# Chapter 4: Results and Discussion

## 4.1 Discussion of results in the testing Phase

In a cloud environment, just like the standard and physical environment, risks may occur in every form. As mentioned in the Literature Review section, companies may find the migration experience to the cloud, as a difficult for the simple reason that they lack trust in it. In today’s modern world, migration from a physical environment to a cloud environment may be a decision hard to take. Therefore, the aim of this research is valid for the results to demonstrate if the cloud environment is safe or not. In the Methodology stage of this research, IaaS was utilized to implement the environment. After this, attacks were launched and tools were used. The results were clear, and some of the attacks were successful, while others were not. Along this research, some difficulties were encountered; one of them being that only two instances could be created, and therefore, tests could not be tried as planned. In cloud computing, there can be many risks, and in this case, the risks did not shy away. However, as much there are risks, there are also multiple essential components and features which could assist the user in mitigating risk from happening while utilizing OpenStack.

## 4.1.1 Queries and Challenges

While conducting this research, during the methodology stage, there were multiple challenges that were encountered - some of which were easy to turn around and solve within seconds, while others on the other hand presented a much stronger challenge. Therefore, with bigger challenges, more research was required to identify what the problem was and how to solve it, in order to be able to continue working on the research.

## 4.1.2 What went wrong

One of the main problems which were encountered, as mentioned before, was that not more than two instances could be used at the same time. Another problem was the specifications of the virtual machines. Although a guide was adhered to and the specifications were set accordingly, the Virtual Machines did not turn on properly, and the configurations were adjusted several times until everything functioned properly. An issue was the specifications experienced on the researcher had on his laptop. Although the laptop boasts a 32GB RAM, some of the Virtual machines had to be tested with minimum specifications. This therefore, limited the possibility of testing with higher specifications, which were critically required to test whether OpenStack functions much more efficiently on high specifications or low specifications. OpenStack is expected to function better when having high amount of specifications. Another query was that the licenses were not purchased, and this affected the research because this is the reason why only two instances could be utilized and not more.

## 4.1.3 The Solutions

The solution for the first problem involved making use of a Windows 10 virtual machine. This solution was done to establish a connection between the Linux VM, which OpenStack operated on, and the Windows 10 VM instead of using an instance from OpenStack. The second problem was solved by setting the specifications according to the guide. The solution for the specifications of the physical specifications could not be solved as that was the only device that the researcher had in his possession when conducting this research. The ultimate solution for the licenses query was solved by using another Virtual Machine to establish a connection.

## 4.2 Other problems during the testing phase

When conducting this research, setbacks were expected, and from such difficult encounters, great results can still be achieved. During the testing phase, failures are expected, and in the presentation of work during the methodology, great results are still expected as knowledge and learning can be gained. During the testing phase, it is essential to gain as much insight and knowledge as possible to complete the presentation of work and to present the best result with the utmost effort. During the testing phase of this research, multiple problems were encountered. One which stood out of the many was that despite following the guide and instructions by doing everything step by step, after restarting the Virtual Machine that OpenStack was going to be launching and operating on, it did not turn out to satisfy the researcher’s expectations. There were two problems for this situation to happen.

First of all, the laptop that was being utilized for this exact research does not have very high specifications. Therefore, the laptop did not help. Another problem was that the specifications that were set on VMware Workstation Pro, needed to be customarily set for the virtual machine to work correctly. However, this was not the case. Therefore, to continue working on this research, the virtual machine was not shut down and everything functioned smoothly.

Another problem experienced was that during the testing phase, the second instance was created successfully, but when trying to access via an SSH connection, the process of creating another instance was in vain, due to the fact that only two instances were allowed to be created and utilized. Therefore, another virtual machine was created and set on the same network, in order to be able to connect with the OpenStack Virtual Machine and to minimize the problem of not having another instance on OpenStack. Another problem with the second instance was, that when the researcher tried to ping from the OpenStack server to the newly created instance, the ping was a failure, however when the researcher tried to ping the first instance, the ping was successful. After days of trying to find a solution through research, it was discovered that when utilizing OpenStack without a license, only two instances could be used.

