HANDHELD SECURE PASSWORD MANAGER

ASHWIN AHUJA

Table of Contents

0 – Introduction to the Problem

Today, the world is becoming more and more connected to the internet, as people become reliant on its services[1]. Around the world, just under 50% of the people have access to the internet at home[2], while in the western world, over 80% of people have this access[3]. Ever since the CEO of IBM stated that the market for personal computers was around 5[4], people have attempted to prove him wrong, with around 350,000 computers sold every year today[5]. And while computer sales in the western world have peaked in 2013[6], the rise of portable computers are only rising[7], with mobiles, tablets and laptops filling the void of desktop computers. With this monumental rise of the internet on every device, the number of passwords the average person has vastly increased, till around 19 today[8]. Every single website (including some of mine!) requires you to create a password, making them intensely hard to remember, especially given the list of recommendations that arise with them. While most sites require passwords of at least eight characters, most recommend including at least one capital letter, one lowercase, one number and one special character. This makes any possible password even harder to remember. And with this comes advice to never reuse passwords, even parts of them, as well as changing them at least every couple of months. The penalty for failing to meet this advice is also quite daunting, as stories reach us about identity theft as people have hacked into many accounts and can forge being the other person[9], to monetary fraud, as hackers manage to extract credit and debit card details from our online shopping accounts[10]. In fact, 2/5 people have in the past year[11]: (Okyle, 2015). To say that the problem is an invisible one is certainly not true, as this is highly broadcast, both over traditional and modern media. Every day we hear of another breach, including a high-profile hack into the Defense Department in USA[12], hacks into Talk-Talk[13] and streaming service Netflix[14] in the last couple of weeks. Over 8/10 people are worries about their online security, yet 60% reuse the same passwords[15]. There is simply not a good method to manage all your passwords, ensuring that they are stored easily and securely. In many ways, this is a huge problem which is preventing the further maturing of the internet, as 7/10 people no longer trust their online security[16]. And yet 21% people use passwords that are over 10 years old[17], while 47% use passwords 5 years old[18]. Methods of better security other than passwords as slowly arriving such as two-factor authentication, which requires you to use your smartphone as well as your password, but are not really replacing them. And while there is a large proportion of the population choosing poor passwords due to lack of better management systems, there is also a part of society, largely the elderly part where they are yet to

[1] (Global Village, n.d.)
[2] (Internet Live Stats, 2015)
[3] (Internet Live Stats, 2015)
[4] (Strohmeyer, n.d.)
[5] (Statistic Brain, 2015)
[6] (Statistic Brain, 2015)
[7] (Statista, n.d.)
[8] (Munson, 2014)
[9] (United States Department of Justice, n.d.)
[10] (Honan, 2012)
[11] (Okyle, 2015)
[12] (Reuters, 2015)
[13] (BBC, 2015)
[14] (McAlone, 2015)
[15] (Okyle, 2015)
[16] (Okyle, 2015)
[17] (Okyle, 2015)
[18] (Okyle, 2015)

be educated of these. The best password management system would be effective for a huge target market, given the scale of demand for a solution to the problem

1 – Target Market and Competition

As of this moment, the number of people using the internet (according to intenetlivestats.com) exceeds 3.2bn people[19], and as I have already explained the problem that I am attempting to solve is a relatively widespread problem with no apparent perfect solution. In many way, the choice of target market that I want to attempt to reach depends on the current solution I wish to overhaul, given the various flaws of each solution, since the needs of various groups appears to be so different. Thus, the first thing one must attempt to investigate are the various solutions and their markets.

1.1 - Methods of password storage

1.1.1 - Paper

From the research that I have completed, the 'low-tech' manner of storing the passwords appears to be used the most by the elderly, who are the most comfortable with paper, especially as they attempt to integrate themselves with the new technology, thus attempt to do one thing at a time. With a brief search of the internet, one can find a number of people complaining about the number of passwords. In fact, people are even suggesting storing passwords on a notebook (), in order to keep it as simple as possible, while the technical advice even suggests reusing 'the same password … for the less important sites'[20], which most security experts reject, given the amount of information that can be incrementally gleamed from multiple accounts. In fact, in a recent attempt to do this, Wired, the technology magazine managed to get from one solitary password (in fact the Apple account password) to all necessary information to commit identity fraud (including Social Security Numbers and Credit Card Information), a truly scary prospect[21]. Additionally, this system appears to shine a light upon one of the most vulnerable groups of people, that of the elderly, who are not necessarily aware of more safe methods of storing passwords, which often require more technical knowledge. As will be shown later, my primary research shows that the vast majority of elderly in my area tend to use this method. Also, how often does one lose a book? Or worse still, if it is stolen (either accidentally or purposefully)? This is the ultimate flaw of this system, that the passwords could all be taken or lost without any warning. Your cleaner, plumber and anyone who enters your house with only a little work could easily take all passwords. Additionally, is the simple irritation of the system. One must carry around the notebook at all times, and changing passwords means the information in the notebook is quickly out of date, and the information inside the book cannot be easily changed without creating a mess. Finally, while finding one password from a handful is generally fine, as the number of passwords inevitably grow, into the hundreds, the simple finding the correct password is a hassle, with no coherent manner of the passwords being sorted or searched for, beyond remembering the order in which you wrote the passwords.

1.1.2 - Word processed files

This is the next stage up, with the people who tend to use this system tending to be of a younger demographic than those writing their passwords on paper, since the system requires a degree of technical abilities. This greater use of technology brings with it a raft of benefits, including the ability to find passwords from anywhere, if a cloud-storage system, such as Google Drive or OneDrive is used. Additionally, finding passwords is a matter of using built in systems in the software to search for the keywords. However, rather inevitably, this system is in many ways even more susceptible to being flawed. While some may encrypt these files, the encryption of many document formats is generally basic and easily hacked. Generally, the storage of

[19] (Internet Live Stats, 2015)
[20] (Potter, 2010)
[21] (Honan, 2012) (LastPass, n.d.)

passwords on a network allows for a vastly greater number of people to attempt to gain access to the passwords, as opposed to the relatively small circle for writing on paper. Systems like Dropbox have famously been hacked in the past[22], meaning that whether one can safely assume that no one has access to the passwords is not true. Additionally, the system is clunky, with passwords poorly arranged, and with the system generally being left to the consumer. Additionally, for those with limited computing abilities, such as the elderly, working with word processers is as challenging, if not more so, than remembering password, thus takes an inordinate amount of time.

1.1.3 – Dedicated Password Management Software

The heralded option by most security experts on the internet is the Dedicated Password Management Software, such as the popular LastPass[23]. They claim to have very strong security, with passwords being unhackable[24]. Additionally, the use of the system would allow one to not be attached to a device or notebook, as one can access the passwords over the internet. Additionally, one only needs to remember one password, which is a truly simple way of doing this. Additionally, given the single purpose of the software, it has a very effective user interface which allows people to easily find the password they want as well as adding new and editing old passwords. On the other hand, this system has a number of flaws. Firstly, the system requires you to remember one password, which if you forget there is no recovery system, as there would be for cloud storage systems. Additionally, this one passwords could be hacked, and this relies upon the security of your password and the secrecy of your username or email address. Although LastPass advertise themselves as having 'very strong protection'[25], they were in fact hacked last year, where a number of user's passwords were stolen[26]. The problem is that security inevitably fixes hacks, such that systems are only as secure as the most secure hack that has occurred. Being on the internet inevitably means that it would be a large target for hackers. Additionally, if this software was to be hacked, one loses the security of all of their passwords, a momentous loss indeed, which could have very severe consequences. Additionally, there is a problem with the software in terms of complexity. Though they advertising being as simple as possible with simple, clean UIs (User Interfaces) when I produced the software in front of some elderly people near me, they seemed and said that they still found it confusing, thereby implying they would be uncomfortable using it. In fact, the number of people that use a password manager according to a survey funded by Roboform (a password management system) was 8%[27], simply showing how most consumers are either unaware of it, or find it irritating to use. Additionally, a large proportion of that 8% is the techie sector, where people are not necessarily average in terms of the number of importance of passwords that they have.

1.2 – Survey

In addition to the successes and failures of the various forms of password management, it is important to consider the relative numbers and demographics of people using various password management systems. Thus, I created a series of questions, which I asked people in my area, parents, friends and anyone else. I also

[22] (Kantke, 2015)
[23] (LastPass, n.d.)
[24] (LastPass, n.d.)
[25] (LastPass, n.d.)
[26] (Forbes Tech, 2015)
[27] (Rubenking, 2015) (SurveyMonkey, n.d.) (Ahuja, n.d.)

created a quiz (using SurveyMonkey[28]) and put it on my Twitter[29] to encourage my followers and anyone else to fill the form.

