# Results and Discussion

Throughout this chapter of the dissertation, the researcher will be analysing the results from the methodology which has been conducted before. Moreover, the researcher will also be discussing and comparing such results with one another.

Penetration testing is an effective type of method. This is since such method enhances and maximises network security. In addition to this, penetration testing limits certain and specific activities on the network. Penetration testing should be conducted by the company. The reason being is in order to safeguard the company’s data in every possible way.

The initial process of the prototype was to successfully gain admission to the access points which are unreachable by the penetration test. When this was accomplished, phase two of the testing evoked. From such a phase, a network scan took place in order to retrieve all possible targets. Apart from this, during such a phase, the penetration tester conducted numerous tests to detect and identify the vulnerabilities that the system contains.

All the necessary results gained were obtained from two diverse networks which are the secured access point and open access point with MAC filtering. The secured access point was protected with Wi-Fi protected Access 2 – Pre-shared key authentication. In contrast to this, the second network was an open network with MAC filtering. Such network allowed only specific devices to connect to it.

The results were conducted efficiently through the GPS antenna attached to the drone. The GPS antenna had very good signal which helped to keep the drone steady at the same position. Throughout such process, nobody controlled the drone whilst the attacks where performed.

## Access Point passphrase cracking

### WPA2-PSK 4-way-handshake attack :

The attack commenced when the drone was at the testing site and was within the signal of the access point. To automate an Aircrack-ng attack on the access point, a 4-way-handshake script was used. In addition to this, the Aircrack-ng attack was done an access point with BSSID of 2C:99:24:65:84: A9 and ESSID ARRIS-84AB. As seen in Figure 4.1, this was beneficial to obtain the passphrase in an encrypted format.

As a part of the attack, a de-authentication attack was included. Figure 4.2 highlights that one client was chosen with MAC address FC:DE:90:79:64:A9. Such client was selected to force the device into reconnecting to the network again. The MAC address of the client depicts that FC:DE:90 is the organizationally unique identifier of Samsung Electronics. Therefore, it is concluded that at least one Samsung device was being used on the network.

When the handshake was captured as shown in Figure 4.2, the script was stopped and the encrypted format password was saved in a .cap file. Then it was compared to a pre-defined list and the encryption was discovered. The pass of the encryption was pass12345. In order to obtain the password, it took a lot of time to discover the passphrase. This was due to the complexity of the password as it contained both letters and numbers. As stated by Touchette Barbara; Huson, Mark L., T. H. (2012), around 25% of the password attacks were conducted in different studies which all used the dictionary attack approach. This shows that most of the users tend to choose weak passwords which makes it easier for them to remember.

A pre-defined list of passwords which can be obtained online as compared to the hash gathered by Aircrack-ng. The password gathered was obtained without the need of the crunch command as can be seen in Figure 4.3. If there is any possibility in which the password was not obtained due to the complexity of the password, the crunch command was used with specific added parameters. This can be done in order to produce a custom-made word list.

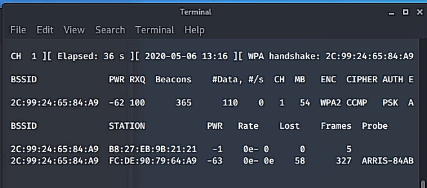
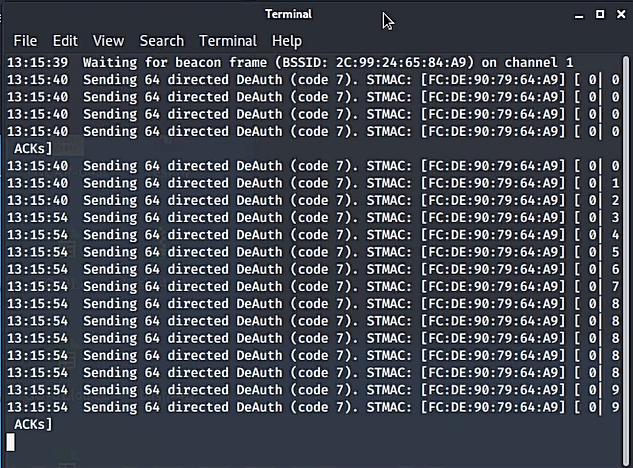


