Exploration of Password Length on Wireless Network Security

James C. Duvall

Western Kentucky University

Abstract

In this research project, the topic of password length within the context of wireless networking security will be explored. An experiment will be undertaken that will use available hacking tools and modest equipment to attempt to gain access to a lab wireless network. The lab wireless network will be built to mimic a production wireless network that could be found in a small business setting. Quantitative data collected from this experiment will be used to produce summary statistics that will then be used to demonstrate the degree of impact password length has on the security of the wireless network. This project will also explore the feasibility of success an amateur hacker may experience with the previously mentioned tools and equipment using information obtained in online articles detailing the use of these tools.

Exploration of Password Length on Wireless Network Security

Background

Almost all computer systems today, no matter how advanced or remote, share one common component, passwords. Passwords are the basic form of securing a system or the data that it holds. Passwords on computer systems dates back to the early 1960s. At that time, Fernando Corbató was in charge of the CTSS computer at MIT. The computer was designed for multiple users to share access and store their individual files. This design, however, lead to issues. David Kalat (2018) explains that "because each user had his own research domain and files, the CTSS needed to be able to distinguish between them and to present them with only the materials authorized to them" (The first password section, para. 1). Corbató came up with a solution to the issue, explaining that "putting a password on for each individual user as a lock seemed like a very straightforward solution" (as cited in McMillan, 2012, para. 7). Since that time, passwords have become an important part of modern computing.

Since passwords have become such an important part of computing, they have naturally become a highly sought-after prize for hackers all over the world. Over the past several years, massive security breaches have sadly become a normal part of Internet-connected life with millions of user's data exposed to malicious threat actors, including passwords and other personal information. Last year, an anonymous actor uploaded massive files full of usernames and passwords from past breaches online for other hackers to download. Andy Greenberg (2019) describes the event, stating that "someone has cobbled together those breached databases and many more into a gargantuan, unprecedented collection of 2.2 billion unique usernames and associated passwords and is freely distributing them on hacker forums and torrents" (para. 1). While one would hope that users are aware that their passwords were compromised and changed

them, this information still gives hackers clues into how people create passwords and how popular some passwords are.

Password policies and best practices have tried to make passwords more secure to prevent hackers from obtaining them. These policies and practices have focused on password length and complexity, frequent changes, using unique passwords for each account, and using randomly generated passwords. The problem with these policies is that they have made it more inconvenient than many want to tolerate so they find workarounds to these policies. People reuse passwords when forced to change them. They rarely use randomly generated passwords because they are difficult to remember. Arguably, the most detrimental to the security of their data has been circumventing complexity requirements by using weak passwords that are minimal length and follow well-known and predictable patterns.

The use of these kinds of weak passwords in one specific area of information technology is what this project will explore, wireless networking. Wireless networking, as the world knows it today, came about it the late 1990's with the IEEE 802.11 wireless networking standard. Wireless access points (AP) could be placed around the workplace or home and data could be transmitted between the AP and a computer with a wireless network interface card that had associated with the AP. What was quickly apparent, however, was that wireless transmissions on these open networks could also be captured by individuals who were not meant to receive them. A technology had to be implemented that would prevent this kind of eavesdropping and stealing of data. This is where Wired Equivalent Privacy (WEP) came in. WEP was designed to encrypt the data as it was sent across the radio waves. Alissa Irei (2019) says as much when explaining WEP's origins, stating that "the Wi-Fi Alliance developed WEP – the first encryption algorithm

for the 802.11 standard – with one main goal: prevent hackers from snooping on wireless data as it was transmitted between clients and APs" (Wired Equivalency section, para. 1).

In theory, anyone who did not have the encryption key would not be able to decipher any packets that they managed to capture coming from or going to the AP. The encryption keys would be input into the AP and clients that would associate to the AP. This key was combined with another value called and initialization vector (IV). The IV was an additional 24 bits, which made the key longer and, thus, more difficult to obtain. What made using the IV even more beneficial was that it was designed to change incrementally with each bit of data sent. David B. Jacobs (2008) explains that "the IV strengthens encryption by causing successive packets to be encrypted with different keys, making it more difficult for a hacker to determine the configured key" (RC4 operation section, para. 1).

