The Weaknesses of Passwords and Alternative Authentication Technologies

Andrew Gareth Harris

Submitted in partial fulfilment of the requirements of Edinburgh Napier University for the Degree of Master of Science in Advanced Security and Digital Forensics

School of Computing

December 2014
Authorship Declaration

I, Andrew Gareth Harris, confirm that this dissertation and the work presented in it are my own achievement.

Where I have consulted the published work of others this is always clearly attributed;

Where I have quoted from the work of others the source is always given. With the exception of such quotations this dissertation is entirely my own work;

I have acknowledged all main sources of help;

If my research follows on from previous work or is part of a larger collaborative research project I have made clear exactly what was done by others and what I have contributed myself;

I have read and understand the penalties associated with Academic Misconduct.

I also confirm that I have obtained informed consent from all people I have involved in the work in this dissertation following the School's ethical guidelines

Signed:

Date: 7th December 2014

Matriculation no: 40098597
Data Protection Declaration

Under the 1998 Data Protection Act, The University cannot disclose your grade to an unauthorised person. However, other students benefit from studying dissertations that have their grades attached.

Please sign your name below one of the options below to state your preference.

The University may make this dissertation, with indicative grade, available to others.

The University may make this dissertation available to others, but the grade may not be disclosed.

The University may not make this dissertation available to others.
Abstract

Cyber-crime costs businesses millions in lost revenue. The number of attacks on businesses is increasing every day. Text-based passwords are one of the key targets of these attacks. This work investigates the cryptographic weaknesses of text-based passwords to demonstrate their vulnerability. This work also considers the social weaknesses of text-based passwords, concluding that users are hampered by their own cognitive limitations and poor security awareness.

Evaluation of two alternative authentication technologies concludes that despite the inherent strengths of each alternative technology, neither would be capable of replacing text-based password use.

Future system design should include a detailed evaluation of the available authentication methods and the final choice made with the user base and the intended use of the system in mind.
# Table of Contents

Authorship Declaration ......................................................................................... 2

Abstract .................................................................................................................. 4

Acknowledgements ................................................................................................. 8

1  Introduction ........................................................................................................ 9
   1.1  Background .................................................................................................. 9
   1.2  Objectives .................................................................................................. 12
   1.3  Structure ..................................................................................................... 13

2  Cryptographic Weaknesses ............................................................................... 14
   2.1  Introduction ............................................................................................... 14
   2.2  Password Cracking & Attack Methods ...................................................... 15
      2.2.1  Brute-force Attacks ............................................................................ 17
      2.2.2  Dictionary Attacks .............................................................................. 18
      2.2.3  Hybrid Dictionary Attack .................................................................... 19
      2.2.4  Rule-based Attack .............................................................................. 20
      2.2.5  Rainbow Tables .................................................................................. 21
      2.2.6  Markov Chains ................................................................................... 24
   2.3  Password Cracking Tools ............................................................................ 26
      2.3.1  Kali Linux ............................................................................................ 26
      2.3.2  John the Ripper & Hashcat ................................................................. 26
      2.3.3  Graphical Processing Units ................................................................. 27
   2.4  Moore’s Law ............................................................................................... 30
   2.5  Can Passwords Be Secure? ........................................................................ 32
      2.5.1  Secure Businesses .............................................................................. 32
      2.5.2  Secure Users ...................................................................................... 33

3  Social Weaknesses ............................................................................................. 34
   3.1  Introduction ............................................................................................... 34
   3.2  Social Engineering ..................................................................................... 34
   3.3  Cognitive Abilities ..................................................................................... 36
   3.4  Password Reuse & Sharing ....................................................................... 38
   3.5  User Awareness ......................................................................................... 39
   3.6  Password Leaks ......................................................................................... 41

4  Alternative Authentication Methods ................................................................. 45
   4.1  Biometrics ................................................................................................. 46
      4.1.1  Strengths .............................................................................................. 48
      4.1.2  Weaknesses ......................................................................................... 49
      4.1.3  Recent Advances ............................................................................... 51
   4.2  Multi-factor Authentication ....................................................................... 53
      4.2.1  Strengths .............................................................................................. 55
      4.2.2  Weaknesses ......................................................................................... 56
      4.2.3  Recent Advances ............................................................................... 57

5  Barriers to Adoption ......................................................................................... 58
   5.1  Diverse Requirements ............................................................................... 58
   5.2  Competing Technical Proposals ................................................................. 58
   5.3  Customer Loyalty ....................................................................................... 59
   5.4  Competing Goals Between Stakeholders ................................................... 59
5.5 User Reluctance........................................................................................................ 59
5.6 No Control of End-user Platforms ......................................................................... 60
5.7 No Single Organisation Can Impose a Solution ....................................................... 60

6 Conclusion .................................................................................................................... 61
   6.1 Summary of Work Completed .................................................................................. 61
   6.2 Further Investigation ................................................................................................. 62

7 Recommendations ......................................................................................................... 63
   7.1 Extent To Which Aims Were Met ............................................................................. 64
   7.2 Personal Reflection ................................................................................................... 64

Bibliography ...................................................................................................................... 66
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Operation of MD5 Hash Algorithm</td>
<td>14</td>
</tr>
<tr>
<td>2-2</td>
<td>Brute Force Attack</td>
<td>17</td>
</tr>
<tr>
<td>2-3</td>
<td>Dictionary Attack</td>
<td>18</td>
</tr>
<tr>
<td>2-4</td>
<td>Hybrid Dictionary Attack</td>
<td>19</td>
</tr>
<tr>
<td>2-5</td>
<td>Rules Based Attack</td>
<td>20</td>
</tr>
<tr>
<td>2-6</td>
<td>Selection of Most Popular Rules</td>
<td>20</td>
</tr>
<tr>
<td>2-7</td>
<td>Sizes of MD5 Rainbow Tables</td>
<td>22</td>
</tr>
<tr>
<td>2-8</td>
<td>Unsalted Password</td>
<td>22</td>
</tr>
<tr>
<td>2-9</td>
<td>Salted Password Hashes</td>
<td>23</td>
</tr>
<tr>
<td>2-10</td>
<td>Official Hashcat Benchmarks</td>
<td>28</td>
</tr>
<tr>
<td>2-11</td>
<td>Exponential Growth of Transistor Count 1971-2011</td>
<td>30</td>
</tr>
<tr>
<td>3-1</td>
<td>Butler &amp; Butler's Password Performance Model</td>
<td>41</td>
</tr>
<tr>
<td>3-2</td>
<td>Splash Data's Top 20 Worst Passwords 2013</td>
<td>43</td>
</tr>
<tr>
<td>3-3</td>
<td>Brute Force Attack Using Hashcat</td>
<td>43</td>
</tr>
<tr>
<td>3-4</td>
<td>Recovered Passwords From Hashcat</td>
<td>43</td>
</tr>
<tr>
<td>4-1</td>
<td>Biometric Device Operation</td>
<td>47</td>
</tr>
<tr>
<td>4-2</td>
<td>ATM Machine Using Multi-Factor Authentication</td>
<td>53</td>
</tr>
</tbody>
</table>
Acknowledgements

I would like to express my gratitude and thanks to my supervisor Frank Grieg. I am extremely thankful for his guidance, invaluable constructive feedback and genuine willingness to support me throughout this project. I will sincerely miss our Skype video calls. I also extend my thanks to the staff at Edinburgh Napier University for their support throughout the last three years.

I would like to thank my family and friends for their support whilst I have completed this work. I cannot thank enough, my parents, Sheila and Gareth, who have always supported me in whatever I set out to achieve.

Finally, special thanks to Rachael, I couldn’t have done this without your love and support, for that, I will be forever grateful.
1 Introduction

1.1 Background

The estimated cost of cyber crime to the global economy in 2013 was $375 billion (McAfee, 2014). This conservative estimate includes not only the money gained by cyber criminals, but it includes revenue lost by a business, intellectual property theft and the multiple costs that are incurred as a result of identity theft. In the same year, over 552 million identities were breached putting a range of consumer information into the hands of the criminal underground. The effects of these attacks are devastating for both businesses and consumers (Symantec, 2014). Data breaches are the most costly and are often the result of weak and insecure passwords.

Today we live in a society considered to be ‘online’. The number of services and consequent user base has grown exponentially over the last decade to the point where there is a reliance on computer systems in everyday lives. With this shift in trend brings great variety to the user base. Computers are no longer reserved for enthusiasts or certain stereotypes, so much so that owning the latest smartphone or tablet is considered essential and as important as any other fashionable trend.

This change in attitude towards technology use, coupled with increased accessibility and interconnected systems, means that many traditional offline services are now available ‘online’. Three areas specifically falling within this trend are Internet banking, ecommerce and social media. Businesses have invested heavily in increasingly more mobile and user-friendly systems with the intent of attracting a wide spectrum of users, regardless of their technological prowess.

Despite each service being targeted to a significantly varied spectrum of users, each service differs wholly in both design and purpose. One thing they do share in common is that any user of such a service needs to prove who they are before they can use or access the service: the requirement of authentication. Authentication by definition is to prove or show (something) to be true, genuine, or valid. In a computer system, this is almost exclusively achieved through knowledge-based authentication (‘what you know’) typically through the use of a username and a password or PIN (Personal Identification Number) code.

In the last two years, there have been a number of successful and high profile cyber attacks against large, well-known businesses. In each instance, sensitive and personal data was stolen and then leaked or sold across the Internet. As expected, these attacks generated substantial and negative press that has had long-term damaging affects on customers’ perception of the business and the services offered, in addition to the financial loss.
In December 2013, American retail juggernaut Target suffered a catastrophic data breach that resulted in the theft and sale of millions of customer credit card details. The results were financially devastating with the following quarter revenue down over 800 million US dollars from the year earlier (Sharf, 2014). This figure does not include the decline in share price and subsequent boycott of the store due to loss of customer confidence. After much speculation, security researcher Brian Krebs reported that the attackers had gained access to the company’s network using a username and password stolen from a third party company that had sufficient access rights (Krebs, 2014). The reality was that a simple username and password was all that stood between the attackers and nearly a billion dollars worth of damage.

Knowledge-based authentication through the use of passwords has been prevalent in computer systems for over 60 years and yet despite suffering from a number of issues, their continued use appears to have no end. After a largely stagnant period there has been a recent surge in interest from academics and researchers to find potential alternatives. Internationally renowned security technologist Bruce Schneier agrees that passwords “are not going away anytime soon” and suggests a strategy where users are educated on what constitutes a weak password and how easily they can be broken (Schneier, 2014). In comparison, pressure from the financial and ecommerce sector has forced businesses to develop stronger and more robust authentication methods in an attempt to mitigate risk and combat fraud (Hiltgen et al., 2006).

Possible alternatives to passwords already exist. These include, but are not limited to using biometrics or multi-factor authentication. Biometrics uses certain characteristics unique to the individual user. This could be a fingerprint or retinal scan. Multi-factor authentication extends simple password use by requiring an additional piece of information the user either knows, has access to, or may be. Biometrics can form part of multi-factor authentication, or it can be a standalone authentication method in its own right. Demand for an alternative solution often results from dissatisfaction with the existing one. The inability to pinpoint the exact problem with passwords is a thorn in the side of researchers and industry professionals alike. Concise understanding of the current problems is critical if alternative solutions are to be proposed.

In 1999, Adams and Sasse concluded that password policies were causing high levels of dissatisfaction which led to insecure security practices and low motivation from staff members to follow security protocols. These insecurities often included the writing down of passwords, re-using the same, or similar passwords across multiple services and choosing simple, easy to guess passwords (Adams & Sasse, 1999).
In the 2000’s many industry professionals, including Microsoft CEO, Bill Gates declared this to be a ‘non-problem’, claiming that biometrics would resolve the password problem, insisting that authentication would evolve to a time where users would no longer be required to remember anything (Sasse, 2014). Now, over a decade later the password burden still continues to weigh down on organisations and users alike with no clear industry-wide accepted or embraced alternative.

Public perception of hackers is extremely inaccurate. There is a belief that computer hacking is complex and requires a skill level and tool set that very few possess. The reality is that anybody with a computer and access to the Internet can download a range of powerful password cracking tools, obtain password sets from previous attacks and engage in forums and message boards that provide help and guidance on hacking.

Text-based passwords are insecure from both a cryptographic and a social standpoint. Their continued use is having severe financial implications, costing businesses millions of pounds a year. These weaknesses will be examined and two alternative authentication methods will be investigated and explored.
1.2 Objectives
This report investigates the insecurities of text-based passwords, examining alternative authentication technologies and ascertaining any barriers that are prevent their widespread adoption.