Another problem was that since the researcher had never used OpenStack before, the utilization of OpenStack was rather challenging for a novice. This is because videos, pictures, and many guidelines on how to install OpenStack while being able to understand what is going it is being utilized and active, can be rather challenging. While making use of OpenStack, the knowledge gained was rewarding as it eased the understanding of OpenStack’s use. Through practice, its methodology and implementation became easier than one expected. Since Wireshark was one of the main tools that were needed during this research, a small problem was encountered during this research - at first Wireshark did not gather the packets when pinging between the OpenStack server and the Windows 10 virtual machine. There were two problems. The first problem was that initially the Virtual machines were put on the same network, while the other problem was that the correct interface was not chosen. Therefore, the correct one was needed to gather the expected packets.

## 4.3 Queries experienced during the presentation of the work

During the methodology stage, the presentation of work is the most crucial and stressful stage in the whole research. The reason is that when presenting one’s work with so much effort and time invested, the result is assumed to be a perfect one, and one which is desired and expected. However, unfortunately, there can be cases during the presentation of one’s work, where something goes wrong in a manner that the problems and queries that were met during the testing phase of the methodology, appear during the presentation of the work phase. This situation is stressful, because while the user thinks that the problem has been solved in the testing phase, the user believes that the problem should not reoccur in the presentation of work as well. However, that is what makes significant research. In the case of this research, there were a small number of queries during the presentation of the work phase. Some of the difficulties were met in the testing phase, while other queries were only encountered in the presentation of the work phase. During the presentation phase of this research, one of the queries appeared when an empty folder was sent from The OpenStack Virtual Machine to the Windows 10 Virtual Machine. The packet was not sent as expected. When trying to read the packets gathered from Wireshark, there was no trace of any communication. However, afterwards, the same folder was sent but this time from the OpenStack Virtual Machine to the host. This means that the laptop utilized worked and the packets were gathered successfully from Wireshark. Another problem was that the folder was trying to be sent, however the command did not work properly. The problem was, that since the command was never utilised and tried, in order to understand the purpose of the command, the research that was made during the presentation of work was referred to. According to the information research, the command needed the IP address of the virtual machine or the host and not the network ID only. The mistake was that instead of entering the exact IP address, the network ID was inserted in the command.

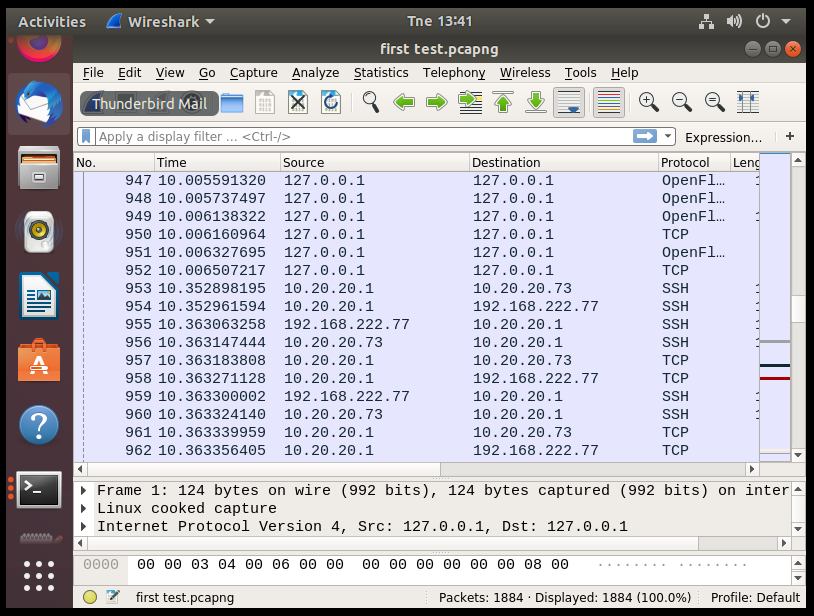
Another problem was that the laptop that was utilized during the implementation was not strong enough for three virtual machines to work simultaneously. The OpenStack virtual machine requires 16GB of RAM, however the Windows 10 Virtual Machine was set on 10GB of RAM and the kali Linux Virtual Machine was set on 4GB RAM. Since my laptop was functioning gradually, the Windows 10 RAM amount was adjusted and decreased to a 4GB RAM. Another challenge which affected the methodology of this research was that the hard disk space of the laptop. The other difficulty was the internet speed, as, it took a lot of time for the folder to reach the destination.