Figure 4.1 - The captured handshake



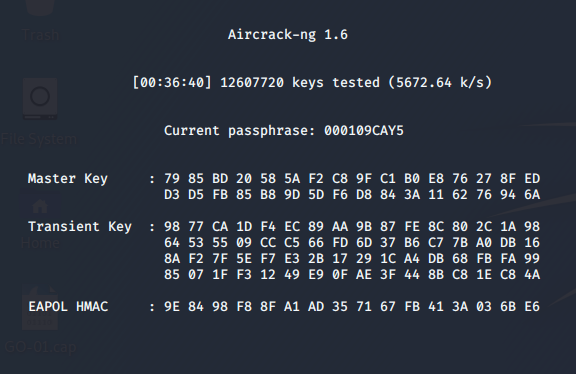
Figure 4.2 - De-authentication attack on client

Figure . - The crunch command being used to find passwords which are much more complex

### Evil Twin attack :

To verify and compare the results, the test was repeated by using another tool named the evil twin attack. The evil twin attack successfully retrieved the passphrase when a client connected to the rogue access point and wrote the network passphrase into a fake webpage.

From the use of the fake access point, the client was successfully tricked into thinking that the password of the access point was needed to connect to the Wi-Fi network. During such process, the attacker was eavesdropping from the victim. Through the help of Airgeddon and Fluxion scripts, the automation of the attack was much simpler. When the attack was done, the passphrase was collected into a log file and was saved where the password will be found.

The conducted experiment performed as was expected. This is due to the fact that the access point named ARRIS-84AB-1 was chosen. Apart from this, the manufacturer ARRIS Group, Inc was listed. This can be seen in Figure 4.4. A benefit from the information gained is that penetration tester gained reliable information regarding the construction of a fake web page. To strengthen this point, the script allows the possibility that the fake web page will copy a genuine webpage which is offered by the manufacturer.

The creation of an SSL certificate offered a fake secure environment which made the attack much more effective. The client believed that the webpage was authentic and so when connected to the access point, the client entered the password without any hesitation. Similar to the attack conducted by Aircrack-ng, a de-authentication attack was used. This was done to connect all the clients to the genuine access point which then, they would disconnect and connect back again with the rogue access point. Furthermore, the IP address 192.168.254.100 was allocated to the connected client. In this case, the connected client was a Galaxy-Tab-A with FC:DE:90:79:64:A9. The MAC address is shown in Figure 4.5 by a DHCP server.

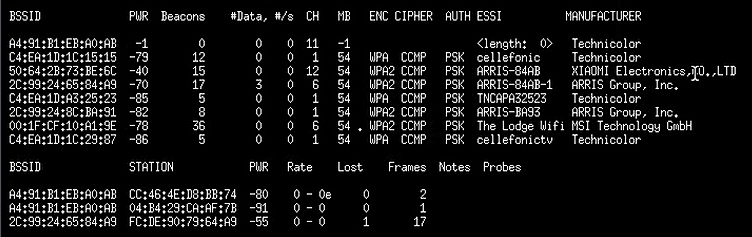
At the end of the script, a log file was constructed to store the password. Due to the successful attack, the password for the network was pass12345 as displayed in Figure 4.6.