It wasn't long after WEP was implemented that vulnerabilities were found and began to be exploited. One of the biggest vulnerabilities is unfortunately the IV itself. The issue with the IV is that there are only so many combinations of 24-bit keys that can be used before they start to repeat themselves. Denis Laverty (n.d.) states "24 bit keys allow for around 16.7 million possibilities. Sounds a lot, but on a busy network this number can be achieved in a few hours. Reuse is then unavoidable" (IV length is section, para. 1). If multiple packets using the same IV can be captured, modern password cracking tools can be used to attempt to break the encryption and obtain the keys.

To overcome the vulnerabilities of WEP, another security standard had to be developed. Wireless Protected Access (WPA) was the immediate successor to WEP, but the current implementation today is its successor, WPA2. WPA2 comes in two main implementations, WPA2-Personal and WPA2-Enterprise. The Enterprise version uses different authentication

techniques and is not explored in this research. For this project, whenever WPA2 is mentioned, it is in reference to WPA2-Personal.

WPA2 uses much stronger encryption, AES with CCMP, to secure data as it is sent between the AP and a device. WPA2 with AES and CCMP is considered quite secure in its operation but that does not mean it doesn't present some opportunities to would-be attackers. When devices associate with an AP using WPA2, a four-way "handshake" is used to establish the connection. During the handshake, the device and the AP exchange four messages that establish the association and exchange encryption information, including the password in hashed form. Capturing the handshake and the hashed password is typically the goal of a hacker who is trying to break into the network.

Once captured, the encrypted password faces its greatest adversary, time. Given enough time, any encrypted password can be uncovered. There are two main types of attacks to uncover this data, dictionary attacks and brute-force attacks. Dictionary attacks work by testing the hashed password against a file of possible passwords. The attacking device creates a hash of each line in the file and compares it to the stolen hash to see if they are a match. If it matches, the attacker has the password. If not, the attack moves down through the list until it finds the match, or it reaches the end of the list. If the password is in the list, it will be found after enough time. A brute-force attack tries every combination of characters until it finds a match to the hashed password. While this type of attack is guaranteed to work, the time it takes to do so may not be feasible or even possible in one's lifetime. The time it takes to complete this type of attack is directly related to length and/or complexity of the password.

As previously discussed, complex, random passwords can be difficult for users to remember so they are typically avoided. This leaves length in many cases as the sole determinant

of password strength against brute-force style attacks. This project will explore the impact that password length has on WPA2 security. Through experimentation and statistical testing, this project will show how increasing lengths of WPA2 passwords will require longer timeframes to obtain the encrypted password. This project will also explore the feasibility of success a novice hacker could have against a wireless network that is secured with a weak password.

Goals

The main goal of this research project is to document the data resulting from experimentation and present summary statistics using that data to show the degree of impact password length can have on WPA2 encryption effectiveness. The experiment will produce quantitative data about the performance of the computing device used during the experiment that will indicate how many hashed passwords can be tested by the system over a certain length of time. This data will then be used to calculate the longest amount of time it would take the same system to obtain WPA2 passwords of differing lengths. These summary statistics will be presented in tabular format for easier comprehension by the reader.

A secondary goal of this project is to document the process used for performing the experimentation and report the degree of success in obtaining the WPA2 password from a simulated small business network using freely available hacking tools and instructions on the Internet and basic computing equipment similar to equipment a novice hacker would have at their disposal. The experiment will approach the objective of obtaining the WPA2 password from the perspective of a new and inexperienced hacker. Basic, inexpensive equipment with mid-grade hardware and online tutorials will be utilized similarly to what an inexperienced hacker could theoretically use. This lab and approach will simulate a real-life threat faced by

small business wireless networks and provide insight into the degree of difficulty of obtaining a weak WPA2 password.

Methodology/Approach

As previously stated, this project will use an experimental approach to deliver the results needed to meet the stated goals. While it is possible to find potential data that could be used for the summary statistics from secondary sources, this data may not provide the same accuracy that primary data will. In his book, Colin Robson (2014) stated that an experimental approach "provides the possibility of getting clear and unambiguous answers to very specific research questions" (p. 29). To obtain the clear and accurate data that the goals of this project require, the experimental approach was chosen.

For the experiment, a lab network will be set up to simulate a small business wireless network. The lab will use a Cisco Small-Business Class router with two-to-three devices connected to the network. The router will be configured for WPA2-Personal security with a simple, common password. Many of the most common passwords found in breaches from previous years have been analyzed and released online. Several of the top passwords that are compromised are made up entirely of numbers. According to the National Cyber Security Centre (2019), "Breach analysis finds 23.2 million victim accounts worldwide used 123456 as password" (Introduction section, para. 1). Since this is the most commonly stolen password in the world, it is highly likely that there is a wireless network with something similar securing it. Since WPA2 requires an eight-character password, "12345678" will be used as the WPA2 password on the lab network. This password will help meet both goals by providing the performance statistics needed for analysis and facilitating the feasible breaking of the WPA2 password.