## 4.4 Discussion of the Results

The results gained during the methodology stage was not as expected by the researcher. This is because the number of instances provided was not enough, and therefore the methodology stage was implemented in a way which views whether the security features that OpenStack provides to the cloud environment user, are secure or not. The way this research was implemented and tested, and the way how the results were gathered is more than enough to see whether Keystone or Nova function in a way to protect the user. The results were gathered by the Wireshark and by the packet Wireshark gathered.

## 4.4.1 First Attack

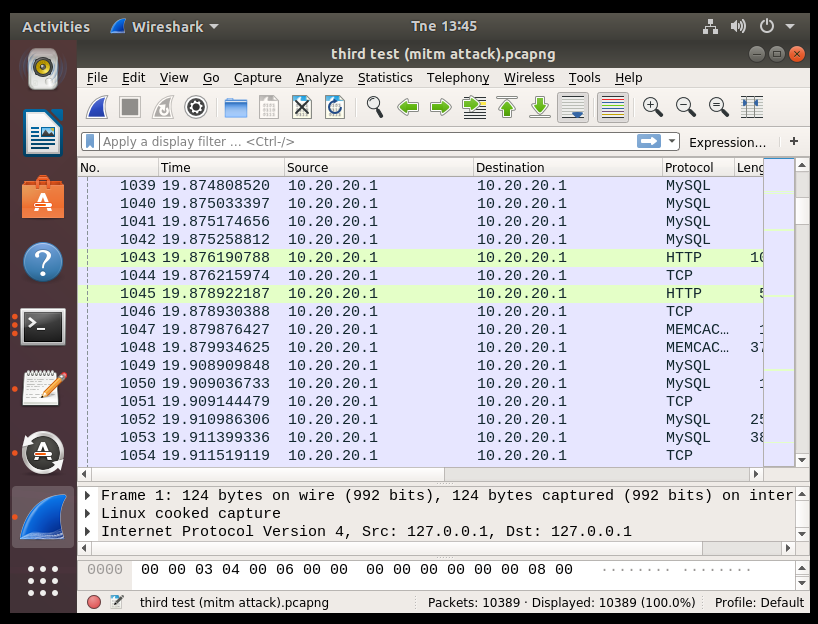
The first attack was made by sending an empty folder to the host, which was the laptop that was in use during the whole methodology stage. This basic MitM attack was done to explore whether Keystone is protecting the instance. The result was positive, which means that Wireshark gathered packets that were actively in transit between the conversation of the local computer and the instance created in OpenStack. While Wireshark was actively running and gathering packets, Keystone did not protect the instance from a MitM during this kind of attack, because Wireshark gathered all the packets that were sent to the host, as illustrated in Figure 8. Keystone was not able to protect either the source instance or the destination because the IP addresses were different from each other, but most importantly they were not on the same network address. Therefore, both Keystone and Nova did not function in a way that protected either the source or the destination device. Another thing is that the server of OpenStack, which has the IP address 10.20.20.1, was not protected either because it was visible in the first scan of this Methodology. The protocol was visible as well; therefore, from this scan, the correct information was gathered.