Figure 4.4 - All available access points listed

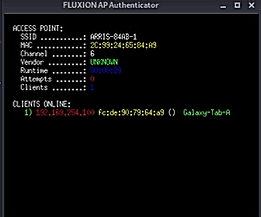


Figure 4.5 - AP authenticator with connected client

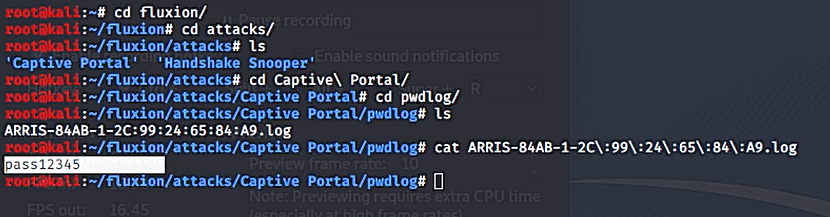


Figure 4.6 - Successfully obtained the password

Table 1 – Comparison between attacks

|  |  |  |
| --- | --- | --- |
|  | **Aircrack-ng (Dictionary attack)** | **Evil Twin Attack** |
| **Script Setup** | The script was initiated where the strongest AP was chosen automatically | The script was deployed, and the penetration tester had to choose the desired AP |
| **Access Point BSSID** | 2C:99:24:65:84: A9 | 2C:99:24:65:84: A9 |
| **Client MAC** | FC:DE:90:79:64:A9 | FC:DE:90:79:64:A9 |
| **Success?** | It was successful | It was successful |
| **Time is taken for an attack** | To obtain handshake 5 minutes, the time taken for an attack to crack the password took around 9 hours | The total time taken for the evil twin attack to crack the password was around 7 minutes. |
| **Efficiency** | Such an attack is more efficient as it is much simpler and is user friendly.  A downside of this attack is that although it is efficient, it takes a lot of time to construct such an attack. | Such an attack is much more complex to obtain the password.  In comparison to the Aircrack-ng, the Evil Twin attack took fewer minutes to perform. |
| **Output Password** | The output password was displayed on the screen | The output password was saved in a log file |
| **Handshake needed?** | Handshake is required | Handshake is required |

## Network Scanning

### Nmap Simple Scan

The first network scan which was performed was a ping Nmap scan. This scan conducted a ping of all the addresses that were answering to ICMP. This was done to gain a clear and vivid view of the topology of the hosts using the network. Moreover, the IP addresses discovered in this attack could be used in another. The Nmap scan discovered five hosts when the script was deployed after connecting to the access point. From the Nmap scan, a text file emerged which was saved with the output of the scan being shown in Figure 4.7. The IP address 192.168.0.1 was the gateway of the network under test. From such tests, the ARRIS device highlighted that it had four TCP open ports and two filtered ports. From this host, it was noticed that port 80 and 443 were open. To further strengthen the test, an internet browser was used and the address 192.168.0.1:80 searched as depicted. In addition to this, the default and main router login page were shown once the address was searched.

Moreover, an online database provided by ‘routerpasswords.com’ was used. Its manufacturer was searched, and the default login credentials were listed.

From the tested credentials, it was discovered that the login was successful. This employs that it is a major vulnerability in the network which needs to be highly addressed. Moreover, both the universal plug and play port were also open. Another vulnerability of such a network is that various and numerous attacks can be conducted on such services. For this reason, this might even cause damage to the internal network. To strengthen the security technique for this device against a harmful attack is to close all the ports which are not required or needed.