The questions were as follows:

1) Which age bracket do you fit into?
    a. 0-20
    b. 21-40
    c. 41-60
    d. 61-80
    e. 81+
2) How do you store your password?
    a. Word Processor or similar
    b. Password Management software
    c. On paper

The results were thus:

	0-20	21-40	41-60	61-80	81+
Word processor	13	2	9	1	0
Password management software	3	0	0	0	0
On paper	6	1	3	12	1

Figure 1.2 i Table reporting the results of survey into current password management systems

The results appear to reflect the research that I carried out, showing that as age changes, the majority of people store passwords differently, with the elderly tending towards storing passwords on paper, and the younger people moving towards more technologically savvy methods

1.3 – Conclusions

After assessing the various possible target markets, it is clear that the best primary target market is the elderly. Though the methods of word processing and password management software are not without their various flaws, many of the issues I hope to fix exist most prevalently with the elderly. Many who I talked to want to move to more technologically savvy methods, but having tried them, were unable to understand or use them easily. Thus, one of my primary aims and specifications must be to ensure that the operation of the product is simple, such that anybody, including the elderly can use it easily. Additionally, as discussed in the introduction to the problem, the product could also meet the need of helping the elderly ensure their passwords are strong enough, educating them, as many who I talked to did not truly understand the general guidelines for password selection, for example the vast majority reused the same or very similar passwords for all accounts.

[28] (SurveyMonkey, n.d.)
[29] (Ahuja, n.d.)

2 – Materials Research

In order to decide what materials should be used for my products, a number of properties of the material must be investigated, in order to perform a decision made with the requisite research. Though in essence the product was not too endure huge forces or stresses, it was still important to complete research to ensure that the product would not break under these forces. Additionally, by maximising the efficiency of use of material, finding the exact material to meet my specifications means that I would be able to produce the best product possible.

The definition of material strength is the point at which the material no longer shows a relationship in a Stress-Strain diagram, and thus no longer obeys Hooke's Law. This means that upon releasing the load from the material it will no longer return to the point that it was before the application of the strain, instead there will be a deformation. Another possible definition, though less used, of the material strength is a function which defines the point at which the object will ultimately fracture, thus no longer satisfying the purpose at all. In addition to the specific moduli of reponses to stresses, one must consider whether the material is isotropic. Especially since my product will likely have a relatively constant stress as there would be no specific external stresses applied other than the normal usage stresses, which though may be concentrated upon specific regions, will not have extremely high pressures, isotropic materials would seem to be a perfect fit. However, depending on the geometry of the exact design, orthotropic materials (where the properties are different at right angles) would also be symmetrical

2.1 - Stresses

When a load is placed upon a bar, as in Figure 2a, this induces a number of internal stresses, which are known as the Tension forces. When a plane (mn) is introduced, in Figure 2b, the internal forces (σ) arises from the intermolecular forces that counter the external forces that attempt to distort the body, and return the body to its normal shape. At this point: $\Sigma\sigma = 2F$, as the object is in equilibrium, with no result deformation, presuming it is below the elastic limit of the object. In the same way as this works for tensions in Figure 2a and B, the same logic works for compressions, in Figure 3a and 3b.

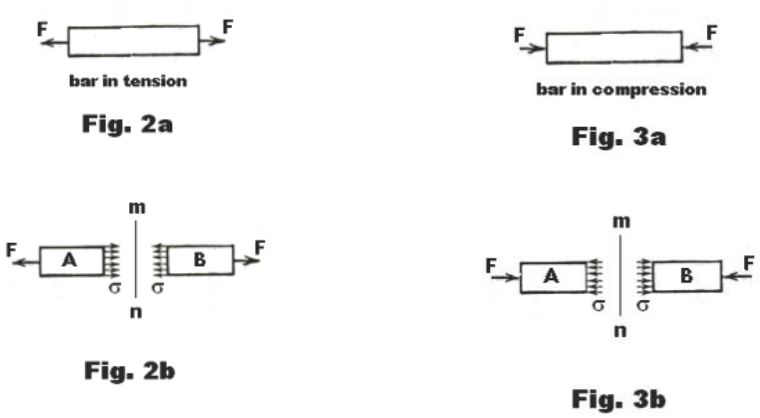

Figure 2.1 i Forces representing the stresses in tension and compression[30]

[30] (Miller, n.d.) (Arif & Stuerzlinger, 2013)

In addition to the Tensile and Compressive stress, there is also shear stress, where equal and opposite forces act on an object from opposite ends. The shear stresses are the forces that resist the tendency for one part of a body to slide over another. Thus, one can generate the following equations:

$$\boldsymbol{Tension}: Stress = E\ (Young's\ Modulus)\ \epsilon$$

$$\boldsymbol{Compression}: Stress = K\ (Bulk\ Modulus)\ \epsilon$$

$$\boldsymbol{Shear}: Shear\ Stress\ (\tau) = G\ (Shear\ Modulus)\ \gamma$$

Much of the consideration of which of the following is the most important very much results from the chosen geometry of the piece. For example, an ergonomic design, designed to be held in both hands, in a tight position will likely result in increases in Tension strains, thus, meaning that the Young's Modulus and Bulk Modulus of the material. On the other hand, if the geometry promotes the user using both hands to grasp the device from top and bottom, the shear rate would be more likely to become important, thus requiring us to look at the Shear Modulus of the product.

On the other hand, though relatively easy to evaluate, it is obvious that the Poisson's Ratio and ductility of the material should not have an impact on the mechanics of the product, as we continue to assume that the scale of the forces would not be large enough to have such an impact upon the product. In fact, based upon some preliminary research, based upon the idea of the size of the product not being disimlar to that of a Mobile Phone, and in fact the impact being very similar, with both having similar usages, the forces are very low. For example, if a touchscreen were to be used, the average force applied by a finger on the screen would be 1.04 – 3.64N[31]. Though the grasp of the hands could contribute to a higher total compression stress, statistics are not easily available, thus experimentation should be carried out to evaluate this. This could be relatively easily completed, using a couple of force-meters and an object, with a size similar to the product being produced.

Once the information has been found, the requirements can be combined with the desired specifications for the product, such as a light weight design, and relatively low costs. Ashby plots are being used to attempt to evaluate a number of materials. Figure 2.1ii allows us to find materials which are both relatively cheap while meeting the need of Young's Modulus, which can be derived using the above equation. Figure 2.1iii on the other hand shows us the Young's Modulus vs the Density of the material which woud be very important, since we want the product to be as light as possible, in order that it would be easy to handle and carry. From both pieces of information, it is clear that an ideal material would likely be derived the polymers section since they seem to combine a high enough Young's Modulus, thereby resistance to tension, and with low enough densities, thus allowing the product to be light and portable. In addition, to be considered is the manufacturing properties of various materials, especially the various pros and cons of specific polymers, given that the material of choice is likely to descend from that particular group.

[31] (Arif & Stuerzlinger, 2013)

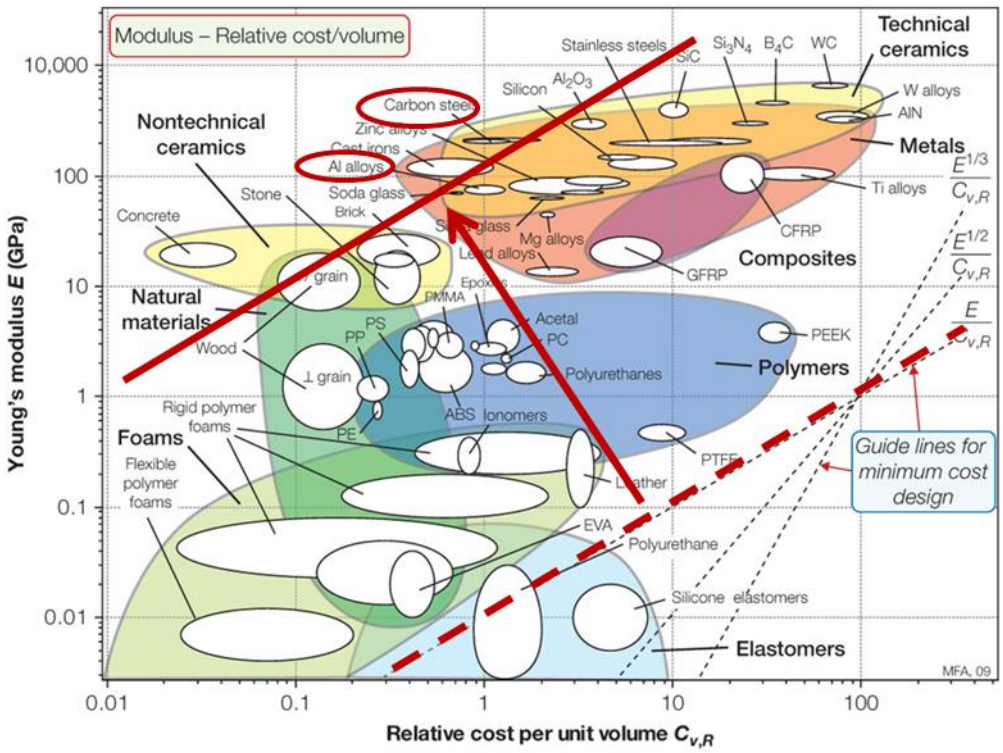

Figure 2.1 ii Ashby Plot of the Young's Modulus vs Cost per unit volume of various components[32]