This will be achieved by:

- Exploring and analysing password weaknesses from a cryptographic and social standpoint
- Investigating and evaluating two alternative authentication methods: biometrics and multi-factor authentication
- Examining any barriers of adoption of the aforementioned alternative technologies
- Concluding whether either of the aforementioned alternative authentication methods has the potential to replace text-based passwords as a primary authentication method, or if not, recommendations to improve password selection and their use from a security perspective.
1.3 Structure

Chapter Two investigates the cryptographic weaknesses of text-based passwords. This investigation will include the evaluation of a range of different techniques and tools used in password cracking, including the recent emergence of password cracking using GPUs (Graphics Processing Unit). This chapter determined the effect Moore’s law has on password cracking and will conclude by determining that secure passwords really do exist.

Chapter Three explores the weaknesses of passwords from a social standpoint, examining the relationship between a user’s cognitive abilities and how this affects and influences their own security practices. This chapter ascertains whether a balance between usability and security exists, debunking the myth that both are mutually exclusive.

Chapter Four critically evaluated alternative authentication methods that use biometrics and multi-factor authentication. This chapter evaluated the strengths and weaknesses of each technology in comparison to text-based passwords.

Chapter Five determined whether one of these alternative authentication technologies had the potential to replace text-based passwords to become an industry-wide standard authentication method. If not, what barriers to its adoption exist?

Chapter Six concludes whether passwords are safe enough for continued use in today’s society and whether there is a likelihood of an alternative authentication method being adopted within the near or distant future.

Finally, Chapter Seven makes recommendations based on the findings of this work. These recommendations include how secure use of text-based password can be achieved and when it is appropriate to use alternative authentication methods.
2 Cryptographic Weaknesses

2.1 Introduction

The simple, text-based password is the authentication method of choice for modern computer systems, despite suffering from numerous problems that suggest their days may be numbered. Modern authentication is often comprised of two parts: identification, usually in the form of a user ID and authentication, typically a secret, text-based password supposedly known only to the user.

Text-based passwords used in authentication date back as far as the 1960’s, although security was not the forefront of their implementation. Passwords were primarily used to control users’ use of the limited resources available whilst operating the first multi-user computer: the Compatible Time Sharing System (CTSS) at MIT.

Perhaps uncoincidentally, the want and desire to compromise computer passwords has existed for just as long. A graduate-student wanting to obtain more resource than he had been allocated admitted some years later that by exploiting a bug in the system code, he was able to obtain a copy of the password file. This enabled him to use resources allocated to other users (McMillan, 2012). Unsurprisingly, the password file was stored in plaintext; neither encrypted nor hashed.

Modern passwords are typically implemented using a cryptographic one-way hash. Figure 2-1 shows the operation of the popular MD5 hash algorithm. A hash algorithm takes a user’s plaintext password of any size and complexity and creates a fixed-length hash value, which is then stored, usually in a database. This hash value will typically contain a fixed-length jumble of alphanumeric characters that will appear meaningless to those who read it, as a result, it is considered impossible to reverse engineer a hashed password (Yiannis, 2013).

Passwords are more vulnerable then they have ever before due to a number of readily available and very powerful password cracking programs available to download from the Internet. These programs can be run on standard computers with no special hardware required. Security professionals typically develop these programs with their use intended to be in a legal capacity. Unsurprisingly, an increasing number are being used for malicious activity.
Many of these programs are developed with a user-friendly and intuitive graphical user interface (GUI) making them extremely easy to use. The more powerful and advanced tools still use a traditional command line interface (CLI). With a basic understanding of Linux it is still easy to launch many of the basic attacks without needing to be an expert.

The perception exists that computer hackers are an elite band of Internet criminals, highly skilled and equipped with complex software and powerful hardware. Whilst there are a very small number of hackers who fit this stereotype, the majority are people of average IT intellect, using a commercially available computer, a word list downloaded from the Internet and a copy of John the Ripper.

The Internet has a plethora of password cracking guides and a number of forums and messages boards are dedicated to hacking. These resources are often free to access and any user with basic IT knowledge has the ability to launch a potentially dangerous attack. Movies like Swordfish and Hackers have helped shape the general public’s perception of computer hackers with many simply unaware of the availability of tools and simplicity of many attacks.

Text-based passwords are vulnerable to a multitude of different attacks. These types of attacks have changed dramatically over time and are continually evolving to keep up to speed with technological advances. The most popular attack is password cracking, an attack method that has been around almost as long as passwords themselves.

### 2.2 Password Cracking & Attack Methods

Password cracking is often confused with cryptanalysis. Cryptanalysis attempts to find a weakness or insecurity of a cryptographic scheme. Password cracking is the science of finding the most effective automated way of reverse-engineering encrypted passwords back into human-readable plaintext.

Password cracking attacks are predominantly split into two categories: online attacks and offline attacks. Online attacks are performed against a live host or system, typically attacking a login form or session. Various protection methods are implemented to help thwart online attacks; these include Captcha images, enforcing a timeout period after a certain number of unsuccessful login attempts or even locking the account altogether (Corella, 2007). An administrator or the user can then unlock these accounts, only usually after completing several extra verification steps to prove their identity. One weakness of these countermeasures is the risk that an online password attack may inadvertently turn into a denial-of-service (DoS) attack. This risk can easily be mitigated through the correct implementation and configuration of network security devices. Due to the aforementioned countermeasures featuring prominently across most live systems, offline password cracking is considered to be more effective and yield a greater success rate than online attacks.
Offline attacks are typically performed on password hashes without the need to directly interact with a live host or system. An offline attack often follows a successful online attack where an attacker is in possession of a single hash, a database or a password file. Success rate of an attack depends on several key factors; including whether the attacker is attempting to crack a single or a group of passwords and what time and resources are available to them. Large, heavily funded cyber-crime groups often have access to significantly greater resources than the average home user, making previously unfeasible attacks a real possibility.

Password cracking is both a science and an art. Efficiency is vital. The end goal is to recover a hash value as quickly as possible, with the fewest number of computations. Attack methods are continually being improved, through both cryptographic research and technological advancement. There are a number of different attack methods that differ in both complexity and efficiency.

Unless explicitly stated, it is implied that these attack methods can be applied to a range of hashes or on files that have been recovered from a range of places. They do not consider where the password originally came from, what service or system it is used for, or the hash algorithm used on the plaintext password.
2.2.1 Brute-force Attacks

Brute-force attacks are one of the most popular methods of password cracking, mainly due to their simplicity. An attack performs an exhaustive search for the target hash by calculating every possible hash combination for a predefined character set and key length. These calculated hashes are then compared against the target hash until a matching hash is found. For example, a brute-force attack with a character set of all available lowercase and uppercase numbers with a length of 8 characters using characters from the Latin alphabet would begin an attack with ‘aaaaaaaa’ and end with ‘ZZZZZZZZ’. Figure 2-1 displays the steps of a brute-force attack.

The main advantage of a brute-force attack is that it can theoretically guarantee a 100% success rate. Nonetheless, success time can range from seconds to centuries, dictated by length of the password subject to the attack. Using the aforementioned character set, $53,459,728,531,456$ hashes would have to be computed and compared to exhaust the key space, guaranteeing a successful attack ($52^8$). In practice a password will likely be cracked long before the key space is exhausted.

The success rate becomes exponentially worse as additional characters are added to the password string. For example, adding just two additional characters, increasing the password length from eight to ten characters will increase the possible number of combinations, and resulting calculations necessary from $53$ trillion to $144$ quadrillion.

Increasing the key space directly decreases the likelihood of a successful attack due to the increased computation time. Nevertheless, analysis of 130 million passwords leaked as a result of a successful attack against Adobe revealed that users continually opt for passwords between six – eight characters, often using only lower case character sets or numbers, rather than a mixture (Gosney, 2014).

Research by MandyLion determined that the average number of keys a commercially available desktop computer can try efficiently is around $17$ billion per hour. A password comprised of between six and eight lowercase characters can be recovered in as little as few seconds, exhausting the entire keyspace in as little as 6 hours (MandyLion, 2010).
A brute-force attack is considered to be one of the easier and more accessible password cracking attacks due to the availability of a wide range of tools available on the Internet. Its success rate relies on a number of factors, including resources available and the length and complexity of the password under attack. Despite its theoretical 100% guaranteed success rate it is considered an exceedingly inefficient attack method due to the number of wasted computations trying a large number of strings that are unlikely to be real-world passwords.

2.2.2 Dictionary Attacks

Due to the inefficiency of brute-force attacks, Figure 2-3 shows one of the more popular attack methods, a dictionary attack. The key difference between the two methods is that a dictionary attack is launched with an accompanying file containing a pre-defined list of probable password matches, as opposed to trying every possible password combination.

![Dictionary Attack Diagram](image)

To prevent a dictionary attack turning into a brute-force attack, the dictionary file used during the attack needs to be both well balanced and optimised. A dictionary file is more commonly known as a ‘word list’. A word list is plain text file where each individual row is comprised of a single password to be used during an attack.

There are a large variety of word lists available to download from the Internet and many popular word lists are bundled into security tools by default. John the Ripper is one of these tools and is used by attackers and security professionals. John the Ripper comes bundled with several different word lists as part of its default installation and can be optimised specifically for dictionary attacks (Openwall, 2014).

Traditionally, the key source used to build a word list was the English dictionary (Morris & Thompson, 1979). As a result of a number of high profile password leaks, many of the passwords recovered during those attacks are typically included in modern word list files. Passwords recovered from leaks are considered a valuable resource as they show the experiential password choices and trends of real end-users. In practice, these lists often yield a greater success rate than randomly generated word lists (Singer & Anderson, 2013). Crackstation has compiled one of the most popular combined word lists available. The word list contains just fewer than 1.5 billion different strings made up of every word list, dictionary and password database leak, and every word from the Wikipedia database up until 2010 in multiple languages (Crackstation, 2011).
The success rate of a dictionary attack is dependent on the word list in use, the number of hashes to be recovered and ultimately, the strength of the passwords contained within those hashes. Dictionary attacks are still popular and still pose a very real threat as many leaked password sets confirm that many users chose passwords made up of lexicon from the English dictionary. This attack method, like a brute-force attack is easy to launch and can be highly successful.

2.2.3 Hybrid Dictionary Attack

Figure 2-4 demonstrates a hybrid dictionary attack. This attack method combines certain characteristics of both a brute-force attack and a dictionary attack. Similar to a dictionary attack, it uses a word list rather than attempting to exhaust an entire keyspace. In a brute force manner a hybrid dictionary attack appends additional strings to each entry in the word list.

These additional strings are generated based on a simple rule set. The most popular basic rule set simply appends numbers to the start or end of each word list entry. Using this basic rule the word list entry ‘password’ produces the additional strings ‘password1’ through to ‘password100’ at attack time.

Attackers will typically use more focussed word lists consisting of the more popular well-known passwords. This is critical to the performance of an attack as both time and computations increase exponentially with each additional character that is added within the rule set. Appending additional strings to a comprehensive word list can render a hybrid dictionary attack as inefficient, reducing performance to that of a brute force attack.

The hybrid dictionary attack is a popular attack method as users are frequently being encouraged to increase the complexity of their passwords. In many instances, users choose to simply append numbers to the end of existing lexical passwords. Passwords of this entropy are considered the most vulnerable to a hybrid dictionary attack, alongside passwords that consist of purely lower or uppercase alphabetical characters.

Like its counterparts, a hybrid dictionary attack is easy to launch, quick in operation when optimised and is prevalent in a number of easily available tools. Simply adding a number to the end of a password is no longer enough to thwart this attack method.
2.2.4 Rule-based Attack

The rule-based dictionary attack method has seen an increase in popularity over the last decade. This increase has been linked to the leaking of large password sets on to the Internet. These leaks revealed first hand, genuine password trends and users password habits. The analysis of these sets has influenced the rule sets found in many popular password cracking programs.

Figure 2-5 displays each step of a rule-based attack. This attack method shares common characteristics of a hybrid dictionary attack, although it is considerably more efficient as it has no brute force element. Rather than unintelligently appending additional strings to each entry in a word list in a brute force manner, the attack uses a complex rule set that transforms word list entries at attack time. These complex rule sets are considered to be akin to a programming language. They are comprehensive and highly customisable, facilitating the ability to modify, transform, cut or extend entries from a word list.

A uniform rule-engine is implemented across the three most popular password cracking programs: Hashcat, John The Ripper and PasswordsPro. This encouraged collaborative development and resulted in a community-developed, highly efficient rule-set (Hashcat, 2014). The table in Figure 2-6 shows the most popular functions of the many rules available in the default rule-set.