Once the lab network has been set up, the next step is to obtain the WPA2 four-way handshake. To do this, a laptop running a live USB boot of Kali Linux will be used. The Aircrack-ng suite of tools comes preinstalled on Kali and will be used for the capture. The online tutorial will provide step-by-step instructions on how to capture the handshake using Aircrack and the captured data will be saved into a file to later perform an offline password attack.

The file with the captured handshake will be transferred to a Windows desktop where Hashcat will be installed. Hashcat is considered one of the fastest password-cracking tools in use today. It can use the graphics card to process hashes instead of the CPU. Before Hashcat can be used to obtain the password, the file must be converted to a format that Hashcat can use. The file will be saved in a pcap format. Hashcat requires files to be in a hccapx format. To convert this file, the makers of Hashcat offer an online tool called Cap2hccapx. With this tool, anyone can upload a pcap file and the tool will convert it to the required hccapx format. The tool can be downloaded and compiled to use locally, but for this purpose uploading the file is much more efficient.

Once the file is converted, it will be used in Hashcat to obtain the password. An online tutorial will be used to provide instructions for using Hashcat. The attack that will be performed will be a variation of a brute-force attack known as a Mask Attack. A Mask Attack is a more structured and specific attack than simply trying every single combination. With a Mask Attack, the attacker can specify the length of password to test plus specific types of characters to test in certain positions. For example, if the attacker wanted to test if the entire eight-character password was lower-case letters, they could use a mask of "l l l l l l l l" (all lower-case Ls). Hashcat would test only combinations of lower-case letters until it found a match or tested all possible combinations. For the purposes of the experiment, a mask of all digits will be used.

Even though the mask and password are simple, they will provide the required data and answers that this project is exploring. Completing the experiment and obtaining the WPA2 password will meet the secondary goal of this project.

Once the password is obtained, quantitative data from the experiment will be recorded. Of primary concern is the performance metric of the testing system. This value should be presented in a hashes per second or similar format. This value will be used to calculate the time it would take to run other mask attacks of all numbers at differing lengths. To calculate the maximum length of time to run a mask attack, several pieces of information are needed. First, the total number of possible combinations needs to be calculated. To do this, the number of character spaces in the password has to be determined. Next, the number of possible characters that can occupy a character space must be determined. After determining these two values, the formula P^S, where P equals the number of possible characters that can occupy a space and S equals the number of character spaces, can be computed. For example, WPA2 requires a password of eight characters as a minimum, which means eight character spaces. If only lower-case letters were used for the password, there are twenty-six possible characters per space. The equation would then be 26^8, or 208,827,064,576 possible combinations. Once the possible combinations are known, the performance metric can be used to calculate the total time the system would need to test these combinations. These calculated values will be used to meet the primary goal of this project.

Deliverables/Expected Results

To meet the main goal of this project and present evidence that demonstrates the impact password length has on WPA2 security, a table will be produced containing summary statistics based on data from the experiment. As mentioned above, quantitative data recorded from the

experiment will be used to calculate mask attack efficiency based on varying lengths of passwords. The expected results of these calculations are to show that as password length increases, so does the maximum length of time it would take to test all combinations of characters that possibly make up the password. What is expected to be clear to the reader is that longer passwords can decrease the likelihood of an attacker breaking into the wireless network significantly.

To meet the secondary goal of this project, the WPA2 password must be obtained during the experiment using the basic hardware and free software detailed in the experiment design. The experiment was designed to emulate an inexperienced attacker. This was determined to be someone with basic equipment and little knowledge about how to hack into a wireless network. Doing simple online research would lead them to software like Hashcat and Kali Linux. Online research will also provide them with many articles and YouTube videos detailing how to use these tools with simple step-by-step instructions. The objective is to test whether one of these inexperienced attackers could use these tools and instructions to perform a real-life hack using basic, mid-priced equipment they most likely already own. Using additional or upgraded equipment to obtain the password will result in failing to meet this goal.