### Figure 8: Result of the First attack.

## 4.4.2 Second Attack

The second attack was executed in the same manner as the first attack, but the results of this attack were different from the previous attack. This attack was made by sending an empty folder from the instance of OpenStack to the Windows 10 Virtual Machine. This was also done to see whether Keystone or Nova were doing their jobs and protecting the instance and the Virtual Machine; only this time both the virtual machine and the instance were on the same network ID. Also, while the folder was sent to the destination, Wireshark was actively sniffing and gathering packets that were in transit between the communication of the instance and the virtual machine. The results were negative. To be exact, this time, both Keystone and Nova did their job successfully, because both the instance and virtual machine were protected, and Wireshark did not sniff any packet that went from the instance - which is the source to the Virtual Machine, which is the destination, as illustrated in Figure 9. Another important point to note is that these virtual devices were both assigned on the network ID, as mentioned before. Therefore, Keystone had an easy job to hide the conversation from the sniffer, which is Wireshark. Although Wireshark sniffed the OpenStack server, Wireshark did not gain information between the conversation of the virtual devices.



### Figure 9: Result of the Second attack.

## 4.4.3 Third Attack

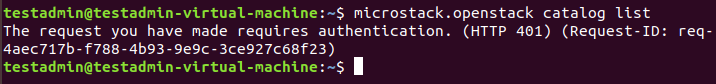
The third attack was made using another Virtual Machine, the Kali Linux. As mentioned before, this Virtual Machine was used as a tool against the other Virtual Machines. Ettercap, was used to launch a MitM attack against the instance and the Windows 10 Virtual Machine. Wireshark, another tool, was involved in viewing whether the MitM that was launched by Ettercap, was able to bypass both security features that OpenStack provides, which are the Keystone and the Nova. Both the IP addresses of the virtual machine and the instance was set on Ettercap. Although this attack was done multiple times and every time that was done, results were clear from Wireshark. The results that Wireshark provided were not successful; therefore, once again, both Keystone and Nova did their jobs once again. The only thing that appeared on the scan as a source was the IP address of OpenStack, which is 10.20.20.1, and the conversation between the instance and Windows 10 Virtual machine that was made by sending an empty file was not visible in the sniffer of Wireshark, as seen in Figure 10. Another point to note is that Wireshark was not able to identify any of the sources that OpenStack has, not even the users.

### 

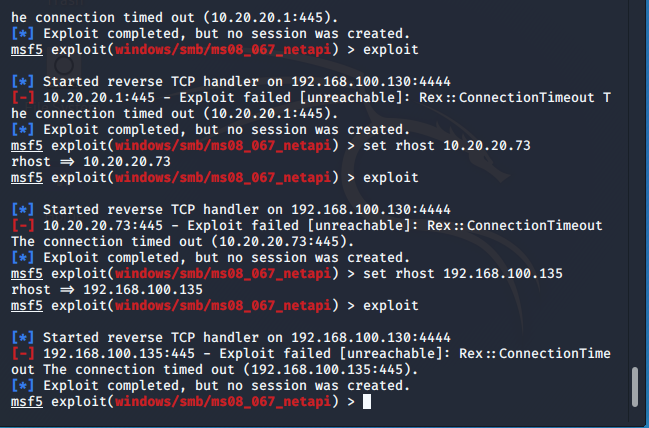
### Figure 10: Result of the Third attack

## 4.4.4 Fourth Attack

The fourth and final attack of this research and the methodology stage were different than the other attacks. Kali was also used, but this time, Metasploit was the primary tool to run a vulnerability scan against the instance that was created in OpenStack, which is the Linux virtual machine. Also, Wireshark was used to try and detect the attack. The first result from this attack was negative because although the vulnerability was launched, the exploit completed the session for the vulnerability scan and it was not created. The result was respectable because, thanks to Keystone and Nova, the vulnerability scan was not successful. After all these attacks during this research, there was no certainty that Keystone or Nova was fully active, and there were doubts that everything which was done, was not done as expected, referring to Figure 12. But before the attacks started, there was proof that the researcher could see the catalogue depicting which software was installed in the OpenStack server. After these attacks, the command “microstack.openstack catalog list” ,was utilized again to view the software installed. Once again a message appeared stating that to view the catalogue, authentication is required. The message is visible in Figure 11.