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
<th>Description</th>
<th>Example Rule</th>
<th>Input Word</th>
<th>Output Word</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nothing</td>
<td>:</td>
<td>Do nothing</td>
<td>:</td>
<td>Pa55word1</td>
<td>Pa55word1</td>
</tr>
<tr>
<td>Lowercase</td>
<td>l</td>
<td>Lowercase all letters</td>
<td>l</td>
<td>Pa55word1</td>
<td>pa55word1</td>
</tr>
<tr>
<td>Uppercase</td>
<td>u</td>
<td>Uppercase all letters</td>
<td>u</td>
<td>Pa55word1</td>
<td>PA55WORD1</td>
</tr>
<tr>
<td>Capitalise</td>
<td>c</td>
<td>Capitalise first letter only</td>
<td>c</td>
<td>Pa55word1</td>
<td>Pa55word1</td>
</tr>
<tr>
<td>Invert Capitalise</td>
<td>C</td>
<td>Lower first, rest uppercase</td>
<td>C</td>
<td>Pa55word1</td>
<td>pA55WORD1</td>
</tr>
<tr>
<td>Append Character</td>
<td>$X</td>
<td>Append character X to end</td>
<td>$1</td>
<td>Password</td>
<td>Password1</td>
</tr>
<tr>
<td>Prepend Character</td>
<td>^X</td>
<td>Prepend character X to front</td>
<td>^1</td>
<td>Password</td>
<td>1Password</td>
</tr>
</tbody>
</table>

Figure 2-6 - Selection of Most Popular Rules
Rule based dictionary attacks are considered highly efficient and require significantly less computation time than brute-force and hybrid dictionary attacks. Their weaknesses are similar to a dictionary attack, wherein if the password is not present in the word list, then no matter which rule is applied to it, the hash will never be recovered.

Rule set development is difficult and is comparable in technicality to computer programming. Although the most popular password cracking tools are bundled with comprehensive rule-lists, applying each rule to a vast word list will ultimately reduce the efficiency. Successfully optimising a rule-set is challenging. Having too few rules will reduce the likelihood of a successful recovery whilst having too many rules will increase inefficiency.

The biggest advantage of this attack method is that theoretically, passwords of varying complexities can be recovered. Those passwords that consist of lowercase lexicon with a capital letter or number appended or prepended might appear to be strong and to satisfy complexity requirements, when in reality, will still be recovered just as quickly as a lowercase lexical password.

2.2.5 Rainbow Tables

In 1980, Hellman introduced a method of recovering a plaintext from its original hash value that involved a trade-off between computation time and storage space (Hellman, 1980). This method evolved into what became the Time Memory Trade-Off (TMTO) attack.

Using a combination of Hellman’s method and the improvements made by Rivest, Oechslin proposed a faster cryptanalytical time-memory trade-off process. This method has since been implemented into many popular password recovery tools (Oechslin, 2003). The pre-computed tables generated as a result of Oechslin’s method are known as rainbow tables.

Precompiling all possible password hash values for an exhaustive search and storing them in a lookup table drastically reduces the time that would be required for an attack. The time-memory trade-off involves using pre-computed, partially stored hash tables with the relevant missing parts computed at attack time. The benefit of this is the hashes only need to be computed once; the same table can be used again and again, compared to a brute force attack, where hashes are computed each time an attack is launched.

Due to storage constraints, rainbow tables are only thought to be effective against passwords up to nine characters in length. However, success is largely dependant on the character set of the password. The table in Figure 2-7 lists a small selection of rainbow tables available specifically for the MD5 hash algorithm (Rivest, 1992) that can be downloaded from the Internet. The table outlines the number of characters used in the set, the minimum and maximum password length and the size of the rainbow table.
These tables can be downloaded free of charge from www.freerainbowtables.com.

<table>
<thead>
<tr>
<th>Rainbow Table</th>
<th>Available Characters</th>
<th>Min/Max Length</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>md5_alpha-space#1-9</td>
<td>27</td>
<td>1-9</td>
<td>24GB</td>
</tr>
<tr>
<td>md5_mixalpha-numeric-all-space#1-7</td>
<td>63</td>
<td>1-7</td>
<td>86GB</td>
</tr>
<tr>
<td>md5_mixalpha-numeric-all-space#1-8</td>
<td>63</td>
<td>1-8</td>
<td>1049GB</td>
</tr>
<tr>
<td>md5_mixalpha-numeric-all-space#1-9</td>
<td>63</td>
<td>1-9</td>
<td>253TB*</td>
</tr>
</tbody>
</table>

Figure 2-7 - Sizes of MD5 Rainbow Tables

Once generated, rainbow tables perform significantly faster than a traditional brute-force attack, however as the table above confirms, as the length of the password increases, so does the table size. Generating rainbow tables for passwords over ten characters in length with an increased character set requires enormous storage space. This is a severe limitation of rainbow tables.

To combat the threat of rainbow tables, the concept of password salting was introduced.

2.2.5.1 Adding Salt

A salt is additional piece of random data that is added to the input of a one-way hash algorithm before the password is hashed.

Figure 2-8 demonstrates where $H$ is the hash function, with $P_1$ and $P_2$ identical plaintext passwords of two different users, where no salting is applied $H(P_1)$ and $H(P_2)$ would generate the exact same hash.

* Theoretical size as table still under construction
Figure 2-9 demonstrates that using the same hash function $H$, now with $S$, a salt that will be concatenated with the password before a hash function is applied. Providing the salt values are different $H(P_1 + S)$ and $H(P_2 + S)$ will produce two different hashes even though the plaintext password is the same.

The main advantage of using a salt combined with a hash is that pre-computation attacks are virtually unusable. Using Oeschlin's time-memory trade-off technique, a table that has been built for a particular character set or specific hash algorithm would be useless, unless it was generated for that exact salt. Theoretically, to guarantee a successful recovery, an attacker would have to possess a set of tables for each salt combination. Due to the number of possible salt combinations and the sheer resource it would require to generate and store, such a set of tables is unlikely to exist.

Due to the weaknesses of their size and the use of password salting, rainbow tables are widely considered by security professionals to be obsolete. Whilst creating a rainbow table for each password salt is theoretically possible, it would be extraordinarily inefficient, in terms of both computation and storage.

Rainbow tables are one of the easiest password attacks to defend against, yet businesses do not take steps to mitigate the threat they pose. The Yahoo and LinkedIn database leaks in 2012 confirmed that large multi-national organisations are not salting their passwords. These incessant insecure password storage practices and end-users continually opting to use passwords between six and ten characters means that rainbow tables are still very much a threat.

Rainbow tables are often confused with the more the more primitive method using hash tables. In a hash table the password-hash pairs are computed and stored in a database, typically sorted by hash value. To recover a password hash, a binary search is performed on the table; if the hash is present the plaintext is returned. Whilst a hash table has several advantages, including its superior speed and choice of what passwords are included, for example, top 10,000 passwords, it’s biggest weakness is that it must store every single password-hash pair, therefore constructing a hash table for the entire key-space.
2.2.6 Markov Chains

Narayanan and Shmatikov were the first to propose using the standard Markov modelling techniques used in Natural Language Processing (NLP) and combining it with the time-memory trade-off method developed by Hellman. Their aim was to prove that fast and successful smart dictionary attacks would be feasible, regardless of the size of the key space which proved to be the main limitations of rainbow tables.

The theory of Markovian filtering is that any alphabetical password generated by a human is unlikely to be completely random. In fact, even if asked to create a completely random password, an English-speaking user will produce a sequence in which each character is roughly equidistributed with the frequency that it occurs throughout the English language (Narayanan & Shmatikov, 2005).

Using the Markovian theory, Narayanan and Shmatikov developed an algorithm that successfully recovered 67.6% of hashes from a password set of around four million using a search space of just $2 \times 10^9$. This was considerably higher than the percentage of passwords recovered by Ochsli’s ‘rainbow table’ attack, which at the time was fastest known technique for searching large key spaces.

In 2008, Marechal proposed a technique that used a combination of a brute-force attack and Markov chains. The first two characters of a password were generated via brute-force, with the remaining characters selected using Markovian theory (Marechal, 2008). The results were positive; the newly developed attack method significantly outperformed a standalone brute-force attack and reduced the time taken to recover the same amount of passwords compared to an attack using a rainbow table.

To yield an acceptable success rate, the Markov chains attack technique relies heavily on the targeted password set being predominately lexical. It can be argued that a Markov chains attack is little more than an optimised dictionary attack. Performance degrades to that of a standard brute-force attack when truly randomly generated passwords are used.

Despite this perceived limitation, recent analysis of a number of password sets made publicly available on the Internet confirmed conclusively that a large number of users choose alphabetical passwords, predominantly in lowercase. Where attempts have been made to increase complexity through numbers and symbols, these are almost exclusively appended to the beginning or end of lowercase lexicon. Passwords of this criterion are the most vulnerable to a Markov chains attack.

Hashcat, one of the most popular and widely used password cracking tools is one of a select few that is capable of using Markov chains as an attack method (Hashcat, 2014). Hashcat intelligently examines passwords that have already been cracked, performing a probabilistically ordered brute-force attack.
Comparably, John the Ripper uses an attack named ‘Incremental’ mode as default, this attack is based on trigraph frequencies. As part of a wider community “Jumbo patch”, John the Ripper has the capability to launch an attack using Markov chains (Openwall, 2014).

Whilst the Markov chains technique is proven to be more effective than many of the other cracking techniques described here, its inclusion remains absent in many password cracking programs. Where it is available as an attack method, it is often considered an afterthought and its use is not widely known, or publicised.
2.3 Password Cracking Tools

A number of password cracking tools can be downloaded free from the Internet. The public perception of computer hackers is very distorted. Participants in a study by Gaw & Felten believed that hackers are highly skilled cyber criminals, that use specialised tool sets and have access to super computers (Gaw & Felten, 2006). The reality is, anybody with a computer and an Internet connection can access to a range of tools and information about password cracking.

Many password cracking tools are developed by security professionals. The same tools in the hands of attackers turns them into dangerous, malicious programs capable of causing businesses disruption and revenue loss. Kali Linux is a great example of a suite of tools developed for IT security professionals that attackers use for malicious purposes.

2.3.1 Kali Linux

Kali Linux, formerly known as BackTrack is a Linux distribution developed and maintained by Offensive Security. Kali Linux is marketed as a “Penetration Testing and Security Auditing Linux Distribution” (Offensive Security, 2013).

Kali Linux is free to download and can be run off a standard laptop either installed from a USB stick or using virtualisation software. There is detailed documentation available from Offensive Security’s website and there are a number of books available to buy from online retailers. These resources are typically aimed at security professionals but there is nothing to stop potential attackers downloading and using tools for malicious purposes.

Kali Linux is equipped with over 300 penetration testing tools, 43 of those tools are categorised under the ‘Password Attack’ category. Of these 43, the two most commonly used password cracking tools are John the Ripper and Hashcat.

2.3.2 John the Ripper & Hashcat

John the Ripper is a password cracking tool available across a variety of computing platforms including Windows, Mac OS and Linux. John the Ripper was originally developed to aid system administrators in detecting weak Unix passwords, now it is used by penetration testers as a password auditing tool (Openwall, 2014). John the Ripper can recover a variety of different hash types including many of those used by commercial authentication services. John the Ripper is very powerful and supports all of the attack methods discussed above. There is extensive documentation available on the Internet about how to use John the Ripper and although it is not the most intuitive piece of software, there are many tutorials available online that instructs users on how to launch a variety of password attacks.
Hashcat is “the world’s fastest CPU-based password recovery tool”. Hashcat was developed in 2009 when near-perfect password recovery tools, including John the Ripper were readily available. However, none of these tools supported multi-threading that would enable the use of multiple cores of modern CPUs (Hashcat, 2014). Hashcat is capable of recovering over 90 different hash algorithms and supports all of the attack methods that have been discussed above. Hashcat is considered one of the easiest password recovery tools with comprehensive documentation and tutorials available on the Hashcat website.

In 2013, the development team responsible for Hashcat unveiled a new password recovery tool: oclHashcat. oclHashcat was the first password cracking tool that utilised the computational power of a Graphical Processing Unit (GPUs) for the generation of hashes. Its release changed the landscape of password cracking forever.

### 2.3.3 Graphical Processing Units

GPUs have been prevalent since the boom of the computer games and graphic design industry in the nineties, due to their ability to calculate floating-point operations and undertake parallel processing. Whilst these industries were able to harvest the power of GPUs, the cryptographic community was unable to utilise the power due to a lack of a user-friendly Application Programming Interface (API). In 2006, NVIDIA unveiled its parallel computing platform known as CUDA (Compute Unified Device Architecture), making it possible for the first time for high level computer code to be sent straight to the GPU without the need for assembly language, making it easier and more user friendly to integrate with existing computer code (NVIDIA, 2006).

Harrison and Waldron were the first to demonstrate that GPUs could far outperform high-performance Central Processing Units (CPUs) on symmetric cryptographic computations (Harrison & Waldron, 2008). Furthering the work of Harrison and Waldron, Li developed the first efficient implementation for the MD5-RC4 algorithm; this is considered the first successful implementation of GPU hash generation for a one-way cryptographic hash algorithm (Li, 2009). Their combined work was the foundation for what evolved into modern GPU password cracking. The development team behind the password cracking tool Hashcat released a GPU-based tool in 2013 aptly named oclHashcat, capable of computing billions of hashes per second for a over a 100 of the most widely implemented hash algorithms (oclhashcat, 2013).

