various patches for the known vulnerabilities that were discovered from our protocol scanner. Secondly, we need to regularly update each machine so that they have proper defense mechanisms and safety precautions. For certain attacks such as Denial of Service attacks, we

need to include a Denial-of-service Response plan on how to deal with issues like that. For certain Brute Force attacks, you are supposed to implement strong password creation and regularly change or update passwords to high standard of Security. If a machine has several vulnerabilities in its system, then the best thing to do is to upgrade the machine into a never version so that updates will be automatically taken into account. These are some of the Defense Mechanisms that needs to be taken care of.

Conclusion

In conclusion, in this phase our main purpose was to generate Indexes, create Visualization to detect normal and abnormal traffic, Identify any queries with valuable information, and Finally perform a Network scan using Nessus. In most part of this phase, we were able to locate the abnormal traffic in the system as there are many instances to find these abnormalities, but our team does have a high feeling that we could have made it better by going deeply with the anomaly detection from the machines. But Overall, our team was able to get information regarding this matter and was able to clarify the difference between the normal traffic and the Abnormal traffic within the Environment. In the future studies, we can make it better, by performing more attacks and by further populating the Log files that has to be analyzed.

References

"Elasticsearch SQL: QUERY ELASTICSEARCH INDICES WITH SQL," Elastic. [Online]. Available: https://www.elastic.co/what-is/elasticsearch-sql. [Accessed: 21-Apr-2022].

"How to detect ddos attacks on my network," Wireshark Q&A. [Online]. Available: https://osqa-ask.wireshark.org/questions/60763/how-to-detect-ddos-attacks-on-my-network/. [Accessed: 21-Apr-2022].

Administrator, How to perform TCP SYN flood dos attack & detect it with Wireshark - Kali Linux HPING3. [Online]. Available: https://www.firewall.cx/general-topics-reviews/network-protocol-analyzers/1224-performing-tcp-syn-flood-attack-and-detecting-it-with-wireshark.html. [Accessed: 21-Apr-2022].

"SSHV2 server," SSHv2 Server. [Online]. Available: https://www.synopsys.com/software-integrity/security-testing/fuzz-testing/defensics/protocols/ssh2-server.html. [Accessed: 21-Apr-2022].