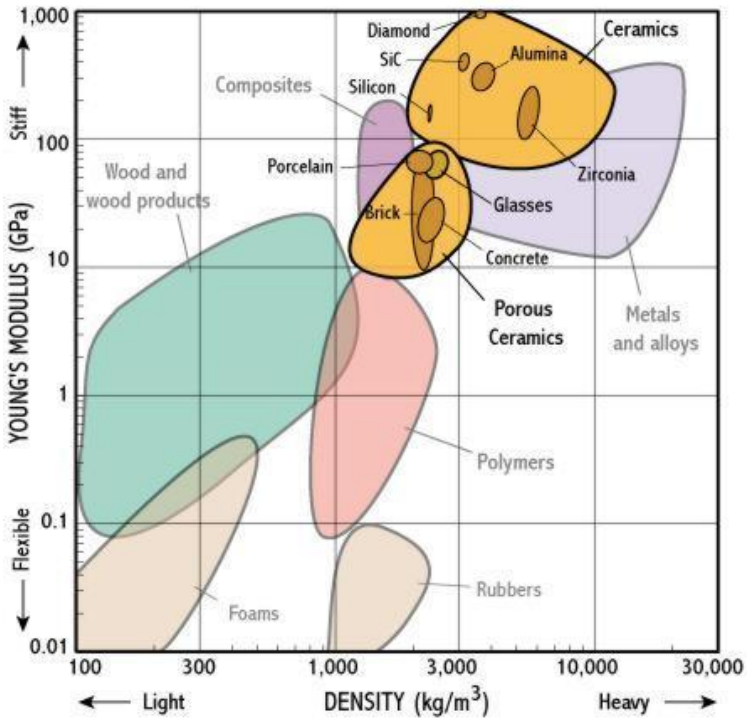

Figure 2.1 iii Ashby Plot of Young's Modulus vs Density of various materials[33]

[32] (Google Sites- Executive Office Chair, n.d.)
[33] (University of Cambridge, n.d.)

2.2 - Hardness

Though not an intrinsic part of the use, another important property of any material to be used is hardness, especially given that the product will be aimed especially towards the elderly. The hardness of the material would be important if the product were to be dropped, which is clearly a possibility as coordination declines at age increases, thus a specification of the product should be that it should not be damaged if dropped. However, if the product were not hard enough, it might deform or break, not meeting this specification. There are a wide variety of manors of testing the hardness of a product, and thereby allowing me to experimentally determine which polymers I should use and which I shouldn't. Thus it would be important to evaluate these different statistics of hardness, how they work and thus decide which measurement of hardness to use. In addition, I can electronically model various geometries and use the inbuilt drop test in SolidWorks Simulation, which will allow me to determine for specific materials and geometries whether they are hard enough to withstand the drop.

2.2.1 – Mohs Test

The Mohs test involves observing whether a material surface is able to me scratched by a substance of known or defined hardness's. It produces ranking for materials along an arbitrary scale from 1-10, and is still often used when completing fieldwork regarding rocks and such-like, though we can now measure quantitatively the exact size and depth of any indentures the attempts at scratching makes.

2.2.2 – Brinell Test

The Brinell Hardness test involves using a desktop machine to put a specific load to a sphere of the chosen material, then we can find the Brinell Hardness Number by the following:

$$Brinell\ Hardness\ Number = \frac{Surface\ area\ of\ indenture}{Load}$$

This has the advantage of ensuring that it takes into account the grain structures of the material, since the load is distributed over a sphere. Additionally, the method is no longer measuring hardness relative to other materials, ensuring a greater accuracy of results, allowing us to individually examine the specific hardness of specific materials.

2.2.3 – Rockwell Test

The Rockwell test makes use of a strong indenter (normally using hard metals or diamond if necessary) which can be precisely controlled to apply a specific load upon a sphere of the material that is being measured. The indenter can also measure the size of the indentations in the materials created, and the size of the indentation is inversely proportional to the Rockwell hardness (though this can be returned in many different units). This is widely used, as the answer is printed immediately, without the requirement of the use of completing any maths. In fact, the major advantage of the using the test for me, is that one can easily find statistics for the hardness of materials using the Rockwell Test online, whereas the information is patchy for the other tests. This would then allow me to create graphs (similar to Ashby Plots) specific to exact polymers, comparing the hardness and various other properties, including density, cost and Moduli as required.

3 – Cipher Research

3.1 - Preliminary Definitions

Code – The set of letters, number, symbols etc, that are used to encrypt messages sent to others.

Cipher – A way of changing a message in order to keep it secret.

Cryptography – The process of writing or reading secret messages or codes.

Encrypt – To change information from one form to another, in order to disguise the true meaning of the message.

3.2 - The Origins of Ciphers[34]

The first ciphers that we know of today originated in Ancient Egypt, created over 4,000 years ago in Menet Khufu, as an adaptation of hieroglyphics, in the tomb of Khnumhotep. Though generally agreed of as a cipher, in many ways, its purpose was in fact much different to what we might term as a cipher. It was as much to impart dignity and authority as to keep something separate. It was intended to impress the reader, as they understood the text, yet it was interesting and novel. Though this inspired the development of such ciphers, simple adaptations of the other Cyrillic scripts, as Professor Owen Lattimore of the University of Leeds states, 'literacy was always restricted to such a small minority that the mere act of putting something into writing that a mere act of putting something into writing was to a certain extent equivalent to putting it into code'[35]. In many ways, this was a constant for the ancient world where messages would be hand delivered, and few could read it, even without a cipher applied. In many ways, the first true attempts to produce secretive codes were in India, where Artha-satra described the ways of the espionage services of India in the years prior to 500BC, where the tasks and reports were provided to superiors using code writing. In fact, Lalitavistara Sutra, which tells of the life of Gautama Buddha describes the now god-like figure's prestigious skill of encryption, while the Karma Sutra, the infamous text describing how one reaches a virtuous life, but today is known for its graphic descriptions of human sexual behaviours, describes 'secret writing' (Miecchita-Vikalpa) as the 45th skill that all women should be proficient in. At this point, the ciphers were largely restricted to substitution ciphers, where consonants and vowels were swapped, or there were random reciprocations.

3.3 - Types of Ciphers

As I intend to make use of at least one cipher to ensure that the passwords stored in the system are protected and even hacking into the code would prevent anyone from accessing the secure data. Thus, one must look at the various ciphers that exist, and the various successes and flaws of the ciphers, including the feasibility of coding and how hackable they are, both by hand and by software. The complexity and thereby generally how secure the ciphers are largely also progress chronologically.

[34] (Kahn, 1996)
[35] (Kahn, 1996)

3.3.1 - Shift Ciphers – Caesar Ciphers

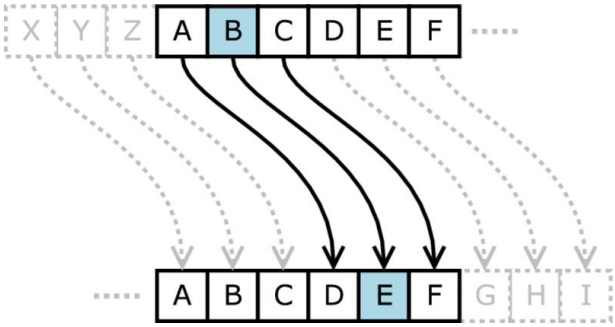

Figure 3.3 i Visual representation of Shift Cipher

One of the earliest form of ciphers (as expressed by Figure 3.3i[36]), as used famously by Julius Caesar, to ensure secrecy in messages was a shift cipher where each letter of the message was moved a certain number of characters up or down the alphabet, such that a shift encryption with a 3-shift of CAT is FDX. This is in essence a very simple cipher, and relied mostly upon the reader not spending the time to break the number of possibilities. This is clearly 26, representing the number of possible shifts that could occur, each one producing a different result. This form of cipher of represented with the figure.

3.3.1.1 - Frequency Analysis

Over time, a more sophisticated form of breaking shift ciphers and in fact a number of other ciphers, was developed by the Arab polymath, Al Kindi, in his paper, 'A Manuscript on Deciphering Cryptographic Messages'[37], by investigating the text of the Qu'ran. The process relies upon the fact that in certain languages the frequency in which certain letters turn up is characteristic, with Figure 3.3ii[38] showing the graph for English. Thus, by investigating the frequency of letters in the encrypted text, one can easily see the code, as the graph would likely be shifted to the side, such that the peaks and troughs would be moved a certain amount. Thus, one can easily find the likely shifts, reducing the number of attempts required to find the exact shift used.