Requirements/Resources

There are several resources that are necessary for this project to be completed. Most of these resources can be broken down into two categories, hardware resources and software resources. One of the most important hardware resources is a small business-class router that will be used to set up the experiment lab. The router chosen for the experiment is a Cisco RV215W Small Business RV Series router. This router will provide an adequate simulation to what an actual small business would use in their production network. Another hardware resource that will

be used will be a basic Acer laptop computer. This laptop will be used to monitor and capture the wireless transmissions that will be used to obtain the WPA2 password. An ALFA network adaptor will also be used with the laptop for monitoring and capturing wireless transmissions. This adaptor works well with the Kali Linux distribution since it can monitor in promiscuous mode. A fourth hardware resource will be a basic desktop computer running Windows 10 with a midgrade GeForce GTX 1060 graphics card. The desktop will be used to break the WPA2 encrypted key using the graphics card to test matches.

There are also multiple software resources that will be required for this project. One of these will be the Kali Linux operating system. This will be running as a live USB boot on the laptop and, as stated earlier, will utilize the ALFA network monitor for transmission monitoring. Other software that will be needed are two penetration testing tools used for cracking passwords, Aircrack-ng and Hashcat. The Aircrack-ng suite of tools will be used within the Kali Linux system to capture packets and to obtain the WPA2 four-way handshake. Hashcat will be used on the Windows desktop to perform an offline Mask Attack using the GPU to process the attack. Hashcat was chosen due to the performance ratings that it receives from many websites. In speaking on the performance of Hashcat, Alex (2020) explains that “the peculiarity of hashcat is the very high speed of brute-force passwords, which is achieved through the simultaneous use of all video cards, as well as central processors in the system” (How to run section, para. 4).

Another resource that will be utilized will be online tutorials for using these tools to obtain wireless passwords. The first tutorial that will be used will be “Crack WPA/WPA2 Wi-Fi Routers with Aircrack-ng and Hashcat” by Brannon Dorsey from Hakin9.org. This tutorial gives step-by-step instructions on using Aircrack-ng on Kali Linux to monitor the wireless network and capture the four-way handshake needed to obtain the password. This tutorial does provide

some information about using Hashcat but are not detailed enough for an amateur hacker to be able to perform the attack. A second tutorial that will be used will be “Hashcat manual: how to use the program for cracking passwords” from MiloSerdov.org. This tutorial explains installing and using Hashcat on Windows to obtain a WPA2 password using a Mask Attack and the graphics card on the machine. One final resource that will be used is the online file converting tool, Cap2hccapx, that will convert the pcap file into a file type that Hashcat can use, as previously mentioned.

Additionally, there will be two critical success factors required in this project that will be used to measure the level of success in meeting the project goals. One of these success factors will be the performance metric of the test system recorded upon completion of the experiment. Without this metric, calculating the summary statistics presented in the table deliverable will not be possible. Without recording this data, there will be no evidence to present that demonstrates the impact password length has on wireless security and the main goal of the project will not be met.

The second critical success factor required by this project for successful completion will be the WPA2 password obtained during the experiment. Obtaining the password by performing the steps outlined in the project methodology is vital to meeting the two goals of this project. By obtaining the password during the experiment, the results will aid in demonstrating that shorter, weaker passwords can be obtained by novice hackers with basic equipment and skills and help validate the statistics presented in the main deliverable table.

Assumptions

Several assumptions can be made about this project before it is undertaken. One assumption is that the small business router that will be needed will be in stock by the vendor and ready to ship when ordered. It is also assumed that the delivery of the router will be within five days of placing the order. These assumptions are based on current vendor shipping and availability information for this product. Assuming this allows for project timeline and completion date projections because building the experiment lab must be completed before further work can be accomplished.

A second assumption for this project is that the software tools used for this experiment is compatible with the used systems and will function as intended. The Aircrack-ng suite of tools comes preinstalled on the Kali Linux operating system so it will be assumed that this will function correctly. There are several online tutorials and videos detailing the use of Hashcat on the Windows operating system so it will also be assumed that this will function correctly as well.

To use the graphics card for hash processing, Hashcat requires GPU drivers that support this function. As previously mentioned, the desktop will utilize a GTX 1060 graphics card, which uses an NVIDIA GPU for its operation. According to Hashcat.net (n.d.), “NVIDIA GPUs require ‘NVIDIA Driver’ (367.x or later)” (GPU Driver requirements section, para. 1). The desktop currently has driver version 442.59 installed. It will be assumed that the required driver version for Hashcat has not changed and that the currently installed driver has not been changed to adversely effect the use of Hashcat.