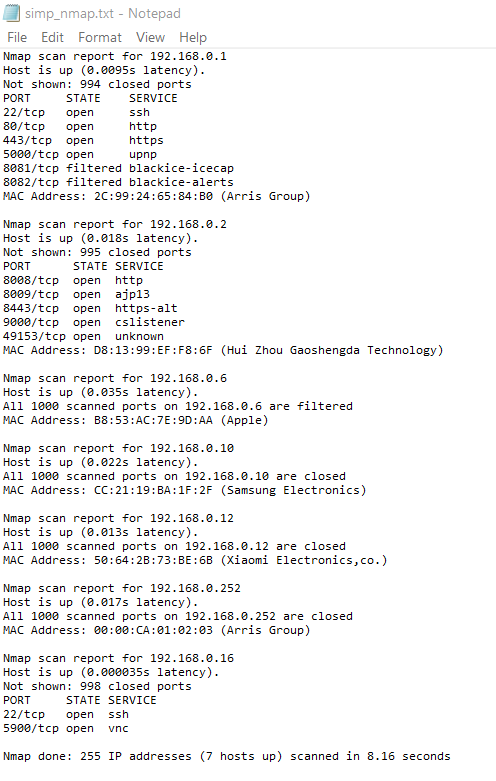


Figure 4.7 - Simple Ping Nmap Scan

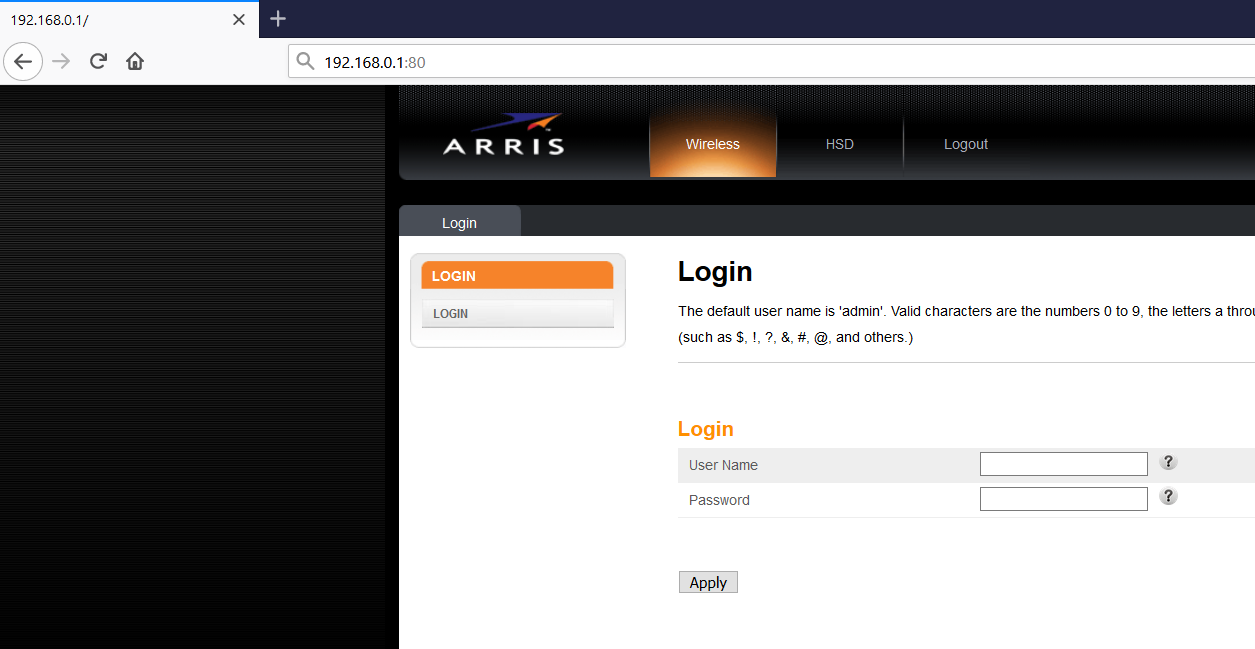


Figure 4.8 - Router Login page

### Vulnerability Nmap Scan

To obtain any vulnerabilities of the connected devices, the Vuln script was used. Such script is included in Nmap scan which makes it easier to scan and find these types of vulnerabilities. The Nmap scan offered the ability to check whether the network was set up to standards. These vulnerabilities called holes was found by using a script which is included in Nmap. From the results obtained, the HTTP port of the access point which was being tested had two locations. This evoked the possibility of a backup which could be stored. Furthermore, the scan discovered that an HTTP-SQL-injection attack could be conducted by two dissimilar queries which are found in figure 4.9. To be able to successfully implement HTTP-SQL-injection, one must use a malicious SQL statement to control the database of the device behind the web application. Similar results as the port 80 can be seen in the scan of port 443. Both the universal plug and play port were vulnerable to three versions of the denial of service attack. These were discovered by the scan as seen in Figure 4.9.