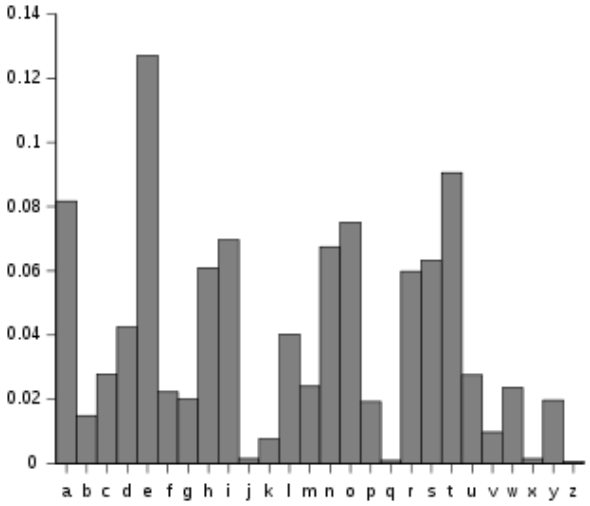

Figure 3.3 ii Graph of frequency of characters in English

3.3.2 - Substitution Ciphers

A substitution cipher is an expansion on the shift cipher, where a predesigned code (such as DFHU…) is used to define where the letters of the message to be encrypted are transposed to, such that in this example: CAB becomes HDF. The number of attempts that would be required to find the specific code in this is much higher, such that it is much harder to break by hand, in fact this is 26! (26x25x24x23x22x21…x3x2x1) = 403291461126605635584000000, since A could be replaced by 26 characters (A through Z), B by 25 (A through Z, except the replacement for A) and so forth. Though in essence the code is easy to implement, one must ensure that both parties and no one else knows the code, generally a challenge, given it should be a random string of 26 characters. Additionally, this is possible to break, using a variation of Frequency Analysis, since one can tend to find the most frequent character is likely to be an E in real life and so forth. Additionally, most systems would also make use of the fact that English and most other languages include a number of repeated

[36] (Exercism.Io, n.d.)
[37] (Alpen-Adria-Universität Klagenfurt, n.d.)
[38] (Wikimedia, n.d.)

words, which can be isolated to find the specific substitutions that are used in that case. Finally, it must be noted that once a few substitutions have been found, the number of possibilities to check reduces very quickly, while for many things to be readable, the entire code is unrequired. For example, it is trivial to work out that 'T?E TI?E IS ?ID?IGH?' is likely to be 'THE TIME IS MIDNIGHT' when the entire substitution code is found.

3.3.3 - Polyalphabetic Ciphers – Vigenere Ciphers

The polyalphabetic cipher was the next step, after the Shift Cipher, where each character was shifted by a different number of position. It included a way in which the code (as a word) itself could be memorised, thus making the process of being the coder and the decipherer much easier. The process works in the following manner, let's say the code is RABBIT, and the message to be coded is HELLOMYNAMEISASHWIN:

RABBIT = 18 1 2 2 9 20

	H	E	L	L	O	M	Y	N	A	M	E	I	S	A	S	H	W	I	N
	R	**A**	**B**	**B**	**I**	**T**	**R**	**A**	**B**	**B**	**I**	**T**	**R**	**A**	**B**	**B**	**U**	**T**	**R**
	8	5	12	12	15	13	25	14	1	13	5	9	19	1	19	8	24	9	14
+	18	1	2	2	9	20	18	1	2	2	9	20	18	1	2	2	9	20	18
	26	6	14	14	24	33	43	15	3	15	14	29	37	2	21	10	33	29	32
=	26	6	14	14	24	7	17	15	3	15	14	3	11	2	21	10	7	3	6
=	**Z**	**F**	**N**	**N**	**X**	**G**	**Q**	**O**	**C**	**O**	**N**	**C**	**K**	**B**	**U**	**J**	**G**	**C**	**F**

Figure 3.3 iii Table showing example encryption using a Polyalphabetic Cipher

Thus, the message, when encrypted becomes: ZFNNXGQOCONCKBUJGCF, which is entirely incoherent, except to someone who is aware that the code is RABBIT, and so can easily reverse the encryption by reversing the change. Though generally simple, the cipher's security relies upon the key remaining private. For example, if Alice were sending a message to Bob, the general example in which the interceptor Eve, is attempting to be bypassed, only remains secure as long as Eve does not know it. In practise, this was used when the two people knew each other well, so were able to securely (in person) choose such a word, without others knowing it. Additionally, the cipher is still intensely vulnerable to the method of Frequency Analysis, by investigating the letters at different intervals, such that the graph of expected frequency distribution will be eventually found, thus allowing us to know the length of the code word. From here, the task is as simple as breaking a number of shift ciphers, a painful, but doable task for a person, while a trivial task for a computer. Despite the apparent and clear flaw in the system which was generally well known, the use of polyalphabetic ciphers continued for a long period, only truly being superseded at the end of the first world war, by more sophisticated systems such as Enigma. In this period, ciphers such the Viginere's cipher (used by the French in WW1), were made more complex by utilising a number of polyalphabetic ciphers, each with different code words, thus drastically increasing the number of calculations that an intercepting enemies would need to do, to break the message.

3.3.4 - Enigma

A final adaptation of the Substitution Cipher was Enigma, the code used during the Second World War by the Nazi Germans. Though there was no repetition, the system used a systematic approach to produce the codes. Firstly, a sender would type the letters of a message into a keyboard. This would create a signal which would pass through the plug board, which systematically switched the signals to those of other letters. Then this signal would pass through a set of 3 (later became 4) rotors which interconnected letters. However, these rotors changed at intervals, with the first rotor spinning every letter, the second after 26, the 3rd after 26x26 and so forth, which changed which character linked to which. Then this now changed character was made then lit up a specific letter on the light board, under the keyboard, to be used (all shown by the diagram in Figure 3.3iv[39]). Though originally one would assume the key space was 263, the rotors could also be rearranged and there were over 60 rotors to choose from, thus leading to a key space of over 60x59x58x57x264 = over 5 x 1012 possible settings, far too many to check by hand. In fact, when one also included the plugboard settings, where the original simple substitution occurred, the number of keys were as follows:

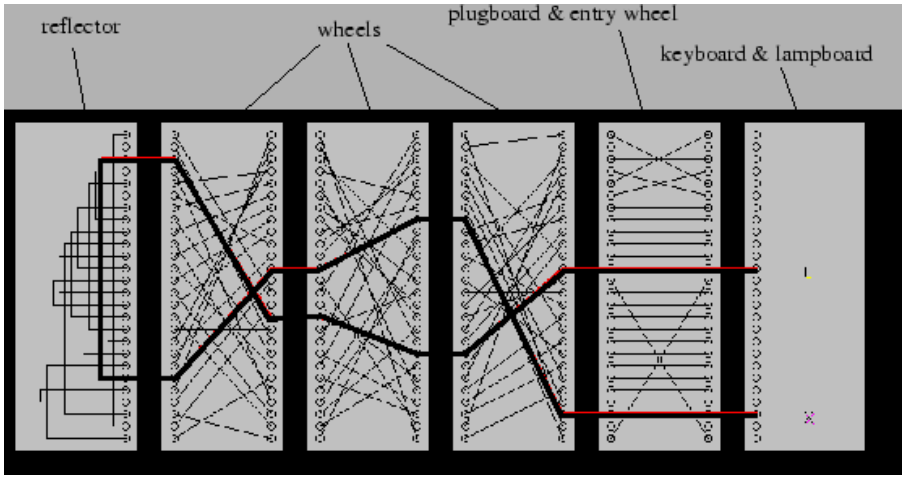

Figure 3.3 iv Diagram of the structure of the Enigma Machine

$$Total\ Number\ of\ Keys\ (with\ 4\ rotors) = 26^4 \times \left(\frac{26!}{14! \times 6! \times 64}\right)$$

$$= 4.5 \times 10^{16}$$

However, the system was ultimately flawed due to one simple reason, that one letter could never encrypt to itself. This was due to the electrical wiring in the system which meant that the letter connector could not connect to itself after the reflector. This combined with the knowledge that certain crypts (keywords) such as 'Heil Hitler' would always be in the messages, thus vastly reducing the number of settings that one needed to test.

3.3.5 - One-Time Pad

According to many, Claude Shannon is the father of modern cryptography, from here, many of the ciphers discussed from here on are still used. Shannon's work addressed the 'problems of cryptography'[40]. He largely separated the types of cryptography into two, one where the cipher was designed to protect against hackers who have infinite resources, now known as unconditional secrecy and a second, where the cipher protects against hackers with finite amount of resources. He also began to define the idea of 'Perfect Secrecy' the cipher text conveys 'no information about the content of the plaintext'[41]. The manner in which this can occur

[39] (Anon., n.d.)
[40] (University of California San Diego, 2008)
[41] (University of California San Diego, 2008)

is using the One-Time Pad, where frequency analysis and other cryptanalytic techniques would have no effect upon the encrypted text. The One-Time Pad is similar to the Polyalphabetic and Substitution Cipher except that the code (key) is entirely random and as long as the text to encoded, such that there is no repetition of the entire code, which left the polyalphabetic code vulnerable. Shannon definitely proved that perfect secrecy was only possible if this was true. Notice, the code is required to be perfectly random, as opposed to pseudo-random, two entirely different concepts. If one were to approach a person and ask them for four random numbers, they are more likely to choose certain numbers, in fact to a certain extent a large flaw and the reason for the failure of the Enigma System. If the code was entirely random, the code is very hard to break, as there are few ways of breaking it, lest a form of trial and error, where the number of possibilities are 26n where n is the length of the code to be encrypted.

An example of the use of a One-Time Pad

Text to be encrypted: EVEYOUCANTHEARMENOW

Code to be used: 23 19 14 4 26 16 8 8 10 13 10 26 14 20 2 13 15 4 15

	E	V	E	Y	O	U	C	A	N	T	H	E	A	R	M	E	N	O	W
	23	19	14	4	26	16	8	8	10	13	10	26	14	20	2	13	15	4	15
+	5	22	5	25	15	21	3	1	14	20	8	5	1	18	13	5	14	15	23
=	28	41	19	29	41	37	11	9	24	33	18	31	15	38	15	18	29	19	38
=	2	15	19	3	15	11	11	9	24	7	18	5	15	12	15	18	3	19	12
=	B	O	S	C	O	K	K	I	X	G	R	E	O	L	O	R	C	S	23

Figure 3.3 v Table showing an example use of the One Time Pad for encryption

Though the One-Time Pad is by far the most sophisticated type of cipher that we have encountered so far, there are a few issues that it introduces. Firstly, the length of the code must be at least as long if not longer than the message that the people wish to encrypt, thus taking up a lot of length. This makes it very difficult to remember and use effectively. Additionally, the question becomes how to ensure that both parties have the same code, to decrypt and encrypt the message, since much of the communications are not in person and rely upon inherently insecure systems such as the internet. The mechanisms of fixing these issues were fundamentally fixed in the DES and Diffie-Helman Key Exchange System.