The biggest advantage of GPU password cracking is the speed at which hashes are computed when compared to a CPU. On average, a GPU is typically up to 30 times faster than an identically priced CPU (Sprengers, 2012). Early optimised implementations for the DES and AES symmetric encryption algorithms were capable of reaching speeds of up to 60 times that of a standard CPU (Yang & Goodman, 2007).
To illustrate the difference in speed, using oclHashcat’s benchmarks against the still widely used MD5 hash algorithm: it can be calculated that a computer with a single AMD HD7970 GPU is capable of computing 7894M hashes per second. Consider that the total number of possible combinations for a password up to eight characters in length that contains a mixture of uppercase, lowercase and/or numbers is just over 220 trillion. The entire key space can be searched in approximately 7.8 hours. Furthermore, 7.8 hours is the time taken to try every possible character set combination. Theoretically, a successful recovery could just as likely occur during the first hash as it could the last.

In comparison, a computer equipped with a high end CPU produces significantly slower results. The number of hashes that can be computed for the same MD5 algorithm is 86M hashes per second. This is around 91 times slower compared to a computer with a single GPU. In comparison to the above calculation where the same character set is used, the time to exhaust the entire keyspace increases substantially to around 24 days. These calculations confirm that a single GPU considerably outperforms a high-end CPU.

Standard motherboards are equipped with one or two CPU sockets, which limits the number of CPUs that can be installed. In comparison, a computer can run multiple GPUs using a Scalable Link Interface (SLI). The number of GPUs that can be run is dependant on the number of available PCI slots. Off-the-shelf motherboards normally have three or four PCI slots although dedicated GPU password cracking motherboards have been developed, some with as many as eight PCI slots.

In 2012, password cracking expert Jeremi Gosney unveiled one of the first desktop computers equipped with multiple GPUs built for the sole purpose of computing password hashes. Gosney demonstrated the power of his creation by cycling through 15.5 billion guesses a second. More than 90% of the 6.5 million leaked LinkedIn SHA-1 hashes were recovered in just five days (Goodin, 2012).

Figure 2-10 displays the number of hashes per second can be computed when using either a single GPU or using 8 GPUs in parallel against some of the most popular one-way cryptographic hash algorithms. This demonstrates not only the computational power of a single GPU, but the significant increase when multiple GPUs are combined in parallel.

<table>
<thead>
<tr>
<th>Hash Type</th>
<th>1 GPU</th>
<th>8 GPUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>MD4</td>
<td>1544M c/s</td>
<td>183232M c/s</td>
</tr>
<tr>
<td>MD5</td>
<td>7894M c/s</td>
<td>93800M c/s</td>
</tr>
<tr>
<td>SHA1</td>
<td>2495M c/s</td>
<td>29528M c/s</td>
</tr>
<tr>
<td>SHA256</td>
<td>1036M c/s</td>
<td>12328M c/s</td>
</tr>
<tr>
<td>SHA512</td>
<td>179M c/s</td>
<td>1952M c/s</td>
</tr>
</tbody>
</table>

Figure 2-10 - Official Hashcat Benchmarks
Due to these substantial performance increases, passwords that are currently considered secure by today’s security standards are in fact vulnerable against a GPU brute-force attack. The recommended password complexity requirements of many social networks and ecommerce web sites states that a password should consist of eight characters, using a mixture of uppercase, lowercase and numbers. The aforementioned calculation has demonstrated that under certain criteria, password hashes of this complexity could be recovered in a matter of hours.

An attacker with access to a computer capable of running GPU based attacks would see a significant reduction in attack time. In some circumstances attacks that would have taken months or years to complete can now be achieved in a considerably shorter time period. GPU password cracking is a genuine threat as the technology and hardware is both available and affordable. An attacker with competent computer skills can build and optimise a dedicated password cracking machine for around several thousand pounds.

Whilst GPU based password cracking is highly efficient at a brute-force based attack, the same cannot be said for a dictionary attack. For faster hash algorithms like MD5 or NTLM it takes longer to transfer a dictionary into the GPUs global memory and compute the hashes, than it would to simply compute using a CPU. oclHashcat is currently the only tool that has the ability to run as a GPU-based basic dictionary attack without a major reduction in efficiency.

In most hash algorithms password-length is set to a maximum of 55 characters. Presently, GPU-based password cracking can support the recovery of passwords up to 27 characters in length. Whilst this is arguably a weakness of GPU technology, analysis of the RockYou password set confirmed that less than 0.1% of 32 million users chose a to use password of 20 characters or more (Passscape, 2012).

GPU password cracking has resulted in the number hashes that can be generated per second increase significantly in the last five years. Whilst GPUs out perform CPUs considerably, the cost of hardware and complex configuration ensures that CPU based password cracking will still be a significant threat. CPU speeds increase at an exponential rate due to Moore’s law. Due to Moore’s law, password cracking speeds are expected to increase at the near exponential rate.
2.4 Moore’s Law

Moore’s law is a computing principle that states the number of transistors on central processing units will double in size every two years (Moore, 1998). These exponential increases have lead to increased processor speeds resulting in additional computational power. Moore’s law is credited to Gordon Moore who proposed the trend in 1965. From 1971 until 2011 microprocessor transistor counts have accurately followed his theoretical trend. The graph in figure 2-11 demonstrates an almost exponential growth between time and transistor count.

![Figure 2-11 - Exponential Growth of Transistor Count 1971-2011](image)

There is a direct correlation between the number of transistors per chip and the number of password hashes that can be generated per second. As CPUs have increased in computational power over time, the numbers of hashes that can be generated per second has also increased at a near exponential rate.

In 1995, historical benchmarks revealed that John the Ripper was capable of generating approximately 12,500 MD5 hashes per second using a Pentium Pro processor. In 2014, a computer equipped with an Intel Core i3 processor is currently capable of generating approximately 6.5 million hashes per second.
Consider the total number of combinations possible for a password consisting of lowercase characters or numbers, and no more than six characters in length. The total number of password combinations is a little over 2.3 billion. In 1995, it would have taken around 51 hours to exhaust the entire keyspace, guaranteeing a successful hash recovery. In contrast, the same attack in 2014 takes just six minutes.

This is unarguably a substantial reduction in time taken to recover a hash. Passwords that would have been considered secure in 1995 are considered highly vulnerable in 2014 due to these increases. This trend is set to continue.

Moore’s law is expected to continue until 2020 although transistor count is expected to double every three years rather than every two as per the original prediction. During this time, CPU speeds will undoubtedly increase and with it, the number of hashes that can be generated per second will rise. Due to these increases, eventually passwords that are considered secure today will eventually become vulnerable. Password complexity requirements have to increase, or continue to allow the use of vulnerable passwords.

Whilst the exponential growth proposed by Moore’s law is expected to eventually curve, GPU parallelism and resulting speeds are currently doubling every year. Although there is currently no formal roadmap for GPU development, between 2005 and 2010, the number of gigaflops per second increased from 250 to 1000, in the same period, the numbers for CPU increased from 40 to 200 (Srinivasan, 2010). GPUs can currently recover password hashes that were thought to be secure. Exponential growth in the next 10 years will have a serious effect on password cracking as previously unfeasible attacks will become computationally efficient.

There is currently no industry-standard password policy. Many recommended polices vary greatly between differing regulatory bodies. The unofficial, but generally followed recommendation are that passwords should contain a minimum eight characters, whilst consisting of lowercase, uppercase and numbers (Shay & Komanduri, 2010). Whilst it is impossible to accurately predict the computational speed increases that will occur over the next 20 years, history has demonstrated that as the number of hashes per second increases, password complexity undoubtedly has to increase to mitigate the risk of passwords becoming vulnerable.

There will come a point in time where password complexity requirements become counter-productive. Whilst the requirements may ensure passwords are theoretically secure from a cryptographical standpoint, they will more than likely suffer from bad password management. Wiedenbeck et al. stated that a good password must be “easy to remember” and “hard to guess”. Currently, satisfying these two conflicting requirements is difficult and where technology will undoubtedly continue to increase, satisfying both conditions will become even more challenging (Wiedenbeck, Waters, Birget, Brodskiy, & Memon, 2005).
2.5 Can Passwords Be Secure?

This chapter has discussed the cryptographic weaknesses of text-based passwords and how technological development will only increase their vulnerability. Passwords can be weak and vulnerable to attack, similarly, passwords can be secure and in some instances password hashes can be theoretically impossible to recover.

2.5.1 Secure Businesses

The most popular cryptographic hash functions used in modern password storage are MD5 and SHA-1. There are many other hash algorithms available, a number of which are deemed to be insecure and not recommend for public use. These algorithms are used due to their speed and low resource usage. Due to theoretical collisions both MD5 and SHA-1 are actually considered insecure even though cracking the cryptographic cipher would take an enormous amount of time. The SHA-3 algorithm is seen as the long-term successor for both MD5 and SHA-1, although adoption has been limited.

Many commercial businesses and software vendors still use either MD5 or SHA-1 regardless of the known weaknesses. These algorithms are favoured not just because they are quick, but security is often an afterthought during software development.

Password Hashing Competition (PHC) is an expert panel consisting of security professionals and expert cryptanalysts. In 2013 they announced a ‘competition’ calling for the submission of new password hashing schemes with the intent to identify a new scheme to improve on current hashing technology (PHC, 2013). Submissions closed in March 2014 with the winning hashing scheme scheduled for announcement in 2015.

Competitions of this nature emphasise a critical flaw of the development of hashing algorithms and to an extent, commercial computer software. The Internet is decentralised and has no governing body, there is no overruling committee that states that a specific hash algorithm should be used, or is secure. Businesses are reliant on industry experts for guidance, and even with their knowledge, there is no obligation to follow their advice. Industry specific regulatory bodies typically define security requirements and set the penalties for non-compliance. This is often not enough of a deterrent to combat insecure practices.

A number of tools, libraries and processes are available to ensure passwords can be securely implemented and stored. Businesses and their developers should ensure they are used, and used to their full potential. To mitigate risk, businesses should use a strong hash algorithm, encrypt any stored passwords and ensure salt is applied during the hash generation.
2.5.2 Secure Users
A password of at least ten characters in length that is comprised of uppercase, lowercase and numbers and is not made up of words found in the dictionary will ensure a password that will be extremely difficult to crack. Users should be encouraged to take a phrase and use the first letter of each word to make up a completely random sequence of characters, appending uppercase and numbers where appropriate. Consider the example phrase below.

Secure Passwords Are The Best Protection Against Password Attacks

Taking the first letter of each word, capitalising the first letter and adding a numeric character to the end would create the password: ‘Spatbpapa1’

This seemingly random assortment of characters would be significantly harder to crack than a password that appears in a dictionary, such as ‘Password99’ even though both are considered the same complexity by being ten characters long and consisting of mixed alphabetical characters and numbers.

The total possible number of password combinations would be over 839 quadrillion. The estimated time for a desktop computer to recover the hash of ‘Spatbpapa1’ is approximately 24,426,826 hours, which equates to around 27,656 years. This time includes a reduction in the keyspace based on the law of averages. The actual time to recover the hash could be longer. When used correctly, passwords are secure from the attacks discussed throughout this chapter.

Unfortunately, while a password of this length and complexity can be considered secure. Human cognitive limitations will result in users struggling to use and remember passwords of this criterion. No amount of technology will ever be able to eliminate the risk posed by human nature.
3 Social Weaknesses

3.1 Introduction
Chapter two evaluated the weaknesses of text-based passwords from a cryptographic standpoint. This evaluation confirmed the most vulnerable passwords are those that are short in length, comprised of a limited character set and are lexical. In comparison, stronger passwords are more complex. Complex passwords use a range of available characters and can be made up of a phrase or sometimes are even a completely random set of characters.

Businesses invest heavily in security software and hardware and whilst these tools offer a greater level of protection against attacks at a system or network level, the end-users are often protected by nothing more than security guidelines or policies, often with a lack of enforcement. This neglect creates risk and leaves end-users vulnerable. The human aspect of security is often dubbed as the ‘weakest link’ in the security chain (Mitnick, 2002).

This chapter will evaluate this human aspect and consider the weaknesses of text-based passwords from a social standpoint by investigating the popular attack method social engineering. This chapter will also establish how cognitive limitations affect password management, investigate attitudes to password re-use and sharing, and determine whether a user's awareness of threats and the subsequent risk has an effect on their own password management.

3.2 Social Engineering
Social engineering is one of the biggest threats to businesses in 2014 with end-users the frequent target. The basic premise of social engineering is to convince users to divulge confidential information that they would not usually willingly reveal (Thornborough, 2004). One of the biggest targets of social engineering is usernames and passwords (Pfleeger & Pfleeger, 2006).

There are several different social engineering attack methods, many of which have developed over time. Shoulder surfing is an attack that simply involves an attacker watching a user as they type in their password or PIN number. Shoulder surfing emphasises a key weakness of passwords as once an attacker has observed submission, they are free to leverage it to their gain without any further action required. The danger of this attack is that anybody can be a potential attacker as there is no specialist technical knowledge or tools required (Orgill, Romeny, Bailey, & Orgill, 2006).