Another assumption that will be made is that the online tutorial used to complete the experiment portion of this project contains information that can be applied to currently available

versions of Hashcat and Windows operating systems. It will also be assumed that the information and step-by-step instructions provided in the tutorial are accurate and detailed enough to complete the research experiment and successfully meet the goals of project.

Additional assumptions that will be made for this project concern the web-based application that will be used to convert the file that will contain the captured handshake, Cap2hccapx. One assumption is that the web application will be available for use during the experiment. This will mean assuming that the web server that hosts the application will be operating correctly and accepting uploaded files to convert. Another assumption regarding the web application will be that it works as intended and will correctly reformat the file to the required hccapx file format needed for using Hashcat.

Limits and Exclusions

A limitation to this project, due to time and equipment constraints, will be that the experiment will feature a condensed simulation of a wireless network attack. In a real-world attack, the attacker would likely have no prior knowledge of what mask to use for the attack nor the actual length of the password. Without this information, it would be impossible to guarantee a reasonable chance for success within the amount of time available to complete this experiment. For the feasibility of obtaining the data needed to meet the goals of this project, a simple password and mask will be utilized that will allow the researcher to adequately test the system and generate the data.

There will also be some exclusions to this research project. One exclusion to this project will be that the summary statistics produced by the project will only show the maximum length of time that it would take to obtain the passwords of varying lengths using the selected mask

variation and computing equipment. The actual time it takes to find the match to the encrypted password will depend on many different variables, including available hardware, the actual mask used and the actual password among others. Another exclusion to this project is the wireless network security options that will not be tested. As mentioned previously, WEP, WPA, and WPA2-Enterprise are other wireless security technologies available to secure a wireless network. Due to time restraints, additional tools being needed, and, in the case of WPA2-Enterprise, advanced skills being needed, these options will not be explored here.

Milestones

There will be five major milestones that will be reached during this research project. These milestones will be receiving the Small Business router, the creation of the experiment simulation lab, obtain the WPA2 four-way handshake and save it to a file, obtain the WPA2 password, and produce the table needed to meet the main goal of the project.

The projected start date for the project is April 6th, 2020. This will allow two weeks after submitting the proposal for obtaining approval for the project and feedback. On this expected date, the router can be placed on order. According to current vendor shipping information, this will take approximately five business days. This puts the projected completion date of the first milestone, receiving the Small Business router, at April 10th, 2020.

The second milestone, creating the experiment lab, will be given an estimated completion timeframe of three days. This will allow time for gaining familiarity with the interface to configure the router, configuring the correct settings and associating devices with the AP for use. Lab creation is estimated to begin on April 13th, 2020 due to dependence on the completion of the first milestone and will have a projected completion date of April 15th, 2020.

The third milestone, obtain the WPA2 handshake and save it to a file, will be given a projected completion timeframe of one day. Once the lab has been created, the experiment can begin. The laptop will be used to monitor traffic to and from the AP to obtain the handshake. A specified file to place the information will be created during the attack to hold the results of the attack. Due to the dependence of the second milestone being completed, this task is projected to begin and be completed on April 16th, 2020.

The fourth milestone, Obtaining the WPA2 password, will be given an estimated timeframe of one day. Based on estimates of the performance of the GTX 1060 graphics card that will be used in this experiment, it is estimated that obtaining the selected password using the know mask pattern can be accomplished within one day. Since completing this milestone depends on the completion of milestone three, the projected start and completion date of this milestone will be April 17th, 2020.

The fifth milestone, producing the table that will meet the mail project goal, will be given a project timeframe of two days. The quantitative data resulting from experimentation can be collected once milestone four is completed. Once this is collected, the analysis and statistical testing can be completed. The results from this testing will then be placed into a created table for final project reporting. This task cannot begin until milestone four has been completed, so the estimated start date for this task is April 20th, 2020. The estimated completion date for this milestone will therefore be April 21st, 2020.

Once all milestones have been completed, work on the final project report can begin. To make the correct changes to the project proposal, changing estimated and expected information to actual information obtained over the course of the project, will take an estimated five days.

During this time, the data table created as a result of completing the fifth milestone will be added to the report with sufficient detail and explanation for the reader. Details and results from the experiment will be added as well. The projected completion date of entire project is April 28th, 2020.