A Linux operating system was used by the access point. The other devices which were discovered consisted of various devices such as Apple, Xiaomi and another Linux device as depicted in Figure 4.10.



Figure 4.9 – Vulnerability Nmap scan of 192.168.0.1

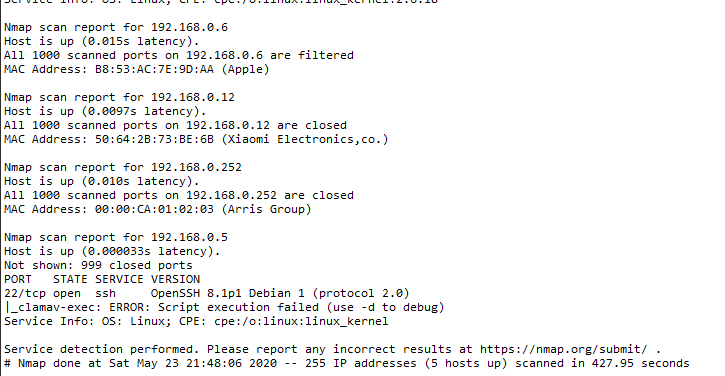


Figure 4.10 - The devices discovered by the Nmap scan

Table 2 - Network Scan results

|  |  |  |
| --- | --- | --- |
|  | **Simple Nmap Scan** | **Vulnerability Nmap Scan** |
| **Test successful?** | Yes | Yes |
| **Time** | The simple Nmap scan takes minimum time. Hence, the penetration tester could use the information obtained in the next attacks. | The vulnerability Nmap scan takes more time to check and establish all vulnerabilities. |
| **Looking for targets** | In order to look for targets, such scan created a simple list of all the required hosts and ports. | In order to look for targets, this scan created a list with known vulnerabilities of the hosts. |
| **Use** | This scan is used to identify the topology of the network behind the access point. | This scan is used by a penetration tester. This tester can conduct any suggested tests on the host indicated in this scan. |
| **All hosts discovered?** | Yes | Yes |

## Man-in-the-middle

A man-in-the-middle attack was conducted after collecting a list of all the hosts from the previous simple Nmap scan. The script intended for this attack was successfully implemented. The man-in-the-middle attack initiated by taking the IP addresses of both access points. In addition to this, as pointed out in Figure 4.11, the man-in-the-middle attack included a client who entered at the beginning of the script. The IP address of the access point was 192.168.0.1, the gateway of the network and the client access point was 192.168.0.6. Once this information was obtained and entered by the penetration tester, the attack emerged. The Raspberry PI began to collect the packets which were sent from the two nodes. For this attack to be successful, two security issues such as ARP request and ARP reply.

The ARP request was used to tell any device that it is in the network and that the Raspberry PI is the gateway. On another hand, the ARP reply was used to allow the Raspberry PI to respond to ARP requests from devices under test.

The attack was left to run for around two minutes and resulted in three different results. As illustrated in Figure 4.12, this scan was conducted to see network traffic. The result of this command was a list of all the networks traffic being generated between the AP and the client. Furthermore, it was noticeable that the client used ‘google.com’ to enter into ‘facebook.com’. This analysis can be seen in Figure 4.15. This command is also useful in order to monitor any activity that takes place on the network in real-time.

The ‘S’ and ‘F’ flags can be seen where they are matched to random destination ports. The S flag shows that a connection has started. The connection is an SYN scan. On another note, the F flag is a Fin flag which is shown as a connection finish flag. As seen in Figure 4.16, one can notice that a mixture of internal and external IP addresses has been found.