Dumpster diving is another attack method that involves an attacker scouring through industrial paper-waste attempting to discover confidential information. Due to increased waste-reduction policies, the threat level of this attack has reduced over time, although similar to shoulder surfing, the danger lies in its low-tech approach.
Social engineering can be conducted face-to-face, over the phone and more recently online. An attack method known as phishing has became prevalent over the last five years. Phishing is an attack method that attempts to extract confidential information, usually passwords or financial information by impersonating a legitimate party. During the early growth of the Internet, phishing scams were traditionally conducted via email, where attackers would promise their target large amounts of money in return for their bank account details.

As technology progressed, attack methods evolved. Modern phishing attacks often claim to be from a bank or other ecommerce business. Emails will prompt users for confirmation of their username and password or other confidential information. Users are then re-directed to a legitimate looking webpage where they are duped into handing over this information (Jakobsson, 2007). The Gartner Group in conjunction with Cambridge University estimated the total annual losses from phishing attacks are $2 billion per year (Moore & Clayton, 2007).

Another popular phishing attack is a drive-by-download. Users receive a legitimate looking email containing a webpage URL (Uniform Resource Locator) which redirects them to a fake website. By exploiting either Flash or Java, the website automatically downloads and installs malicious code on to their computer. Dependant on the strain of malware, the compromised machine may inadvertently end up as part of a wider botnet and in many cases, has keystroke logging software installed with the sole intention of stealing passwords (Zarras, Kapraveols, Stringhini, Holz, Kruegel, & Vigna, 2014). A similar attack method exists named link-hijacking where the victim is directed to a malicious location, not the location they originally intended.

Another popular attack method aptly named ‘deceptive download’ was named as the top threat to businesses in a report published by Microsoft (Microsoft, 2014). Popup adverts inform users they are missing a necessary plugin such as Flash Player or their machine is infected with viruses, malware or spyware. Users are promised the software will help eliminate these threats and speed up their machine, when the reality is the software they download is often malicious. These attacks are targeted to end-users with limited technological expertise.

The last decade has seen a significant increase in the number of services going online. This increase has seen an influx of new users, and with that, new passwords. These new users are prime targets that attackers will attempt to exploit, with the end goal of deceiving them into revealing confidential information (Cao, Yang, Yu, & Palow, 2014).

A user may have chosen a secure password, but if they are tricked into revealing it, it can be considered no more secure than if they had selected a weak password.
3.3 Cognitive Abilities

An end-user’s cognitive ability to ensure secure password use is often called into question. In the largest password habit study conducted, Florencio & Herley concluded that users have an average of 25 accounts, but only have six or seven passwords that are used between them (Florencio & Herley, 2007).

Florencio & Herley’s research appeared to align with the longstanding psychological principle known as Miller’s Law. A paper published in 1956 by leading psychologist George Miller established that the number of objects a human can hold in working memory is 7 ± 2 (Miller, 1956). They concluded that Miller’s law directly affects the number of passwords a user is able to mentally remember.

In a study conducted by Adams and Sasse, participants were interviewed about their password habits. 50% of participants revealed that they wrote down their passwords. Many users justified their actions as a coping mechanism as they were unable to competently remember passwords for all their different accounts (Adams and Sasse 1999). These users also admitted to writing passwords down and sticking them to their monitors or to the inside of their desks.

Dhamija and Perrig confirmed this practice during their own study of password habits. Their study also revealed that the majority of users wrote their passwords down, despite users admitting they had secure password management awareness and understood the risks and potential consequences of their actions (Dhamija & Perrig, 2000).

Due to technological advances in password attacks, businesses enforced stricter password policies with the intent to mitigate the risk of weak password use. These policies will typically enforce a stricter password complexity, a limit on historical password reuse and force users to change their passwords at regular intervals (Weirich & Sasse, 2001). Research by Summers & Bosworth revealed these strategies often have a detrimental affects on a user’s cognitive load resulting in a negative impact on security. Users admitted to devising their own coping methods that prioritised their own convenience over security, which led to risky practices (Summers & Bosworth, 2004).

Conklin et al. established that users develop memory aids when confronted by complex passwords requirements. During observation, these memory aids consisted of either writing passwords down or reverting to bad password management. Conklin et al. concluded that even with security awareness and password management training, basic cognitive limitations make it impossible for users to simultaneously accommodate complex security recommendations for multiple systems. Their study further concluded that users could be expected to remember no more than four passwords that would satisfy complex password requirements (Conklin, Dietrich, & Walz, 2004).
Earlier research by Gehringer aligned with the findings of Conklin et al. Gehringer determined that an average user’s cognitive ability significantly reduces the chance that they would choose a password that would be considered secure. Instead users would opt for weaker passwords that were easier to remember (Gehringer, 2002).

A study by Petrie analysed the password choices of 1200 users. Petrie identified that password choices can be categorised into one of four categories, these are:

- **46% - ‘Family Orientated’** where passwords were comprised of names, nicknames, dates of birth and telephone numbers.

- **33% - ‘Fans’** where passwords were comprised of the names of movie stars, athletes, celebrities and fictional characters.

- **11% - ‘Fantasists’** where passwords were comprised of sexual references or topics.

- **10% - ‘Cryptic’** where passwords were comprised of random strings of letters, numbers and symbols.

Psychology expert Andrews analysed Petrie’s research and concluded that passwords are ‘inadvertently revealing’. Andrews declared that users naturally choose words or phrases in the forefront of their mind when they create a password, which typically occurs when accessing a system for the first time or when a user is trying gain access as quick as possible (Andrews, 2012).

Andrews concluded that during password creation users unconsciously choose something that is of particular emotional significance. Due to these cognitive weaknesses, users often inadvertently choose passwords that are vulnerable to dictionary-based attacks.

Conklin et al. concluded that whilst the concept of a user ID to password relationship is both efficient and cost effective, it was developed in a time of fewer systems and fewer passwords. As technology has advanced, the number of accounts has increased and with it, the number of passwords. There has been limited success to help mitigate the threat caused by these increases (Conklin, Dietrich, & Walz, 2004).
3.4 **Password Reuse & Sharing**

A study by Gaw & Felten established that the number of passwords that are reused increases as users acquire more accounts. Their study confirmed that whilst the number of accounts increases with a user’s age, the number of passwords in a user’s ‘password pool’ does not (Gaw & Felten, 2006).

Gaw & Felten also concluded that users typically have between three and five core passwords and will vary the complexity of these passwords depending on the account it is to be used for. In their study, participants admitted using a weaker password for social media and junk accounts and a complex variation of the same password for banking and financial accounts.

Notoatmodjo and Thomborson explored the idea that users are aware of what makes a password secure but choose to use stronger passwords only on banking and ecommerce accounts. Their study revealed that there is no single motive for password re-use, where 35% of participants revealed they re-use passwords simply because ‘it’s easy to remember’ with a further 13% admitting that the account was shared with other people (Notoatmodjo & Thomborson, 2009).

Notoatmodjo and Thomborson did determine that users mentally group their accounts and make stronger password choices for those they considered the most important. Many participants revealed they chose a secure password for a particular account because there was some incentive to do so. Participants acknowledged using more secure passwords for financial accounts where they perceived a risk that their money could be stolen and not retrievable.

Haque et al. conducted an experiment where participants were asked to create passwords for ‘lower level’ accounts such as the weather, or news and ‘higher level’ passwords for accounts such as banking. The ‘lower level’ passwords formed part of a word list that was loaded into John the Ripper. One third of the participant’s passwords created for the ‘higher level’ accounts were recovered by using passwords that were created for ‘lower level’ accounts. This study suggests that using simple variations of weaker passwords to create more complex passwords does not offer an adequate level of security (Haque, Wright, & Scielzo, 2013).

In a study by Riley, over 50% of participants admitted to using the same password for every account they had, with an additional 33% using simple variations of the same password across all of their accounts (Riley, 2006). This supported the work of Summers & Bosworth that determined that users are more concerned about convenience than security and password re-use, a frequent coping mechanism.
In Tam et al., 42% of users admitted to sharing passwords with friends and family members, despite knowing the security risks involved (Tam, Glassman, & Vandenwauver, 2010). One observation from their study was that whilst users understood the potential risks from password sharing, they saw no differentiation in sharing passwords for their accounts to sharing a home Wi-Fi password with other family members.

Password sharing is prevalent in homes and in many commercial businesses. Tam et al. argued that this practice promotes the image that password sharing is acceptable in certain situations. Popular services such as Netflix and Hulu do not explicitly forbid password sharing and openly promote that the same account can be used concurrently across multiple devices (Netflix, 2014). When large, household services permit account sharing, they are inadvertently promoting a culture where password sharing is seen as acceptable. Participants in a study conducted by Kaye confirmed this attitude towards password sharing by admitting they did not believe password sharing was a ‘deviant practice’ but could carefully be utilised in the right circumstance (Kaye, 2011).

### 3.5 User Awareness

Adams and Sasse discovered that many users knowledge of password security is poor. A large number of participants in their study did not realise the number of tools capable of launching password attacks, or how easy they were to obtain and use (Adams & Sasse, 1999). They concluded that a user’s motivation for security is low and many knowingly circumvent password policies for the sake of convenience.

During their study, Gaw and Felten discovered that participants visualised the most dangerous cyber security threat to be human attackers. Most participants were unable to differentiate between human attackers and the automated tools they potentially would possess. The majority of participants did not comprehend their seemingly secure, personalised passwords, such as names or phone numbers could be cracked with either brute-force or dictionary attacks in a matter of minutes (Gaw & Felten, 2006).

Riley conducted a survey to investigate password generation practices. Over 85% of participants composed passwords consisting of lowercase characters only. 55% indicated they used names of children, pets or other personal, identifiable information (Riley, 2006). Riley discovered the average length of a participant’s password was seven characters. Although password length has increased since the studies conducted by Morris and Riddle, as mentioned in chapter two, passwords of this length are especially vulnerable to brute-force attacks, many of which could be cracked in a matter of hours.
Riley’s study also revealed that a majority of participants had good awareness of password policies and knew they should not use words that are personal, in a dictionary, or of a certain length. Despite this awareness, users continually ignored guidance and chose weak, insecure passwords. When users were asked to justify this behaviour, the response was “easy to remember”. This is a change in trend from Adams and Sasse where for the first time, users who were aware of risks associated with weak password use, still chose to adopt bad password management practices.

Campbell et al. revealed that many users are unrealistically optimistic about Internet security, or simply don’t know what attackers are capable of (Campell, Greenauer, Maculso, & End, 2007). Campbell et al. also revealed that many participants had an “it won’t happen to me” attitude. This may explain the results of Riley’s study where users are in denial about the risks they potentially face.

Tam et al. users were asked to evaluate a password’s strength. 87% of users were aware of what made a secure password and 95% understood that a random series of digits and numbers in mixed case would provide the most secure password. Participants were asked to rate their current passwords on a scale of one to six, with one being easy, and six being the most complex, no users admitted their password was six on the scale. (Tam, Glassman, & Vandenwauver, 2010).

The most important revelation in Tam et al. was that although users understood what makes a password secure, they associated secure passwords with a loss of convenience and would be unwilling to give up this convenience. Tam et al. further suggest that education about the negative consequences of bad password management would do little to change their behaviours. Whilst this contradicts the findings in Gaw & Felten and Herley, it can be submitted that users will do whatever they want to do where password management is concerned.

Tam et al. confirmed the trend in that the number of users educated in secure password management has increased. Earlier studies suggested that users chose insecure passwords due to insufficient awareness. Nonetheless, more recent studies have shown that users that are aware of the risks of insecure password management use will still opt for convenience over security.
Butler & Butler proposed that a user’s password performance is based on three characteristics: Password related knowledge, capability to apply that knowledge and the motivation to behave in a secure manner (Butler & Butler, 2014). Figure 3-1 shows their Password Performance Model.

![Password Performance Model](image.png)

They concluded that the majority of participants in their study had sufficient password related knowledge and the ability to apply it. Many of the users lacked motivation to apply secure practices that led to bad password management. Butler and Butler’s finding were consistent with the previous research of Tam et al. Users have security awareness and understand what makes a password secure, but choose not to apply this knowledge to their own password use.

One theory for the lack of motivation for secure password use is an absence of a common password complexity standard. Gehringer concluded that for as long as there are many differing password standards and policies, a lack of common standards will have a detrimental affect on password management practices (Gehringer, 2002).

Inglesant & Sasse called for a radical overhaul of password policies with the intention to align with human cognitive ability and that password strength and change frequency is the wrong focus. Their study also confirmed that users are aware of the security implications of password policies, but due to their inflexibility, they are forced into bad password management habits (Inglesant & Sasse, 2010).
3.6 Password Leaks

There have been a number of studies conducted about users passwords habits. Early studies by Morris and Riddle gave the first insight into password habits, and recent studies by Brown & Sasse have concluded that whilst passwords have become more secure over time, they are still not secure enough.