3.3.6 - Diffie-Helman Key Exchange – Assymetric Key Exchange

The next big breakthrough came in 1976, produced by Whitfield Diffie and Martin Helman, where they designed a manner in which keys could easily be exchanged, solving one of the theoretical problems in Shannon's description of the One-Time Pad. For the first time, the two parties (Alice and Bob) never needed to come into contact for the message to be secure. It established a method of key exchange called Assymetric Key Exchange, where both parties have both a 'Private Key' and a 'Public Key'. Ensuring that Eve does not have any access to the key uses a situation known as the Discrete Logarithm Problem.

3.3.6.1 - How it works:

First, we must establish the concept of Modular Arithmetic, where we alter the base of a number. For example, 3 = 1 (mod 2). This is the same as calculating the division of 3/2 and then finding the remainder of this. This is an integral part of the method, since the modular function is very easy to calculate, but very hard to reverse.

So, to start off, both Alice and Bob decide (publically) on two prime numbers, where p is a prime, and q is a generator of p. A generator is a number that when raised to whole number (integer) powers less than p never produces the same result (every modulus is equally likely). These numbers can be distributed over the internet, thus, Eve is now aware of these numbers. From here, the pair each create their own personal key (a and b) and find another number using the following formula:

$$a' = q^a \ mod(p)$$

$$b' = q^b \ mod(p)$$

From here, they transfer a' and b' over the insecure network to each other (and thereby Eve), thus allowing them to communicate using the following formula:

$$Key\ (Bob) = a'^b \ mod(p)$$

$$Key\ (Alice) = \ b'^a \ mod(p)$$

As it turns out, these two are identical, since, by substituting how these keys were produced:

$$Key\ (Bob) = q^{a^b} mod(p) = q^{ab} \ mod(p)$$

$$Key\ (Alice) = q^{b^a} mod(p) = q^{ab} \ mod(p)$$

However, despite Alice knowing a', b', p and q, finding a and b, as would be required to find the key is a challenge which requires a large amount of computing power, with computers normally taking at least a decade to solve. This is due to a problem known as the Discrete Logarithm Problem. From here, the key can be used as the encryption to encrypt and decrypt messages as required without Eve being able to decrypt them as they are sent using the insecure internet connection.

3.3.6.2 - Discrete Logarithm Problem

Let g be a generator of two integers; x and p (where p is a prime);

$$Answer = \ g^x \ mod(p)$$

For example:

$$3^{29} \ mod(17) = 12$$

However, knowing this, it is hard to find 29 given all the rest of the information, in fact relying upon trial and error to solve the reverse function:

$$Answer \ mod(17) = \ log_3(12)$$

3.3.7 - RSA

From there, the next largest development was the RSA system, so names as it was created by Ron Rivest, Adi Shamir and Leonard Adleman in 1977. Though the algorithm could (and I will show how it could) be used for encryption, the system is a relatively slow form of encryption and so is generally only used to verify identities of users and suchlike, for example for passwords sent over the internet. The system relies upon two more mathematical functions:

3.3.7.1 - Prime Factorisation:

In his 'Elements'[42], Euclid stated the following: 'if q (any integer) is not prime, then some prime factor p divides q'. This allows us to create a system through which every number can be written as a multiplication of prime factors; eg

$$24 = 2 \times 2 \times 2 \times 3$$

Though this is simple to do for low numbers, this is a fundamentally hard problem. Though we have all factorised low numbers in our childhood, this becomes significantly harder as the number increase in length. In fact, in order to factorise a 232-digit number, researchers at the Swiss Federal Institute of Technology took over two years[43], using multiple computer machines and the most efficient algorithms that we know. Ultimately, according to Dirichlet's Theorem, this is due to the random distribution of primes, which prevents prime factorization occurring over a simple iterative cycle time, (i.e. polynomial p^n time). Though there are some mathematicians today who suggest that P=NP (all problems that would otherwise run in non-polynomial time could be run in polynomial time using quantum computing) it is yet to be solved, despite carrying with it a million-dollar prize, as one of the Millennium Prize Problems. As a representation of this, Figure 3.3vi[44] compares the time taken for multiplication vs the time taken for prime factorisation as the numbers increase. RSA relies upon this, to ensure that the function is very hard to reverse.

Figure 3.3 vi Graph showing time taken to complete multiplication and factorisation as the size of numbers increase

[42] (Clarke University, n.d.)
[43] (Scientific American, 2010)
[44] (Khan Academy, n.d.)

3.3.7.2 - Phi (Φ) Function (Euler's Totient Function):

This is a function which counts the number of positive integers below the number which do not share any common factors with it, thereby are relatively prime to n.

$$\phi(n) \text{ is the number of integers } i \text{ in the range } 1 \leq i \leq n \text{ where } GCD(i,k) = 1$$

$$eg. \Phi(8): 1,\cancel{2},3,\cancel{4},5,\cancel{6},7,\cancel{8} => \Phi(8) = 4$$

An important fact about this is that the Phi Function is multiplicative, therefore if Phi(n) = a, and Phi(m) = b, Phi(mn) = Phi(m) x Phi(n) = a x b = ab, presuming that A and B are coprime (thereby A and B do not share any factors bar 1). This can be proved thus (also deriving the Chinese Remainder Theorem):

$To\ prove\ this, we\ must\ establish\ a\ one\ to\ one\ correspondence\ (each\ element\ maps\ directly\ to\ one\ of\ the\ other\ between\ \Phi(mn)\ and\ \Phi(\mathrm{m})(\mathrm{n})$

$Thus\ we\ establish\ two\ sets:$

$$S1 = \{a: 1 \leq a \leq mn\ and\ \gcd(a, mn) = 1\} - this\ clearly\ has\ \phi(mn)\ number\ of\ elements$$

$$S2 = \{(b,c): 1 \leq b \leq m\ and\ \gcd(b,m) = 1\}$$

$$and\ 1 \leq c \leq n\ and\ \gcd(c,n) = 1\}$$

$We\ can\ show\ that\ this\ has\ length\ of\ \Phi(m)\Phi(n)\ by\ thinking\ how\ we\ could\ vary\ each\ of\ the\ two\ coordinates, the\ first\ of\ the\ ordered\ pair\ by\ \Phi(m)\ and\ the\ second\ by\ \Phi(n), thus\ the\ number\ of\ companations\ possible = \Phi(m)\Phi(n)$

$By\ considering\ any\ value\ for\ m\ an\ n, we\ can\ show\ there\ is\ an\ association\ between\ them, for\ example\ m = 4\ and\ n = 5$

$\mathbf{1} \rightarrow (\mathbf{1}, \mathbf{1})$

$\mathbf{2} \rightarrow (\mathbf{3}, \mathbf{3})$

$\mathbf{7} \rightarrow (\mathbf{3}, \mathbf{2})$

$\mathbf{9} \rightarrow (\mathbf{1}, \mathbf{4})$

$\mathbf{11} \rightarrow (\mathbf{3}, \mathbf{1})$

$\mathbf{13} \rightarrow (\mathbf{1}, \mathbf{3})$

$\mathbf{17} \rightarrow (\mathbf{1}, \mathbf{2})$

$\mathbf{19} \rightarrow (\mathbf{3}, \mathbf{4})$

In order to show that different numbers in S1 get sent to different numbers in S2:

If this is true –

Let a1 and a2 be distinct elements of S1 and are mapped to the same pair in S2

$$a1 \equiv a2 \ mod(m) \ and \ a1 \equiv a2 \ mod(n)$$

This implies: $mn|(a1 - a2)$ *since m and n are relatively prime*

$\rightarrow$ $a1 \equiv a2 \ mod \ (mn)$ *which contradicts the original assumption that a1 and a2 were different*

In order to show that for ever pair in S2, there is an association with an alement in S1:

For any example: $a \equiv b \ mod(m) \ AND \ a \equiv c \ mod(n)$

According to the Euclidean Algorithm, this is only always true when the $GCD(m, n) = 1 \rightarrow$ *they are coprime.*

Another important fact to note, which is relatively trivial is that Phi(p) where p is prime = p – 1, since by its very definition, p does not share any common factors with any number less than it except one.