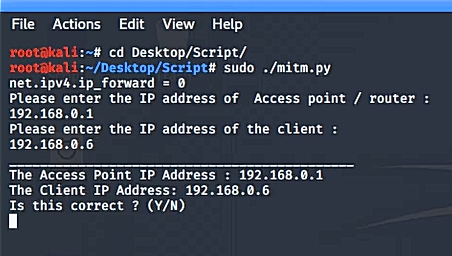
The server IP address which has the most hits is the address 31.12.86.36 and that has a total number of 42 hits. This IP address is used by’ facebook.com’ which verifies that the first scan conducted offered valid results when compared together. The final command used was the urlsnarf. This was not able to record any packets that have been passing inside the network. The main use and objective of urlsnarf are to provide results accordingly. This can especially be seen when using an old operating system such as Windows XP or Windows 7. Through the use of this attack, communication is intercepted and there is a possibility where the communication can be modified with further tools.

Figure 4.11 - Targets for the man-in-the-middle

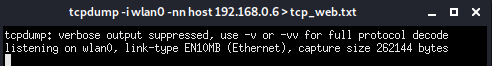


Figure 4.12 - TCPdump to capture any port 80 traffic

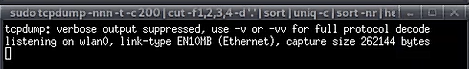


Figure 4.13 – TCPdump used to collect any external IP addresses used



Figure 4.14 - URLsnarf to capture any websites being used

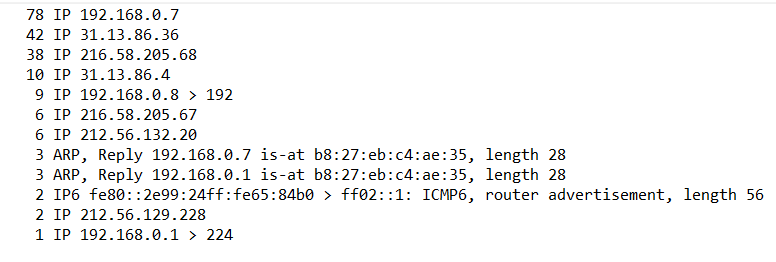
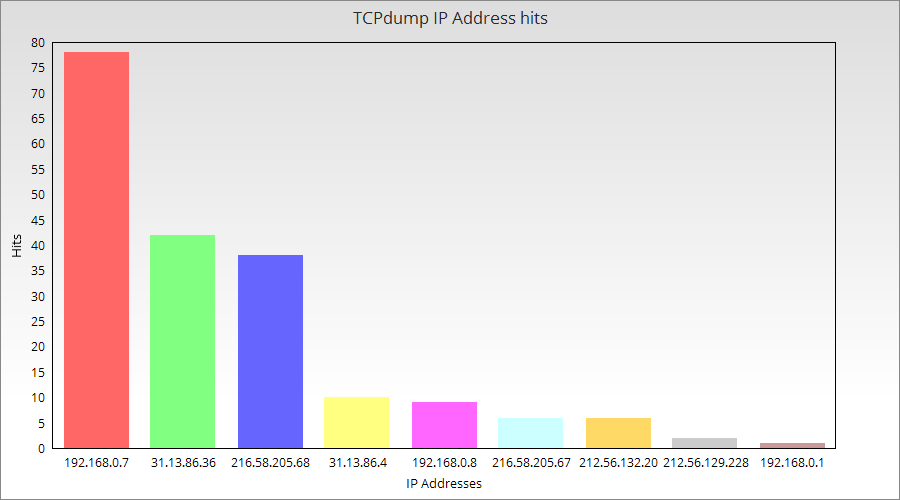
Figure 4.15 - Result of tcpdump -nn command