Figure 1 below shows the Gantt chart created during the project planning phase. This chart lists the project tasks and milestones that will be completed during the course of the project, the start and finish dates and a visual representation of the path to completion.

	Name	Duration	Start	Finish
1	Project Start	0 days	4/6/20 8:00 AM	4/6/20 8:00 AM
2	SB Order and Receive SB Router	5 days	4/6/20 8:00 AM	4/10/20 5:00 PM
3	Milestone: Receive SB Router	0 days	4/10/20 5:00 PM	4/10/20 5:00 PM
4	Create Experiment Lab	3 days	4/13/20 8:00 AM	4/15/20 5:00 PM
5	Milestone: Experiment Lab Created	0 days	4/15/20 5:00 PM	4/15/20 5:00 PM
6	Obtain Handshake and Save File	1 day	4/16/20 8:00 AM	4/16/20 5:00 PM
7	Milestone: Handshake Saved to File	0 days	4/16/20 5:00 PM	4/16/20 5:00 PM
8	Obtain WPA2 Passphrase	1 day	4/17/20 8:00 AM	4/17/20 5:00 PM
9	Milestone: Passphrase Obtained	0 days	4/17/20 5:00 PM	4/17/20 5:00 PM
10	Produce Table w/Summary Statistics	2 days	4/20/20 8:00 AM	4/21/20 5:00 PM
11	Milestone: Table Produced	0 days	4/21/20 5:00 PM	4/21/20 5:00 PM
12	Complete Project Report	5 days	4/22/20 8:00 AM	4/28/20 5:00 PM
13	Project Complete	0 days	4/28/20 5:00 PM	4/28/20 5:00 PM

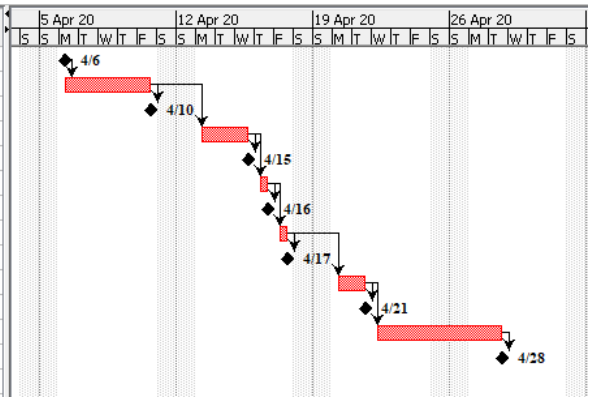

Figure 1: Gantt Chart of Research Project with Task List and Start and Finish Dates

Bibliography

Alex. (2020, Mar 13). Hashcat manual: how to use the program for cracking passwords. Retrieved from https://miloserdov.org/?p=953

Greenberg, A. (2019, Jan 30). Hackers Are Passing Around a Megaleak of 2.2 Billion Records. Retrieved from https://www.wired.com/story/collection-leak-usernames-passwords-billions/

Hashcat.net. (n.d.) Hashcat. Retrieved March 20, 2020, from https://hashcat.net/hashcat/

Irei, A. (2019, Sept). Differences among WEP, WPA and WPA2 wireless security protocols. Retrieved from https://searchnetworking.techtarget.com/feature/Wireless-encryption-basics-Understanding-WEP-WPA-and-WPA2

Jacobs, D.B. (2008, Feb). Wireless security – How WEP encryption works. Retrieved from https://searchnetworking.techtarget.com/tip/Wireless-security-How-WEP-encryption-works

Kalat, D. (2018, Nov 4). The History of Passwords and the Case of the First Theft. Retrieved from https://www.thinksetmag.com/issue-6/the-case-of-the-purloined-password

Laverty, D. (n.d.). Why is WEP crackable? How WEP weaknesses affect your wireless network security. Retrieved from https://openxtra.org/article/wep-weaknesses

McMillan, R. (2012, Jan 27). The World's First Computer Password? It Was Useless Too. Retrieved from https://www.wired.com/2012/01/computer-password/

National Cyber Security Centre. (2019, Apr 21). Most hacked passwords revealed as UK cyber security survey exposes gaps in online security. Retrieved from

https://www.ncsc.gov.uk/news/most-hacked-passwords-revealed-as-uk-cyber-survey-exposes-gaps-in-online-security

Robson, C. (2014). *How To Do A Research Project: A Guide For Undergraduate Students* (2nd ed.). West Sussex, United Kingdom: John Wiley & Sons Ltd.