A large scale study concluded by Florencio and Herley was the first of its kind, capturing and analysing the password use and re-use habits of over half a million users. Whilst this study concluded that users chose insecure passwords and re-used them across multiple systems, very little was known about the actual passwords themselves (Florencio & Herley, 2007).

The RockYou password leak in 2009 was a landmark moment in password security. A data breach resulted in the leaking of 32 million user accounts and passwords. RockYou had failed to take even basic security precautions. The data was unencrypted and 14 million passwords were stored in plaintext. Whilst this was a catastrophe for the company involved, destroying their reputation, the password cracking landscape changed instantaneously. For the first time, attackers had access to real users password habits. It allowed attackers attempting to recover hashed passwords to almost instantaneously crack the weakest passwords. This would free more time and resource to devote to cracking the more secure hashes.

The RockYou password list is considered so comprehensive that it is the bundled wordlist used by both John the Ripper and Hashcat, the two most popular password cracking tools. The basis of many dictionary attacks and hybrid dictionary attacks are based on RockYou’s content and its importance in password cracking is unparalleled.

Since the RockYou password leak, there have been a number of other high-profile data breaches in several other large organisations including Yahoo, LinkedIn and Adobe. These leaks are not only damaging to their respective owners, but analysis of recovered hashes by security professionals has confirmed one thing: users still use insecure passwords and these leaks are indisputable evidence.
SplashData conducted analysis of a number of the leaked password sets and produced a report named "The Worst Passwords of 2013" (SplashData, 2014). Figure 3-2 displays the top 20 passwords that appeared most frequently across every leaked password set, including those from the organisations mentioned above.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Password</th>
<th>Rank</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>123456</td>
<td>11</td>
<td>123123</td>
</tr>
<tr>
<td>2</td>
<td>password</td>
<td>12</td>
<td>admin</td>
</tr>
<tr>
<td>3</td>
<td>12345678</td>
<td>13</td>
<td>1234567890</td>
</tr>
<tr>
<td>4</td>
<td>qwer123</td>
<td>14</td>
<td>letmein</td>
</tr>
<tr>
<td>5</td>
<td>abc123</td>
<td>15</td>
<td>photoshop</td>
</tr>
<tr>
<td>6</td>
<td>123456789</td>
<td>16</td>
<td>1234</td>
</tr>
<tr>
<td>7</td>
<td>111111</td>
<td>17</td>
<td>monkey</td>
</tr>
<tr>
<td>8</td>
<td>1234567</td>
<td>18</td>
<td>shadow</td>
</tr>
<tr>
<td>9</td>
<td>iloveyou</td>
<td>19</td>
<td>sunshine</td>
</tr>
<tr>
<td>10</td>
<td>adobe123</td>
<td>20</td>
<td>12345</td>
</tr>
</tbody>
</table>

Figure 3-2 - Splash Data's Top 20 Worst Passwords 2013

Ten of these passwords are recoverable in just over ten minutes using a standard laptop and HashCat. Figure 3-3 shows a brute-force attack, trying every possible combination using a character set of lowercase letters only and passwords between one and seven characters in length.

All hashes have been recovered

Input Mode: Mask (?1?1?1?1?1?1) [6]
Index......: 0/1 (segment), 2176782336 (words), 0 (bytes)
Recovered.: 10/10 hashes, 1/1 salts
Speed/sec.: - plains, 3.25M words
Progress.: 2123048756/2176782336 (97.53%)
Running....: 00:00:10:52
Estimated.: 00:00:00:16

Figure 3-3 - Brute Force Attack Using Hashcat

The results of the brute-force attack were copied into a file called top_20_found.txt. Figure 3-4 shows the contents of the output file, revealing the original hashes and the corresponding plaintext passwords that were successfully recovered.

root@kali:/usr/bin# cat top_20_found.txt
81dc9bdb52d04dc20036dd13136ed55:1234
827ccb0eea8a706c4c116bb95b3b0f9:12345
21232f297a57a57a743949e04a01fc3:admin
95af79218956d72c92a549dd5ae3011:111111
10ad3949ba59abbe56e957f29f883e:123456
136763edaa9d9bd2a9516230e9044d85:monkey
d5578edf8458ce06fbb6a58c5ca4:qwer123
19a16c4288cb685f286833678922e03:abc123
4297f44eb8955235245b2497399de9:12332
3bf114a986b87ed28fcb5884f8f8:shadow

Figure 3-4 - Recovered Passwords From Hashcat
Until the analysis of recent password leaks, there was little factual information about users password habits. Many studies had speculated about the social weaknesses of passwords where early studies suggested that users were simply unaware of the risks, good password management principles and the importance of security policies.

Recent studies contradict these suggestions and have proven that regardless of a user’s security awareness or their understanding of the threats and risks of poor password management, they freely choose to ignore them and instead opt for convenience over security. The leaked password data sets are consistent with these studies and confirm that many users choose weak and insecure passwords, regardless of their security prowess.

However, the research presented also implies that regardless of their awareness or motivation, users cannot be expected to be able to remember secure passwords because of cognitive limitations, turning to coping mechanisms when faced with complex password requirements.
4 Alternative Authentication Methods

Chapter Two explored the cryptographic weaknesses of text-based passwords and determined that if used correctly, passwords can be secure. Chapter Three investigated the social weaknesses of text-based password and concluded that even when users have good security awareness and an understanding of the risks of insecure password practices, their cognitive limitations have a detrimental effect on their password management.

In this chapter, the strengths and weaknesses of two alternative authentication methods are explored. The intent is to conclude whether either method may be a suitable replacement for text-based passwords.
4.1 Biometrics

Biometrics refers to the automatic recognition of individuals based on behavioural and/or physiological characteristics. Biometrics enables the identification and authentication of a person based on who they are, rather than what they know, or by something they physically possess.

Biometric authentication utilises unique characteristics and enables users to identify themselves with nothing more than their own individualities. Jain defines four distinct requirements of a user’s characteristic that enables its suitability for biometric authentication (Jain A. K., 2005). These include:

• **Universality** – The characteristic can be found in every individual.

• **Distinctiveness** – The same characteristic of two people should be sufficiently distinguishable.

• **Permanence** – The characteristic remains unchanged over a long period of time.

• **Collectability** – The characteristic can be quantitatively measured and stored as data.

Aside from the suitability of a user’s characteristics, Jain also defines a number of other considerations that biometric authentication systems must satisfy in order to be effective. These include:

• **Performance** – A system must be capable of proving satisfactory speeds and accuracy. Consideration is required for operational and environmental factors that may affect performance.

• **Acceptability** – To what extent are people willing to accept the use of biometric authentication as an authentication standard and are they prepared to accept the increased use of their individual characteristics?

• **Circumvention** – How secure is the system and how easy is it to fool with fraudulent methods?

All modern biometric systems are required to satisfy the above characteristics to ensure their success. Whilst there a number of commercially available biometric systems implemented, widespread adoption of biometric authentication has been limited and no solution has been deemed to be completely secure.
There are a number of biometric authentication systems available that use a range of different characteristics. These include fingerprints, voice, iris, retina, hand geometry and facial shape. Wayman speculated in 2001 that due to simplicity, fingerprints would be the most widely used biometric characteristic in the future (Wayman, 2001). Wayman’s predictions proved to be correct. In 2014 there are fingerprint scanners in smart phones, at airports and in many businesses around the world, where the use of other biometric characteristics have seen limited adoption.

A biometric system is a simple pattern recognition system. A biometric system will acquire biometric data from an individual and store that data against a template. This template will be compared at authentication time. If the submitted data matches the template, authentication is granted. One commercially available fingerprint scanner uses a complex algorithm that calculates that values of nine key characteristics from user’s fingerprint; the final value is then stored as a template for comparison when authentication is required (Crossmatch, 2014). This operation is typically standard in many biometric systems.

Biometric devices have two functions: identification and verification. During identification, a user required to submit their biometric data multiple times enabling the device to learn their characteristics. These multiple scans will form a profile of the user and ensure that if an imperfect scan is captured, the correct decision regarding the user is made.

Biometric authentication software often uses a scoring system that compensates for external factors that may result in scanning abnormalities. External factors include: humidity, cuts, skin conditions or a user’s interaction with the scanner. The scoring system must be correctly implemented to ensure that the correct authentication decision is taken each time data is submitted.

Whilst many biometric systems use physical characteristics to identify users, there are behavioural characteristics that can also be utilised. Manduanjo proposed a biometric authentication system that would eliminate password sharing by through the implementation of keystroke dynamics (Mandujano, 2004). Keystroke latencies and the time spent at login time can be used to build a profile about the user. Thus an attacker would have to accomplish two things in order to gain access to the system.
Biometric profiling does not require costly equipment or additional software, unlike many other biometric authentication solutions. However, precision rates are weaker when compared to other biometric methods. Longer passwords are usually required as they yield a better learning pattern. Biometric profiling limits the effect of password sharing and stealing by including an additional variable to the biometric equation. Mandunajo likened biometric profiling to adding salt to passwords. Despite its relatively simplistic approach, biometric profiling has seen limited adoption.

### 4.1.1 Strengths

The use of biometric characteristics has the potential to solve the human cognitive limitations that researchers believe prevent a user from memorising a secure password. The use of a single fingerprint would eliminate the requirement for a user to remember multiple, complex passwords.

Biometrics would eliminate password sharing and reuse across multiple accounts. It would be impossible for a user to share a biometric characteristic in the same way they could share text-based password. Whilst the same characteristic could be used across multiple systems, reuse of this nature would not present the same risks as reusing passwords across multiple systems.

The threat of the cryptographic attacks discussed in Chapter Two would be significantly reduced. Attack methods like brute-force and dictionary attack would be ineffective and password cracking tools obsolete. Attackers would be required to find new ways to attack biometric devices (Sujithra & Padmavathi, 2012).

Users have previously justified bad password management by declaring that complex password policies forced them into devising insecure coping mechanisms. As biometric characteristics never change, there would no longer be the requirement for a user to change their password at specific intervals (Sukhai, 2004).

Biometrics offers a greater protection to users that fall victim of one of the many phishing attacks discussed in Chapter Three. Any keylogging software installed would be ineffective, as there would be no text-based passwords for the software to steal. Other social engineering threats would be mitigated as a user could not be tricked into revealing biometric data and whilst spoofing a biometric characteristic is possible, it is extremely difficult to do consistently and successfully (Jain A. K., 2007).

The use of biometrics in authentication systems promotes a high degree of confidence to both users and stakeholders. Systems that use biometrics for authentication are considered the most secure (Gao, 2011). Biometric systems by default have increased user accountability, something advantageous to businesses. Account sharing would be extremely limited and businesses would have a greater certainty of the actions of their users.
Biometric authentication offers greater convenience for end-users. Prior research confirmed that users opt for convenience over secure practices. Making the authentication procedure as convenient as possible would undoubtedly encourage users to adhere to stricter security practices. This would satisfy the password performance principle proposed by Butler and Butler. An increase in convenience will typically increase a user’s motivation to behave in a secure manner (Butler & Butler, 2014).

Many commercially available biometric systems are incredibly quick, with some devices taking just one or two seconds to scan and authenticate a user (3M, 2014). In comparison, a user typing complex passwords on a keyboard could take up to ten seconds (Villani, 2006).

Unlike passwords, a biometric characteristic cannot be lost, nor can they be forgotten. This removes the requirement for password-reset processes. These secondary processes are often equally as insecure as text-based password use. In the event that a data breach occurred and biometric data of users were compromised, it is very difficult to reuse the data in the same way that attackers have used password leaks to advance their password attacks (Wang & Rane, 2012).

### 4.1.2 Weaknesses

The use of biometric data raises a number of privacy and security concerns. Prabhakar et al. concluded that many people are uneasy with the idea that their biometric data could be stored by many businesses across multiple databases. Participants were particularly apprehensive about the storing of fingerprints due to its association with criminal activity and prisons (Prabhakar, Pankanti, & Jain, 2003).

Research by Mordini was consistent with Prabhakar et al. in concluding that users find biometrics devices intrusive and personally invasive (Mordini, 2008). These negative connotations may change as Apple and Samsung are beginning to integrate fingerprint authentication into their flagship smartphones where the biometric data is stored on the device, rather than on the business’s database.

Biometric devices are costly to buy, implement and maintain. There are a small number of expensive laptops, tablets and smartphones that come equipped with fingerprint scanners, the majority of devices do not. Fingerprint readers are now the cheapest devices to buy, with facial recognition systems one of the most expensive. Biometric devices often have a limited shelf life. Fingerprint scanners that used frequently should be replaced every 12 months (Bhargav-Spantzel, Squicciarini, & Bertino, 2006).

A biometric authentication device typically acts as an intermediary between the user and the system they intend to authenticate with. All mechanical devices have the risk of temporary malfunction or permanent defect. Biometric authentication systems often require an alternative or backup access method or processes.
A user may have to call a helpdesk to gain access, or revert to using a password. These necessary backup processes typically reduce the level of security offered and may become the new target for attackers. Where no alternative process exists, there may be instances where a legitimate user is unable to authenticate until the problem with device is resolved.