### Figure 11: Authentication needed



### Figure 12: Result of the Fourth attack.

## 4.5 Discussion on comparing the results against other results

From the result gained during the methodology stage, not all the results came out as expected, as some of the results came out positive, and some of the results came out negative. A critical factor during the this stage was, that the testing and the presentation of work, was not executed according to the plan because of the licenses, which were an essential factor for the testing and implementation stage. The outcome of the first result, after multiple tests and researches, was not successful since Wireshark gained the packets which were in transit between the OpenStack instance and Windows 10 Virtual Machine. The reason was that neither of the used tools - Keystone or Nova, encrypted the communication. When it came to comparing the results, others were able to download the folder, however they could not view the folder because the folder was encrypted and protected by Swift.

The other outcome of the methodology stage was quite successful. Wireshark was not able to gather any information from this type of communication and most importantly, it was not able to gather the folder that was in transit from the source address to destination address. When comparing the result to that of others, admins were able to download and decrypt the folder, while in the methodology stage, the folder could not be opened and was not visible to Wireshark since the folder was protected and encrypted by Keystone. Another essential note while referring to the first and second attacks is, that the results show and give a clear picture to the clients of what the risks are, if the clients do not make proper use of the security features of OpenStack, should the client opt to operate with OpenStack.

The third result of this research was also successful, and the outcome was as expected; the MitM did not work since the packets were not gathered either by Ettercap and from the sniffing, which is the Wireshark. Thanks to Keystone, since the reason for data was not gathered because of the enabled feature. To compare to other results, Marathu et al (2016) utilized Bandit, as this feature assists in discovering vulnerabilities in OpenStack when the analyzations have been combined both statistical and dynamical. Using software developing, the life cycle is the best mixture to make the tool even more secure.

The last and final outcome of this research was as expected because the session of the vulnerability scan was not launched correctly against the instance of OpenStack because of the protection of Nova. Keystone and Nova are some of the features that OpenStack offers to the clients. Saranya et al (2016), state that authentication needs to be checked in an OpenStack environment in order to verify whether the right user is accessing the intended information and instances created in the topology of OpenStack, the cloud environment. A file should be uploaded using the component called Swift in an OpenStack environment. Since multiple companies are thinking of migrating to the cloud, or companies are pondering on whether they should switch to a cloud environment, the features that OpenStack provides are required to protect both the instances and the cloud environment of the company, by assuring them that they work in a safe and riskless environment.

## 4.6 What others gained from their results

Every outcome in a research must have a different result to be able to be compared and to analyze the results according to plan. In the case of this research, the results varied from each other. Some of the outcomes were positive, which means Wireshark managed to get the in-transit information, and the other was negative, which means that Wireshark did not get the critical data from the session. Results that were gained from other papers and studies, were different from each other as well, and every paper has a different opinion on OpenStack and how OpenStack should be used in operation. According to Wuyou (2016) study, there is a reference which states that using Keystone and Swift is a decision which protects your topology from certain risks, but one of the disadvantages was that although Keystone and Swift were utilized, improvements were still in need to provide absolute safety for the cloud user. According to another article by Marathu et al, (2016), Bandit offers much more security in the cloud topology. Saranya et al (2016) went further to state that authentication in a cloud environment should be monitored frequently, so the best service is provided to those who make use of the cloud infrastructure and the security features that OpenStack provides to the user.