Figure 4.16 - IP addresses used

Table 3 – TCP Statistics

## Mac Spoofing Attack

Mac Spoofing main objective was to change the MAC address of the WI-FI adapter on the Raspberry PI as necessary. For the purpose of this attack, an open network was implemented with MAC filtering. Although this network was open, the Raspberry PI was not able to connect to the access point. At this stage, it was clear that some kind of restrictions were being used against certain MAC addresses. With the use of a script created for this function, it was possible to check what the clients are using on the network. This was needed in order to obtain one available client MAC address and save it. In addition to this, this was needed to change the MAC address of the Raspberry PI into the one being used by the client. Hence, the client connected had a MAC address of B8:53:AC:7E:9D:AA. The original MAC address of the Raspberry PI network adapter was B8:27:EB:C4:AE:35. As can be seen in Figure 4.15, the mac address was changed temporarily. For this reason, the connection to the Access Point was tested again and it was successful. After the attack was conducted, the MAC address was reversed back into its original state. This is done in order to be changed again if needed when another attack was conducted. As suggested by Touchette  Barbara; Huson, Mark L., T. H. (2012), it might be challenging to discover such attack. Hence, one can make use of Reverse Address Resolution protocol which can be compared to the MAC address. The Reverse Address Resolution protocol is used to check if the client using the services are legitimate or malicious. Furthermore, Sweigart, C. (2003) also highlighted that MAC address Spoofing offers multiple advantages when conducting penetration testing. This will allow untraceable scans from both nodes of the network and also redirect network traffic as needed.

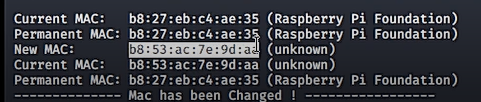


Figure 4.17 - MAC Address Change

## Concluding the Penetration testing using the drone.

The tests conducted were done in order to obtain how a drone can be used to penetration test a network. Once all tests where done, the data was placed into different text files. The files were then transferred from the Raspberry PI using an FTP server. Moreover, the data collected were merged into a document. This document provides a user-friendly representation of all of the results obtained during tests, which were carried out by the Raspberry PI and the drone. The results obtained outlines that the access point and network under test need to be made more secure and less vulnerable to attacks. The best practice for protecting the network and access points is to implement a third-party wireless intrusion detection system. According to Avenue, N. B., & View, M. (2006), through such a system one can minimize the risk of such attacks. This system can detect and stop different forms of attacks before they are applied and harming an enterprise. An important aspect is that all of the vulnerabilities in the network are addressed. Using the results obtained from the penetration tests done above, a vulnerability assessment can be conducted. This includes a scale of highest and lowest vulnerabilities. The results of the assessment are tabulated in order to allow the owner of the network to first fix the most important vulnerabilities with the recommendations and mitigations to remove the risks of an attack.

**Bibliography**

Abramov, E., Kobilev, M., & Makarevich, O. (2013). Using quadrocopter as a pentest tool. *SIN 2013 - Proceedings of the 6th International Conference on Security of Information and Networks*, 404–407. https://doi.org/10.1145/2523514.2527019

Avenue, N. B., & View, M. (2006). *Best Practices for Securing Your Enterprise Wireless Network Best Practices for Securing Your*.

Cangea, O. (2018). *to Defeat Cyber Attacks*. *LXX*(2), 29–37.

Gandhi, C., Suri, G., Golyan, R. P., Saxena, P., & Saxena, B. K. (2014). Packet Sniffer – A Comparative Study. *International Journal of Computer Networks and Communications Security*, *2*(5), 179–187.

Ghering, M., & Poll, E. (2016). *Evil Twin vulnerabilities in Wi-Fi networks*. http://www.cs.ru.nl/bachelorscripties/2016/Matthias\_Ghering\_\_\_4395727\_\_\_Evil\_Twin\_Vulnerabilities\_in\_Wi-Fi\_Networks.pdf

Murphy, B. F. (2013). *Network Penetration Testing and Research Major : Computer Science Program USRP Summer Network Penetration Testing and Research*. 0–11.

Sweigart, C. (2003). *Interested in learning more ? In sti tu te , A ut ho ins ll r igh*. *Security 401*, 1–39. https://www.giac.org/paper/gsec/3907/introduction-computer-security-incident-response/106281

Touchette  Barbara; Huson, Mark L., T. H. (2012). Password Security - What Factors Influence Good Password Practices. *Proceedings of the Southwest Decision Sciences Institute (SWDSI) 2012 Conference*, *March 2012*, PA116.