All biometric devices suffer from ‘false rejection’. A false rejection is where a user is denied access due to the device incorrectly rejecting their biometric data, even though they are a legitimate party. Fingerprint readers have some of the highest false rejection rates, with some as high as 9% for an optical reader and as low as 4.5% with a chip reader. In comparison, facial recognition has a false rejection rate of approximately 2.5%. The lower the false rejection rate, the more expensive the biometric device (Clarke & Furnell, 2008). The biggest risk of a high false rejection rate device is that users and stakeholders will rapidly lose confidence in the system due their frustration and inability to authenticate in a timely fashion.

Today we live in a society that is considered inclusive and liberal. There are a number of high-profile laws that protect individuals with disabilities in the work place and any form of discrimination in businesses is illegal. Biometric devices will be accessible to a large proportion of a business’s workforce, although there may be occasions where using a fingerprint reader may not be appropriate for a user with amputated limbs, or using a retina scan for a user who is visually impaired (Matyas, 2003). These users would either have to use an alternative biometric method or use a secondary authentication method, which is likely to be a text-based password.

Biometric characteristics do not change. As previously considered, a user would no longer be required to remember a number of complex passwords or change them at frequent intervals. Whilst this is an inherent strength of biometrics, it has been argued that this is also a major risk. If a user’s data is lost or stolen, a text-based password can be changed easily. It is nearly impossible for a user to change their biometric characteristics (Uludag, 2004).

Research by Gaw & Felten established that users are more security conscious of specific accounts, such as online banking. The same attitude will likely apply to the use of biometrics for authentication. Users will be receptive to adopt biometric authentication for accounts they deem important, but are unlikely to adopt it across all of their accounts (Ahmad & Hariri, 2012).

Many biometric systems require a database to store the template data for each user registered for that system. This database will typically be housed in a cloud-computing environment, requiring a constant Internet connection for operation. There may be instances where a constant connection is not available and a legitimate user will be unable to access the system. Users will either have to resort to backup methods or remain unable to authenticate.
The biggest disadvantage of biometrics is a lack of standardisation. This is a
similar challenge faced by text-based passwords. Due to the Internet’s
decentralised structure, there is no overall governing body. There are multiple
biometric vendors, devices and standards. Businesses are free to choose
whichever hardware and software best fits their business needs and budget.
The biggest concern about the usage of biometric data is the lack of security
standards and potential privacy implications (Ratha, 2010).

Due to the absence of standardisation, biometric technology is frequently
patented and is very costly to license and implement (Froomkin & Weinburg,
2012). Text-based passwords are still widely used due to their relatively low
cost and efficiency.

Similar to the public’s perception of computer hackers, there are a number of
false assumptions about biometrics used in authentication. Some users
believe their fingers are at risk of being severed and their fingerprints liable to
being lifted from coffee cups. Whilst these types of actions might be found in
the latest science fiction film or James Bond movie they do not translate into
real-world activity. Research by Sukhai confirmed that reusing biometrics from
dead people is all but impos-
sible (Sukhai, 2004).

Any authentication technology is susceptible to attack. The most common
attacks against biometric authentication systems are replay attacks and
spoofing. A replay attack occurs when a user’s biometric data is required to
pass over untrusted network, such as the Internet. The data could be
intercepted during transmission and stolen if it is not encrypted. An attacker
could interact with the target system, sending the stolen data to maliciously
authenticate (Czajka, 2008). Spoofing attacks exist, but successful attacks
are difficult to achieve consistently. Security researchers have shown that it is
possible to fool biometric devices with prosthetics (Chaos Computer Club,
2013). The TABULA RASA project is a European Commission joint venture
that is aiming to address the issues of spoofing and that falsification of
biometric characteristics. TABULA RASA will develop a standard framework
for attack prevention (TABULA RASA, 2014).

4.1.3 Recent Advances
There has been a major advancement in biometric authentication in 2014 that
will increase adoption and awareness. In 2013, Apple introduced a built-in
fingerprint reader named TouchID into their flagship smartphone, the iPhone
(Apple, 2014).

Until October 2014, TouchID was a closed platform and its functionality was
limited to users unlocking their mobile devices without a pass code or
verifying purchases from Apple’s ‘App Store’. As part of the iOS8 software
update, Apple announced it was allowing third party application developers to
integrate and use their TouchID for authentication purposes. Businesses
including PayPal, eBay and Facebook are among the first to announce their
app’s compatibility with TouchID, replacing the requirement for a user to type
in a password to access their account.
By integrating a fingerprint reader into a mobile phone, Apple removed the necessity for businesses to invest in biometric hardware. People are heavily reliant on their mobile phones and many have their device upon their person at all times. With over 250 million iPhones equipped with TouchID, Apple is one of the first major companies to incorporate biometric authentication across their entire mobile platform. Users are more trusting of large corporations and are more willing to adopt their technology and practices (Wier, Douglas, Richardson, & Jack, 2010).

Whilst Apple continues to drive adoption of biometrics in the mobile sector, other businesses across the world continue to invest heavily in biometric technology. In 1990, businesses spent an estimated $6.6 million, this increased substantially to $200 million in 2000. Industry experts expect this rapid growth to continue and estimate the global biometrics market in 2017 to be $16.5 billion (Safran Group, 2010).
4.2 Multi-factor Authentication

Multi-factor authentication, also known as two-factor authentication is a method of identification control that requires the user to satisfy two of three possible authentication modalities. These include:

- **Knowledge-based** – Something the user knows: A piece of information that is theoretically known only to the user, such as a password.

- **Object-based** – Something the user has: A device that is unique to the user, such as a token, Smart card or ATM card.

- **Biometric-based** – Something the user is: A unique biometric characteristic of the user that satisfies specific requirements for use in biometric authentication.

Multi-factor authentication is not a new authentication concept. Its first known use in computer authentication combined a regular text-based password and a one-time passcode that was generated on the user’s terminal (Lamport, 1981).

Any customer of a bank that uses ATM card uses multi-factor authentication as a way of authorising purchases or withdrawing money at an ATM machine. The customer inserts their ATM card into the machine and when prompted, they enter their PIN number. If the machine authenticates the customer, they are able to withdraw their money. Figure 4-2 shows the multi-factor security concept applied to a customer using an ATM machine.

This simple concept protects customers if their card is lost or stolen. If another person tried to withdraw money or authorise a payment, they are required to provide the PIN number. ATM machines will typically retain the card if an incorrect PIN is entered more than a set number of times. This eliminates the risk of a brute-force attack against the PIN number. The same concept can be used for authenticating users on computer systems.

The first remote authentication scheme that adopted the multi-factor model was first introduced in 2000 using Smart Cards (Hwang & Li, 2000). Smart Cards were issued to users who were required to insert them into special Smart Card enabled keyboards as they attempting to authenticate. This ensured that if a user’s password were compromised, an attacker would not be able to log on without possessing the Smart Card.
Despite their initial successes, only large businesses and government departments now use Smart Cards. Their limited adoption is often credited to the cost of their implementation and maintenance.

Banks and other financial institutions that offer online banking as a service to their customers, have incorporated multi-factor authentication into their services to enhance security. Many banks now require a user to submit multiple passwords to access their account and complete further steps to execute transactions or other account changes. Banks have issued their customers with card reader machines that generate a one-time code that can be entered into the web browser. These card readers require the user to be in possession of the card and the PIN number before the code can be generated. To ensure a successful attack, an attacker would be required to possess a card, a PIN number and the password details of the online account. This implementation is considered a multi-level, multi-factor implementation and has seen a reduction in fraudulent activity, particularly in online banking (Mohammed, 2013).

The current most popular method of multi-factor authentication is using a security token. These security tokens generate a one-time use code that the user is required to submit along their password. Security tokens are leveraged by businesses attempting to protect their data and are typically used to enable remote accessing via a Virtual Private Network (VPN). Whilst security tokens are most commonly used in commercial business, there are other services that have utilised their use.

Blizzard Entertainment was the first companies to offer multi-factor authentication to access their online gaming platform: World of Warcraft. This was in response to a number of users who had complained their accounts were compromised, resulting in their characters, items and in-game currency being stolen. In 2008, Blizzard sold game players a key-chain sized device that generated a one-time code to be used when players accessed their accounts - a preventative measure against password theft. Since its inception, Blizzard continues to offer a physical device for purchase, or players can download an application to their smart phones (Battle.net, 2014). This marked the first time that multi-factor authentication was used in the gaming industry as its use was typically confined to the finance and corporate sectors.

Google offer a range of multi-factor authentication options for users of their Google Mail accounts. These options include: SMS (Short Messaging Service), voice call, a mobile application. In August 2014, Google became the first company to launch their own USB security key (Google, 2014).

Due to the rapid increase of mobile phone ownership in the last decade, multi-factor authentication by mobile device is becoming increasingly popular (Lami, Kuseler, & Al-Assam, 2010). A mobile phone replaces the need for a user to possess a card reader or security token at all times. SMS messages can turn any mobile device into a multi-factor enabled device.
Using mobile devices as authentication devices is an attractive option for businesses, as many employees would have access to a mobile phone. SMS costs are cheap in comparison to issuing security tokens or Smart Cards (SecurEnvoy, 2013). Many multi-factor authentication solutions use a combination of knowledge-based and object-based modalities. Use of biometrics is uncommon, although this is likely to change due to the emergence of mobile phones with built-in fingerprint readers.

4.2.1 Strengths
Multi-factor authentication is a proven security concept and is already used by a large number of organisations and their users. For many years, banks and other financial institutions have leveraged the concept of multi-factor authentication through their card and PIN number system.

Multi-factor authentication has reduced identity theft in online banking, saving businesses revenue and increasing customer confidence in its use. Attackers who may be in possession of a target’s password will no longer be able to gain access to a target system without a secondary piece of information. This reduces the risk and threats of the cryptographic attacks discussed in Chapter Two. Attackers may still use these attack methods to recover a text-based password of the user, but would need to leverage other attack methods to obtain the additional object-based information (Buscan, 2010).

There are limited privacy or security concerns with an object-based multi-factor system in comparison to authentication systems that use biometric characteristics. A user would select a password known only to them and possess a security token or Smart Card when required. Authentication systems that use a combination of knowledge-based and biometric-based modalities exist, but are limited in their commercial availability and use in comparison solutions that leverage knowledge-based and object-based modality.

Cost is always a consideration for businesses and the choice of authentication technology is usually dictated by its price as much as the level of security it provides. Businesses can utilise their employee’s mobile phones, turning them into object-based authentication devices. Modern mobile phones have fingerprint readers built into the device, theoretically enabling them to be either a biometric-based or an object-based authentication device.

Multi-factor authentication offers enhanced protection to users who may have fallen victim to the phishing attacks discussed in Chapter Three. Whilst keylogging software may capture the password and one-time-token as they are being entered, an attacker would be unable to utilise both pieces of information, a new code would be required to successfully authenticate (Martino & Perramon).
4.2.2 Weaknesses

Multi-factor authentication is susceptible to a number of attacks including Man-in-the-middle and Social Engineering. Man-in-the-middle attacks (MITM) attempt to hijack a session between the user and the system they are authenticating with. MITM attacks often exploit a network or Internet browser vulnerabilities and while this does not exploit a direct weakness of concept of multi-factor authentication, it should not be thought of as a completely secure solution (InfoSec, 2012).

Social Engineering is still a threat. Security tokens that are independent of a network are the most vulnerable. If an attacker is in possession of a target’s password, they can attempt to trick a user into revealing the one-time-code from their security token.

As previously discussed, cost is often a big influence on which authentication method a business chooses to implement. The cost of a multi-factor authentication solution is not cheap. The cost of a RSA SecureKey solution that provides VPN access to remote employees for a large-sized IT corporation is approximately $250,000 per year. This cost includes the hardware, software and security tokens. This figure does not include the cost of maintaining the system or replacing any lost or damaged security tokens.

Research confirmed that users are more security conscious about specific accounts. Users will adopt stronger security practices for online banking and ecommerce and weak practises for webmail. In Weir et al., participants were interviewed about their attitudes towards multi-factor authentication used in online banking. Participants were receptive, despite the additional steps to gain access to their accounts. Participants confirmed they were willing to take these additional steps as it protected their personal finances. The study also confirmed that participants were not receptive to using multi-factor technology across all of their online accounts (Wier, Douglas, Richardson, & Jack, 2010). Similar to the adoption of biometrics, use of multi-factor authentication for all accounts is unlikely and unnecessary. Users will strategically adopt it where they feel it is the most necessary.