3.3.7.3 - How it works:

First, Alice (the person receiving data at this point) creates a private key, which is not distributed over the internet, instead remaining private such that only she knows what it is. She makes use of the phi function; using the fact that Phi(pq) where p and q are primes = (p-1)(q-1). However, finding Phi(n) where n=pq is very hard (especially when the numbers are high) since one does not know the prime factorisation of n, and doing this would be a fundamentally hard problem. Then we also make use of Euler's Theorem, which states:

$$m^{\phi(n)} \equiv 1 \ mod \ (n)$$

$$Through\ some\ rearrangement\ to\ allow\ us\ to\ make\ it\ work\ in\ this\ case:$$

$$m^{\phi(n)+1} = m\ mod\ (n)$$

Alice chooses two large primes (p and q) and calculates n = pq. Then she chooses a small number e (normally 65537 unless n is a multiple of this), where gcd(3,n) = 1. Then she sends n and e to Bob, while also calculating for herself, as the private key = (Phi(n)+1)/e = ((p-1)(q-1)+1)/e. In order to calculate the message to send back, Bob does the following, where m is his message:

Ciphertext (c) = m^e mod(n)

Then to decrypt the message, Alice does the following:

$C^{private\ key}$ mod(n) = message text = $m^{e((phi(n)+1)/e}$ mod(n) = $m^{phi(m)+1}$ mod(n) = m mod(n) [as shown above]

Thus, Alice is able to ge the message Bob has sent. Meanwhile, Eve, who has n, c and e cannot get m, unless she knows phi(n).

3.3.8 - Rijndael Cipher (Advanced Encryption Standard - AES)

3.3.8.1 - Introduction

The Rijndael Cipher was chosen (in 2002) as the new encryption standard by the National Institute of Science and Technology (NIST) of America to replace the DES (Data Encryption Standard) as a cipher which could be widely used for both personal and commercial used ensuring it is entirely unhackable. The cipher attempts to use a key (16 bytes long) and can encrypt a 16-byte message using an iterative process which is still relatively easy to reverse, given the key is known. Essentially, it makes it harder for a hacker to find the message given the ciphertext, when they do not know the key. The AES is a block cipher, meaning that the number of bytes that it encrypts is fixed. For the purposes of this, I will look at the 16-byte block length, though theoretically the block length could be 32 or 64 bytes, though the encryption process varies between the block length and so is not necessary to mention.

3.3.8.2 - Prerequisites

XOR

XOR is a function which operates on the individual bits in a byte in the same way a 2 pin XOR-Logic Gate would, thereby having the following table[45] of inputs -> outputs:

```
0 XOR 0 = 0
1 XOR 0 = 1
1 XOR 1 = 0
0 XOR 1 = 1
```

Figure 3.3 vii XOR Table

[45] (Berent, n.d.)

HEX

HEX is an alternate definition of numbers in base 16, thus a single digit could have a value of upto 15 (represented by F). This allows us to refer to a single byte as two hex characters as opposed to 8 bits, thus saving lots of time and paper.

First Hex Digit → [e.g. $64_{16}=100_{10}$] $HEX_{16} \leftrightarrow DECIMAL_{10}$

© 2001 Roderic A Davis, 2nd	0x	1x	2x	3x	4x	5x	6x	7x	8x	9x	Ax	Bx	Cx	Dx	Ex	Fx
Second Hex Digit ↓ x0	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
x1	1	17	33	49	65	81	97	113	129	145	161	177	193	209	225	241
x2	2	18	34	50	66	82	98	114	130	146	162	178	194	210	226	242
x3	3	19	35	51	67	83	99	115	131	147	163	179	195	211	227	243
x4	4	20	36	52	68	84	100	116	132	148	164	180	196	212	228	244
x5	5	21	37	53	69	85	101	117	133	149	165	181	197	213	229	245
x6	6	22	38	54	70	86	102	118	134	150	166	182	198	214	230	246
x7	7	23	39	55	71	87	103	119	135	151	167	183	199	215	231	247
x8	8	24	40	56	72	88	104	120	136	152	168	184	200	216	232	248
x9	9	25	41	57	73	89	105	121	137	153	169	185	201	217	233	249
xA	10	26	42	58	74	90	106	122	138	154	170	186	202	218	234	250
xB	11	27	43	59	75	91	107	123	139	155	171	187	203	219	235	251
xC	12	28	44	60	76	92	108	124	140	156	172	188	204	220	236	252
xD	13	29	45	61	77	93	109	125	141	157	173	189	205	221	237	253
xE	14	30	46	62	78	94	110	126	142	158	174	190	206	222	238	254
xF	15	31	47	63	79	95	111	127	143	159	175	191	207	223	239	255

Figure 3.3 viii[46] Conversion table between HEX and decimal

3.3.8.3 - General Steps

The first section of the process is to expand the key; this entire process moves from a 16-byte key to a 176-byte key which can be used in the main encryption system. This is made by completing the following operations:

Rot Word

Sub Word

Rcon

EK

K

These are completed in the following order (with each one acting like a function, taking a certain input and giving a certain output). The inner functions are therefore completed first when there are multiple in one round, followed by the outer one progressively.

[46] (Anon., 2012)

Round	Expanded Key Bytes	Function
0	0 1 2 3	**K**(0)
1	4 5 6 7	**K**(4)
2	8 9 10 11	**K**(8)
3	12 13 14 15	**K**(12)
4	16 17 18 19	**Sub Word**(**Rot Word**(**EK**((4-1)*4))) XOR **Rcon**((4/4)-1) XOR **EK**((4-4)*4)
5	20 21 22 23	**EK**((5-1)*4)XOR **EK**((5-4)*4)
6	24 25 26 27	**EK**((6-1)*4)XOR **EK**((6-4)*4)
7	28 29 30 31	**EK**((7-1)*4)XOR **EK**((7-4)*4)
8	32 33 34 35	**Sub Word**(**Rot Word**(**EK**((8-4)*4))) XOR **Rcon**((8/4)-1) XOR **EK**((8-4)*4)
9	36 37 38 39	**EK**((8-1)*4)XOR **EK**((9-4)*4)
10	40 41 42 43	**EK**((10-1)*4)XOR **EK**((10-4)*4)
11	44 45 46 47	**EK**((11-1)*4)XOR **EK**((11-4)*4)
12	48 49 50 51	**Sub Word**(**Rot Word**(**EK**((12-4)*4))) XOR **Rcon**((12/4)-1) XOR **EK**((12-4)*4)
13	52 53 54 55	**EK**((13-1)*4)XOR **EK**((13-4)*4)
14	56 57 58 59	**EK**((14-1)*4)XOR **EK**((14-4)*4)
15	60 61 62 63	**EK**((15-1)*4)XOR **EK**((15-4)*4)
16	64 65 66 67	**Sub Word**(**Rot Word**(**EK**((16-4)*4))) XOR **Rcon**((16/4)-1) XOR **EK**((16-4)*4)
17	68 69 70 71	**EK**((17-1)*4)XOR **EK**((17-4)*4)
18	72 73 74 75	**EK**((18-1)*4)XOR **EK**((18-4)*4)
19	76 77 78 79	**EK**((19-1)*4)XOR **EK**((19-4)*4)
20	80 81 82 83	**Sub Word**(**Rot Word**(**EK**((20-4)*4))) XOR **Rcon**((20/4)-1) XOR **EK**((20-4)*4)
21	84 85 86 87	**EK**((21-1)*4)XOR **EK**((21-4)*4)
22	88 89 90 91	**EK**((22-1)*4)XOR **EK**((22-4)*4)
23	92 93 94 95	**EK**((23-1)*4)XOR **EK**((23-4)*4)
24	96 97 98 99	**Sub Word**(**Rot Word**(**EK**((24-4)*4))) XOR **Rcon**((24/4)-1) XOR **EK**((24-4)*4)
25	100 101 102 103	**EK**((25-1)*4)XOR **EK**((25-4)*4)
26	104 105 106 107	**EK**((26-1)*4)XOR **EK**((26-4)*4)
27	108 109 110 111	**EK**((27-1)*4)XOR **EK**((27-4)*4)
28	112 113 114 115	**Sub Word**(**Rot Word**(**EK**((28-4)*4))) XOR **Rcon**((28/4)-1) XOR **EK**((28-4)*4)
29	116 117 118 119	**EK**((29-1)*4)XOR **EK**((29-4)*4)
30	120 121 122 123	**EK**((30-1)*4)XOR **EK**((30-4)*4)
31	124 125 126 127	**EK**((31-1)*4)XOR **EK**((31-4)*4)
32	128 129 130 131	**Sub Word**(**Rot Word**(**EK**((32-4)*4))) XOR **Rcon**((32/4)-1) XOR **EK**((32-4)*4)
33	132 133 134 135	**EK**((33-1)*4)XOR **EK**((33-4)*4)
34	136 137 138 139	**EK**((34-1)*4)XOR **EK**((34-4)*4)
35	140 141 142 143	**EK**((35-1)*4)XOR **EK**((35-4)*4)
36	144 145 146 147	**Sub Word**(**Rot Word**(**EK**((36-4)*4))) XOR **Rcon**((36/4)-1) XOR **EK**((36-4)*4)
37	148 149 150 151	**EK**((37-1)*4)XOR **EK**((37-4)*4)
38	152 153 154 155	**EK**((38-1)*4)XOR **EK**((38-4)*4)
39	156 157 158 159	**EK**((39-1)*4)XOR **EK**((39-4)*4)
40	160 161 162 163	**Sub Word**(**Rot Word**(EK((40-4)*4))) XOR **Rcon**((40/4)-1) XOR **EK**((40-4)*4)
41	164 165 166 167	**EK**((41-1)*4)XOR **EK**((41-4)*4)
42	168 169 170 171	**EK**((42-1)*4)XOR **EK**((42-4)*4)
43	172 173 174 175	**EK**((43-1)*4)XOR **EK**((43-4)*4)

Figure 3.3 ix[47] Key Expansion routine for 16 Byte AES Encryption

The other section is the main encryption, which also is composed of a number of functions, acting in a specific order. The list and order of use of functions are as follows:

Add Round Key

[47] (Berent, n.d.)