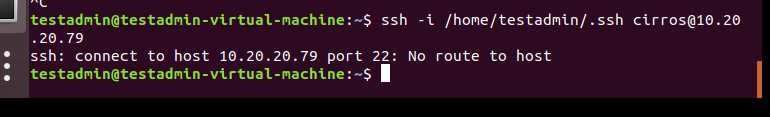
## 4.7 What was gained when collecting the results

At first glance, the achieved results seem far-fetched from the perfect results desired and very well detailed explanation, but when the problem insisted one had to make use of another Virtual Machine to establish a connection and to create a type of conversation. The conversation was in need to see whether Wireshark can collect the data. But in most of the tests, relevant data was not gathered because of the protection of the security features that OpenStack offers, which are Keystone and Nova. These are not all of them since there are even more upcoming tools which intend to provide a better and safer experience to the cloud user. Apart from the outcomes gathered, the most critical data that was gathered from this research were various. First of all, another important thing about the knowledge of the security features is, that a cloud environment can be safe to use for the clients. But this can only be achieved if the clients utilize the features such as Nova, Keystone, Bandit and many other features, which OpenStack offers for the client to feel safe in the cloud and work environment. The last and most important observation is the knowledge that was gathered on how to use OpenStack and how OpenStack works.

## 4.8 Expectations & Differences of the Result

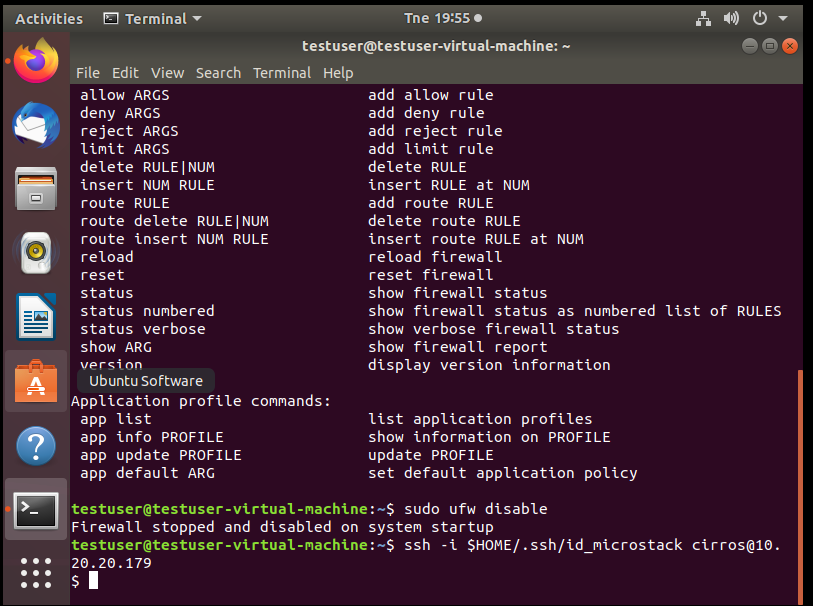
Unfortunately, the result that was expected was not achieved because of the obstacles previously mentioned. The challenge was that since the proper license was not purchased; therefore, only two instances could be used. The result should have shown a scenario where two instances and both instances send folders to each other. During this exchanging of data between two instances, Wireshark should have been activated to gather the information between the communication of the instances that should have been created on OpenStack. That was the expected result. But since this query was persisting, another Virtual Machine took place. Since the results were not the result that the researcher planned to gather, the outcome was not adverse at all - Wireshark was not able to gather the data from the three tests, and that is an outstanding result. Keystone and Nova were showing that the job the features are intended to do is being done successfully, and by blocking Wireshark and not letting the sniffer gather the data from the conversation of both the instance and the Virtual Machine, it is therefore a success.

## 4.9 Screenshots



### Figure 13: Error when tried to access by SSH

The error that is presented in figure 13 is indicating that there is no route to host the IP 10.20.20.79. The error cropped up because the address was wrong, with a wrong IP address. The address was supposed to be 10.20.20.179.



### Figure 14: Error when the researcher tried to create the second instance

The error that is presented as figure 14 presented itself because the instance could not be created. This error was met because of the problem that was caused without the purchase of license.

### 

### *Figure 15: Successfully Accessed the Instance via SSH*

Figure 15 shows that the instance which has the IP address of 10.20.20.179 was accessed successfully by an SSH connection.

### Figure 16: Error when trying to access the second instance via SSH.

Although the second instance was created from OpenStack directly, the instance was not able to be accessed by an SSH connection