Byte Sub

Shift Row

Mix Column

Round	Function
–	Add Round Key(State)
0	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
1	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
2	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
3	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
4	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
5	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
6	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
7	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
8	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
9	Add Round Key(Byte Sub(Shift Row(State)))

Figure 3.3 x[48] Encryption Routine for AES 16 Byte Encryption

At the end of this, the encryption produces a 16 byte encrypted message, which is theoretically unhackable, unless you have the key, in which case it is very easy to reverse.

3.3.8.4 - Definitions of each step

3.3.8.4.1 - Key Expansion

Rot Word

This function takes an input of four bytes, circularly shifting them around, for example, moving:

$(a1\ a2\ a3\ a4)\ to\ (a2\ a3\ a4\ a1)$

The offset (1, 2, 3 or 4) depends on the round, using:

$offset = round\ mod(4)$

Sub Word

This function inputs four bytes, and converts them using the below table, dealing with each byte individually, to produce a different byte for each of them and merging it again to form a block of four bytes again.

[48] (Berent, n.d.)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Figure 3.3 xi[49] Substitution table for AES 16 Byte encryption

Correspondingly, when the step is required during decryption, this table is used:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
2	54	7B	94	32	A6	C2	23	3D	EE	4C	95	0B	42	FA	C3	4E
3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

Figure 3.3 xii[50] Substitution table for AES 16 Byte Decryption

Rcon

This takes an input of which round it is and the keysize (always 16 in the case of a 16-byte key), finds: (Round/(KeySize/4)-1 and returns the following value according to the table:

[49] (Berent, n.d.)
[50] (Berent, n.d.)

```
Rcon(0)     = 01000000
Rcon(1)     = 02000000
Rcon(2)     = 04000000
Rcon(3)     = 08000000
Rcon(4)     = 10000000
Rcon(5)     = 20000000
Rcon(6)     = 40000000
Rcon(7)     = 80000000
Rcon(8)     = 1B000000
Rcon(9)     = 36000000
Rcon(10)    = 6C000000
Rcon(11)    = D8000000
Rcon(12)    = AB000000
Rcon(13)    = 4D000000
Rcon(14)    = 9A000000
```

Figure 3.3 xiii[51] RCon conversion table for AES 165 Byte Encryption

EK

The EK function will return 4 bytes of the Expanded Key after the specified offset (which is the input). For example, if the offset is 4, the EK function will return bytes 4, 5, 6 and 7 of the Expanded Key.

K

The K function will return 4 bytes of the original key after the specified offset (in the same way as EK but for the original key).

3.3.8.4.2 - Encryption

Add Round Key

The function XORs each byte of the state (the ciphertext as it stands) with 16 bytes of the key, such that every bit of the state is compared against a bit of the key. Additionally, for the 11 times it is run, the part of the key (given it is now 176 bytes long) that the state is compared against is changed, such that the first time, it is compared against bytes 0-15, the second 16-31 etc.

Byte Sub

Byte Sub replaces the values of each byte using the values in the table below, such that the value of every byte is changed. For example, a byte with value HEX 19 would be replaced with a byte of value D4.

[51] (Berent, n.d.)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Figure 3.3 xiv[52] SBOX Substitution Table for 16 Byte AES Encryption

To reverse this process, the value is replaced with the corresponding inverses of the SBOX (the specific name for the table) thus:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
2	54	7B	94	32	A6	C2	23	3D	EE	4C	95	0B	42	FA	C3	4E
3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

Figure 3.3 xv[53] SBOX Substitution Table for 16 Byte AES Decryption

Shift Row

First, the state is converted into a 4x4 matrix, with one byte per element of the matrix, thus, with the elements formed vertically:

[52] (Berent, n.d.)
[53] (Berent, n.d.)

$$\begin{matrix} 0 & 4 & 8 & 12 \\ 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \\ 3 & 7 & 11 & 15 \end{matrix}$$

Then the rows are circularly shifted by a certain number of elements, according to which row it. The top row is shifted 0 elements along, the second row shifted 1 element, third 2 elements and bottom one 3 elements along, thus forming this:

$$\begin{matrix} 0 & 4 & 8 & 12 \\ 5 & 9 & 13 & 1 \\ 10 & 14 & 2 & 6 \\ 15 & 3 & 7 & 11 \end{matrix}$$

During decryption, the same process is reversed and all the rows are shifted by the same amounts to the left.

Mix Column

This step involves multiplication of the state and the multiplication matrix thusly:

Multiplication Matrix

$$\begin{matrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{matrix}$$

State

$$\begin{matrix} b1 & b5 & b9 & b13 \\ b2 & b6 & b10 & b14 \\ b3 & b7 & b11 & b15 \\ b4 & b8 & b12 & b16 \end{matrix}$$

The process is completed in the usual manner for matrix cross-multiplication, except that instead of addition of terms, the function XOR is completed instead. For example:

b'1 = (b1 x 2) XOR (b2 x 3) XOR (b3 x 1) XOR (b4 x 1)

Unfortunately, however, another complexity is added by the fact that the multiplication is completed using a Galois Field.

Galois Field Multiplication in Practise

In practise, this involves the manner in a x b (where both a and b are bytes) is completed. Firstly, one must look up the values for a' and b' in the L table – the first HEX digit is the vertical index and second digit is the horizontal index. If the byte has a value below 15, this should be interpreted as 0F. Then these values are arithmetically added together and then the result is looked for in the E Table.

L Table

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		00	19	01	32	02	1A	C6	4B	C7	1B	68	33	EE	DF	03
1	64	04	E0	0E	34	8D	81	EF	4C	71	08	C8	F8	69	1C	C1
2	7D	C2	1D	B5	F9	B9	27	6A	4D	E4	A6	72	9A	C9	09	78
3	65	2F	8A	05	21	0F	E1	24	12	F0	82	45	35	93	DA	8E
4	96	8F	DB	BD	36	D0	CE	94	13	5C	D2	F1	40	46	83	38
5	66	DD	FD	30	BF	06	8B	62	B3	25	E2	98	22	88	91	10
6	7E	6E	48	C3	A3	B6	1E	42	3A	6B	28	54	FA	85	3D	BA
7	2B	79	0A	15	9B	9F	5E	CA	4E	D4	AC	E5	F3	73	A7	57
8	AF	58	A8	50	F4	EA	D6	74	4F	AE	E9	D5	E7	E6	AD	E8
9	2C	D7	75	7A	EB	16	0B	F5	59	CB	5F	B0	9C	A9	51	A0
A	7F	0C	F6	6F	17	C4	49	EC	D8	43	1F	2D	A4	76	7B	B7
B	CC	BB	3E	5A	FB	60	B1	86	3B	52	A1	6C	AA	55	29	9D
C	97	B2	87	90	61	BE	DC	FC	BC	95	CF	CD	37	3F	5B	D1
D	53	39	84	3C	41	A2	6D	47	14	2A	9E	5D	56	F2	D3	AB
E	44	11	92	D9	23	20	2E	89	B4	7C	B8	26	77	99	E3	A5
F	67	4A	ED	DE	C5	31	FE	18	0D	63	8C	80	C0	F7	70	07

Figure 3.3 xvi[54] L Table for AES 16 Byte Encryption

E Table

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	01	03	05	0F	11	33	55	FF	1A	2E	72	96	A1	F8	13	35
1	5F	E1	38	48	D8	73	95	A4	F7	02	06	0A	1E	22	66	AA
2	E5	34	5C	E4	37	59	EB	26	6A	BE	D9	70	90	AB	E6	31
3	53	F5	04	0C	14	3C	44	CC	4F	D1	68	B8	D3	6E	B2	CD
4	4C	D4	67	A9	E0	3B	4D	D7	62	A6	F1	08	18	28	78	88
5	83	9E	B9	D0	6B	BD	DC	7F	81	98	B3	CE	49	DB	76	9A
6	B5	C4	57	F9	10	30	50	F0	0B	1D	27	69	BB	D6	61	A3
7	FE	19	2B	7D	87	92	AD	EC	2F	71	93	AE	E9	20	60	A0
8	FB	16	3A	4E	D2	6D	B7	C2	5D	E7	32	56	FA	15	3F	41
9	C3	5E	E2	3D	47	C9	40	C0	5B	ED	2C	74	9C	BF	DA	75
A	9F	BA	D5	64	AC	EF	2A	7E	82	9D	BC	DF	7A	8E	89	80
B	9B	B6	C1	58	E8	23	65	AF	EA	25	6F	B1	C8	43	C5	54
C	FC	1F	21	63	A5	F4	07	09	1B	2D	77	99	B0	CB	46	CA
D	45	CF	4A	DE	79	8B	86	91	A8	E3	3E	42	C6	51	F3	0E
E	12	36	5A	EE	29	7B	8D	8C	8F	8A	85	94	A7	F2	0D	17
F	39	4B	DD	7C	84	97	A2	FD	1C	24	6C	B4	C7	52	F6	01

Figure 3.3 xvii[55] E Table for 16 Byte AES Encryption

[54] (Berent, n.d.)
[55] (Berent, n.d.)

Galois Field Multiplication in Theory

A field is a set of elements which satisfies the field axioms, thus can be added (subtracted) and multiplied (divided) by other members in the field in order to form a member of the field. Thus, a group such as the integers are examples of sets which aren't a field, since integer division may not produce another integer. Examples of fields include the Complex Numbers, Rational Numbers and Real Numbers. A Galois Field is simply a field with a finite set of members. In order to multiply using a finite field, it is equivalent to multiplication followed by division using a reducing polynomial as a divisor, where the remainder is the product. For the Rijndael Cipher, the irreducible reducing polynomial is: $x^8+x^4+x^3+x+1$

References

Ahuja, A., n.d. *Ashwin Ahuja.* [Online]
Available at: http://twitter.com/Ashwin_Ahuja
[Accessed 4 12 2015].

Alpen-Adria-Universität Klagenfurt, n.d. *The Breakthrough of Frequency Analysis.* [Online]
Available at: http://cs-exhibitions.uni-klu.ac.at/index.php?id=279
[Accessed 4 12 2015].

Anon., 2012. [Online]
Available at: http://freepages.genealogy.rootsweb.ancestry.com/~dav4is/Misc/images/X2D2X.gif
[Accessed 4 12 2015].

Anon., n.d. *Operation of the Enigma machine with an example.* [Online]
Available at: http://www.mlb.co.jp/linux/science/genigma/enigma-referat/node4.html
[Accessed 4 12 2015].

Arif, A. S. & Stuerzlinger, W., 2013. *Pseudo-Pressure Detection and Its Use in Predictive Text Entry on Touchscreens.* [Online]
Available at: http://ws.iat.sfu.ca/papers/pseudopressure.pdf
[Accessed 04 12 2015].

BBC, 2015. *TalkTalk cyber-attack: Website hit by 'significant' breach.* [Online]
Available at: http://www.bbc.co.uk/news/uk-34611857
[Accessed 04 12 2015].

Berent, A., n.d. *AES Simplified.* [Online]
Available at: https://www.ime.usp.br/~rt/cranalysis/AESSimplified.pdf
[Accessed 1 12 2015].

Clarke University, n.d. *Euclid's Elements.* [Online]
Available at: http://aleph0.clarku.edu/~djoyce/java/elements/toc.html
[Accessed 4 12 2015].

Exercism.Io, n.d. *Simple Cipher in Swift.* [Online]
Available at: http://exercism.io/exercises/swift/simple-cipher/readme
[Accessed 4 12 2015].

Forbes Tech, 2015. *Password Manager LastPass Hacked, Exposing Encrypted Master Passwords.* [Online]
Available at: http://www.forbes.com/sites/katevinton/2015/06/15/password-manager-lastpass-hacked-exposing-encrypted-master-passwords/
[Accessed 04 12 2015].

Global Village, n.d. *Internet World Stats.* [Online]
Available at: http://www.internetworldstats.com/emarketing.htm
[Accessed 04 12 2015].

Google Sites- Executive Office Chair, n.d. *Material Selection.* [Online]
Available at: https://sites.google.com/site/automaticdeskchair/hardware-implementation/mechanical-hardware
[Accessed 03 12 2015].

Hey, T. & Walters, P., 1987. *The Quantum Universe.* Cambridge: Cambridge University Press.

Honan, M., 2012. *How Apple and Amazon Security Flaws Led to My Epic Hacking.* [Online]
Available at: http://www.wired.com/2012/08/apple-amazon-mat-honan-hacking/
[Accessed 04 12 2015].

Internet Live Stats, 2015. *Internet Live Stats.* [Online]
Available at: http://www.internetlivestats.com/internet-users/
[Accessed 04 12 2015].

Internet Live Stats, 2015. *United States Internet Users.* [Online]
Available at: http://www.internetlivestats.com/internet-users/united-states/
[Accessed 04 12 2015].

Kahn, D., 1996. In: *The Code Breakers - The Comprehensive History of Secret Communication from Ancient Times to the Internet.* New York: Simon and Schuster, p. 1145.

Kantke, M., 2015. *Lost in the Cloud: Dropbox, Data "Insecurity," and Employee Shenanigans.* [Online]
Available at: http://abovethelaw.com/2015/01/lost-in-the-cloud-dropbox-data-insecurity-and-employee-shenanigans/
[Accessed 04 12 2015].

Khan Academy, n.d. *Jourrney into Cryptography.* [Online]
Available at: https://www.khanacademy.org/computing/computer-science/cryptography
[Accessed 4 12 2015].

LastPass, n.d. *LastPass.* [Online]
Available at: https://lastpass.com/
[Accessed 04 12 2015].

LastPass, n.d. *The Last Password You Have to Remember.* [Online]
Available at: https://lastpass.com/how-it-works/
[Accessed 04 12 2015].

McAlone, N., 2015. *Here's how to find out if your Netflix was hacked — and fix it.* [Online]
Available at: http://uk.businessinsider.com/heres-how-to-find-out-if-your-netflix-was-hacked-and-fix-it-2015-12?r=US&IR=T
[Accessed 4 12 2015].

Merriam Webster, n.d. *Dictionary.* [Online]
Available at: http://www.merriam-webster.com/
[Accessed 04 12 2015].

Miller, J., n.d. *Elasticity. Hook's Law. Tensile, compressive and shear stresses. Strain. Elastic, shear and bulk modulus..* [Online]
Available at: http://www.solitaryroad.com/c1020.html
[Accessed 04 12 2015].

Munson, L., 2014. *Average person has 19 passwords – but 1 in 3 don't make them strong enough.* [Online]
Available at: https://nakedsecurity.sophos.com/2014/10/17/average-person-has-19-passwords-but-1-in-3-dont-make-them-strong-enough/
[Accessed 04 12 2015].

Okyle, C., 2015. *Password Statistics: The Bad, the Worse and the Ugly (Infographic).* [Online]
Available at: http://www.entrepreneur.com/article/246902
[Accessed 04 12 2015].

Oxford Math Center - Emory, n.d. *Euler's Phi Function and the Chinese Remainder Theorem.* [Online]
Available at: http://www.oxfordmathcenter.com/drupal7/node/172
[Accessed 4 12 2015].

Potter, R., 2010. *Too many passwords to remember.* [Online]
Available at: http://www.theguardian.com/technology/askjack/2010/sep/30/password-management-internet
[Accessed 04 12 2015].

Reuters, 2015. *5.6 million fingerprints stolen in U.S. personnel data hack: government.* [Online]
Available at: http://www.reuters.com/article/us-usa-cybersecurity-fingerprints-idUSKCN0RN1V820150923
[Accessed 04 12 2015].

Rubenking, N. J., 2015. *Survey: Hardly Anybody Uses a Password Manager.* [Online]
Available at: http://securitywatch.pcmag.com/security-software/332517-survey-hardly-anybody-uses-a-password-manager
[Accessed 04 12 2015].

Scientific American, 2010. *Record 232-digit number from cryptography challenge factored.* [Online]
Available at: http://blogs.scientificamerican.com/observations/record-232-digit-number-from-cryptography-challenge-factored/
[Accessed 4 12 2015].

Singh, S., 1999. *The Code Book - The Science of Secrecy from Ancient Egypt to Qyantum Cryptography.* London: Fourth Estate Limited.

Statista, n.d. *Vendors' sales of mobile phone sales to end users worldwide from 2010 to 2015 (in million units), by quarter.* [Online]
Available at: http://www.statista.com/statistics/263355/global-mobile-device-sales-by-vendor-since-1st-quarter-2008/
[Accessed 04 12 2015].

Statistic Brain, 2015. *Computer Sales.* [Online]
Available at: http://www.statisticbrain.com/computer-sales-statistics/
[Accessed 4 12 2015].

Strohmeyer, R., n.d. *The 7 Worst Tech Predictions of All Time.* [Online]
Available at: http://www.pcworld.com/article/155984/worst_tech_predictions.html
[Accessed 04 12 2015].

SurveyMonkey, n.d. *Make Better Decisions with the UK's Leading Survey Platform.* [Online]
Available at: https://www.surveymonkey.com/
[Accessed 04 12 2015].

United States Department of Justice, n.d. *Online Identity Theft.* [Online]
Available at: http://www.justice.gov/criminal-fraud/identity-theft/identity-theft-and-identity-fraud
[Accessed 04 12 2015].

University of Berkely, n.d. *RSA.* [Online]
Available at: http://mathcircle.berkeley.edu/BMC3/rsa/node4.html
[Accessed 4 12 2015].

University of California San Diego, 2008. *Lecture Notes on Cryptography.* [Online]
Available at: https://cseweb.ucsd.edu/~mihir/papers/gb.pdf
[Accessed 4 12 2015].

University of Cambridge, n.d. *Young's Modulus - Density.* [Online]
Available at: http://www-materials.eng.cam.ac.uk/mpsite/interactive_charts/stiffness-density/NS6Chart.html
[Accessed 04 12 2015].

Wikimedia, n.d. *Frequency Analysis.* [Online]
Available at:
https://upload.wikimedia.org/wikipedia/commons/d/d5/English_letter_frequency_%28alphabetic%29.svg
[Accessed 4 12 2015].