Multi-factor authentication requires a user to be in possession of a security token or Smart Card. Research by Florencio and Herley concluded that an average user has 25 accounts. If multi-factor authentication were the industry standard authentication method, users would be required to possess a range of security tokens and keys. Whilst using a mobile phone as an object-based authentication device could reduce the amount of physical keys or tokens, users will undoubtedly be inconvenienced by having to manage many passwords and security devices.
Research by Wimberly and Liebrock revealed that users have a distorted perception of multi-factor authentication, believing it to be completely secure. Participants in their study admitted to using much weaker passwords in multi-factor authentication system and believed the strength of the object-based modality would compensate for the weakness of the knowledge-based modality (Wimberly & Liebrock, 2011). Due to the threat of Phishing and Social Engineering, weak password use will always increase the risk of an attack.

The biggest weakness of multi-factor authentication is when the user is unable to provide one of the required pieces of data. This could be resetting a password or where a user’s token is lost or damaged or the authenticating device is broken. Legitimate users will be unable to authenticate, causing inconvenience and frustration. Businesses are expected to provide backup processes in the event of these scenarios. The backup processes may include weaker alternative authentication methods or require users to call a helpdesk, this costing time, money and resource

### 4.2.3 Recent Advances

In an attempt to increase the security of their account holders, a number of online service providers are starting to offer multi-factor authentication as an option. Whilst its adoption has been prevalent for online banking, many social media and ecommerce services offer, but do not enforce its use.

Businesses are starting to offer multi-factor authentication by SMS. As ownership of mobile phones and their use in everyday activities continues to grow, users will eventually become more receptive to using their mobile phone as an object-based authentication device.

Biometric characteristics are currently incorporated in limited number of authentication solutions. The use of biometrics in multi-factor authentication exists, but has seen limited adoption. The inclusion of fingerprint readers into mobile phones is now enabling users to turn their handsets into object-based and biometric-based devices. Apple is the first businesses to adopt this technology with their upcoming service ‘Apple Pay’ (Apple, 2014). Apple Pay will require a user link their debit or credit cards to their smart phone and using wireless technology, users can authorise payments at retailers using just a single fingerprint from their mobile phone. Apple Pay will be the first widespread mobile solution that combines object-based and biometric-based data.

Recent research suggests that the use of biometrics as part of multi-factor authentication will eventually supersede its use as a standalone authentication technology (Gilady, Kindskog, & Aghill, 2014).
5 Barriers to Adoption

Chapters Two and Three investigated the weaknesses of text-based passwords from a cryptographic and social standpoint. Chapter Four has evaluated the strengths and weaknesses of two alternative authentication technologies. This chapter will consider why one of these alternative technologies has not been adopted. Both technologies offer an increased level of security over the sole use of a text-based password, but their adoption has been limited.

Research conducted by Herley and Van Ooschort in alternative password technologies identified several key barriers to adoption. They concluded that regardless of the technology and the level of security offered, an alternative authentication technology would never see extensive implementation across a range of differing services due to the following factors (Herley & Van Oorschot, 2009).

5.1 Diverse Requirements

Text-based passwords are used to protect a wide range of computer services. This range of services includes online banking, social networking, webmail, news, sports and free webmail. Neither of the two alternative authentication methods is suitable for use across every service. This diversity of security requirements weakens the case for a particular technology to be adopted across all available services.

Research by Wier et al. confirmed this; participants in their study admitted their receptiveness to use enhanced security methods to access online banking, but not for their other online accounts (Wier, Douglas, Richardson, & Jack, 2010). Any alternative authentication solution needs to be designed and targeted to a specific service or use case. A tailored approach will result in an increased chance of success rather than trying to develop a one-size-fits-all solution.

5.2 Competing Technical Proposals

In the last twenty years, there has been no shortage of development of alternative authentication methods and a resulting number of differing solutions, all of which have competed against each other. Text-based passwords were first implemented in the 1960s in a time where no other authentication method existed or was required. Their use was just accepted. As technology evolved, authentication tried to evolve alongside, but due to the diverse requirements discussed above, no overall replacement solution was ever implemented.

Biometrics and multi-factor authentication are both very different technologies and offer differing advantages and disadvantages depending on the service they have been implemented for and the users of that service. Both technologies have their own strengths and weaknesses, with solutions varying in both cost and complexity.
5.3 Customer Loyalty

Many users are fiercely loyal to a particular brand. A user, regardless of its success, will instantly dismiss technology introduced by a rival brand. Apple will be the first major company to offer a biometric mobile phone centred payment service: Apple Pay. Users who are loyal to the brand Samsung are unlikely to use Apple Pay and would not have access to it, regardless of its level of security or its success.

Competing businesses will continue to develop their own technology for use within their own products. While businesses continue to force their users into adopting whatever technology they choose, there will always remain a lack of consistency between technologies. A common framework for authentication is an ideal solution. A common framework would require development and adoption by all industry parties. Due to the competitive nature of the technology industry, cooperation of this nature is almost certainly unlikely to occur.

5.4 Competing Goals Between Stakeholders

There are a multitude of different opinions on authentication security held by various different stakeholders. These include the website owners, browser manufacturers, security technology companies, businesses with an ecommerce presence, governments, regulatory bodies and end-users. Each of these stakeholders has a different end goal.

Research has proven that end-users will typically opt for convenience in comparison to businesses that will strive to ensure maximum security for minimum spends. Governments and regulatory bodies will want to ensure maximum security regardless of cost. Browser and technology companies will want to ensure that their hardware or software is available for as many differing platforms as possible, thus increasing their audience size and profit. Security is often an afterthought.

5.5 User Reluctance

Increased security comes at a sacrifice of user convenience. Research has concluded that users opt for convenience over security. Motivating users to adopt better security practices is increasingly difficult. The use of text-based passwords for authentication is so common, that getting users to give up such an easy practice in favour of a more complex method will be almost impossible.

The end goal of an end-user is to access the system in the quickest way possible. Security is an afterthought for many users and where more secure, but less convenient authentication methods have been implemented, there is a risk that users may defect to a rival system or service in favour of convenience.
5.6 No Control of End-user Platforms

Due to the decentralised structure of the Internet, no business or service has its own end-to-end platform. All services at some point are reliant on another party’s technology, software or platform. Users who access a specific service do so using third party software such as an operating system and web browser.

Services are unable to enforce certain technologies on their users, as they are required to comply with third party software requirements, particularly web browsers.

5.7 No Single Organisation Can Impose a Solution

This is debatably the biggest barrier to the adoption of an alternative authentication method. Due to the large number of businesses and services, each with differing goals and attitudes to security, there is no unity when it comes to developing a solution.

At their peak in 2002, Microsoft had a market share of just over 97% in home and office computing. Limited hardware existed and the libraries and APIs offered by Microsoft largely dictated commercial software development. Almost all software was developed exclusively for the use with Microsoft Windows. This would have been the last chance to introduce an alternative authentication method that would have replaced text-based passwords across all services. If Microsoft had forced users to embrace a specific solution or method, users and developers would have been bound to adopt it as there were very few alternative computing options.

Now, due to the rise of several other technology giants such as Google, Apple and Facebook, Microsoft no longer dictates the software development market and would no longer have the power to enforce such a fundamental change. It has been discussed that due to the Internet’s decentralised structure it is unlikely that a direct replacement will ever be adopted. The Internet is a truly global entity, but has no overall governing body to dictate fundamental decisions such as authentication methods. These decisions are left in the hands of technology companies and industry experts, each with their own goal in mind.

For as long as the Internet exists in the current form, a replacement solution will never be fully adopted or accepted.
6 Conclusion

6.1 Summary of Work Completed

This report aimed to establish whether an alternative authentication technology could replace the use of text-based passwords. Investigation into the cryptographic weaknesses confirmed that passwords are an authentication risk, but if implemented and used correctly, they can be considered secure. Passwords mechanisms need to be implemented correctly and utilise existing technology, ensuring that secure hash algorithms are used, passwords are salted and properly stored in an encrypted format.

It was established that many users are either not aware of, or do not fully understand the risks of poor password management and the threat of password attacks. Their naivety results in the significant loss of revenue for businesses. Regardless of their security awareness, users will choose convenience over security at every opportunity. The optimal solution is an authentication system that is both secure and convenient; unfortunately, findings from the research performed to date would suggest that these two attributes appear to be mutually exclusive.

Users were found to group their accounts into differing priorities. Accounts with the highest priority typically had the most secure password. This same approach ought be taken when designing and implementing the authentication method for a system. Use a more secure authentication method for a system that has a strong business case for its use.

The evaluation of two alternative authentication technologies concluded that neither was a suitable direct replacement for text-based passwords. Each technology had its own unique strengths but other factors established their unsuitability for widespread adoption.

The adoption of biometrics is hampered by users reluctance in allowing their biometric data to be used by businesses for authentication purposes, unless it is necessary or they are legally required to do so. A lack of standardisation and governing body has a continued detrimental effect on the adoption of biometric systems despite significant investment by businesses. Biometric systems are a highly secure system and should be the authentication method of choice for systems that require it.

Multi-factor authentication has seen modest adoption. It is not a perfect solution and nor should it be treated as such. Multi-factor authentication offers an additional layer of protection over and above using just a text-based password. Due to the increase of mobile phones, businesses have a real opportunity to turn their users’ own devices into object-based authentication mechanisms. Businesses should look to leverage this opportunity and where appropriate, offer multi-factor authentication to their users.
The key aim of this work was to establish if there could possibly be an alternative authentication system that could completely replace text-based passwords. The conclusion is that this simply will never happen. There are many barriers that will prevent a completely new authentication technology from replacing the use of text-based passwords. Instead of a service-wide replacement solution, the choice of authentication method should be carefully considered at the time of the system’s design. This design process should establish the level of security required and its target user base. Businesses should adopt the approach of using the correct level of security where it is required rather than trying to implement a one-size-fits-all solution.

During this process, businesses should consider the effect their authentication methods have on their users. Users are simply not just the people who use their service. The users’ influence should not be underestimated; users can ultimately ensure success or failure of a business. Businesses that implement too strong an authentication method risk forcing their users to seek a more convenient service provider. Alternatively, businesses that implement weak security risk users switching service providers with stronger security practices. Striking the balance between security and usability is key.

Due to the diversity of systems, the competing goals amongst stakeholders and a lack of standardisation, a replacement authentication system is highly unlikely to be developed or adopted and no single company is likely to have enough market power to force the change.

While the Internet remains in a decentralised state, the use of text-based passwords in authentication will never be completely replaced.

6.2 Further Investigation

Given more time, additional time and focus would have been dedicated to understanding whether it would be possible to overcome the barriers to adoption. The focus of this work was on the cryptographic and social weaknesses of passwords, but concluded that they are only part of a wider problem.

It was concluded that instead of struggling to develop a solution that would satisfy every service and user case, an authentication method should be correctly evaluated and chosen to best fit the needs of the system and its users. Future work includes the development of a framework to help businesses choose the correct authentication method for systems that ensure maximum user satisfaction and usability.

Alternatively, if text-based passwords were to be used in a system, a framework could be developed to ensure their correct implementation as currently businesses are free to implement whichever security standard or process best fits their business needs and budget.
7 Recommendations

No single authentication method will be appropriate for every system, scenario or user base. Security success can be achieved by ensuring the following:

- Choose passwords ten or more characters in length, using a mixture of alphanumeric characters and numbers.

- Use a phrase when creating passwords. Do not use a word that is found in a dictionary.

- Ensure that passwords are not stored in plaintext and are hashed using a secure hash algorithm. Passwords should be salted to ensure maximum security.

- Ensure users are adequately educated. This education should emphasise the dangers of weak password management practices and provide guidance on secure password creation.

- During the development a new system, the chosen authentication method should be carefully considered. The chosen method should consider all key stakeholders including the business and its users.

- Due to privacy concerns, users will be reluctant to adopt a biometric system unless they feel it is appropriate. The implementation of biometric authentication should be carefully considered and implemented where there is a genuine requirement for it.

- Multi-factor authentication does not have the same privacy concerns but provides an additional layer of security. It should be leveraged appropriately based on the needs of the system.

- Mobile phones can be utilised as low cost, object-based authentication devices. This option should be considered over issuing Smart cards or tokens, as it does not inconvenience users.

- Security, including authentication methods should never be an afterthought. Careful consideration and evaluation of available technologies will ensure the balance between security and usability.
7.1 Extent To Which Aims Were Met
The field of authentication technology is vast and published work within the field is often focussed on a particular area, technology or protocol. The aim of this work was to establish whether biometric or multi-factor authentication would replace the use of text-based passwords. It has been concluded that neither of these technologies will completely replace text-based passwords.

This work was inspired by the work of Herley, van Oorshot and Patrick presented in the research paper aptly named “Passwords: If We’re So Smart, Why Are We Still Using Them”. Their work was particularly brief in the details regarding the cryptographic and social weaknesses of passwords, this work attempted to understand the problem from multiple viewpoints. Herley et al. made several predictions about the authentication methods that would be used in 2019. At this halfway stage in 2014, the conclusions of this work are consistent with the predictions from their study in 2009.

7.2 Personal Reflection
As previously mentioned, the area of authentication technologies is vast. It was believed that one authentication technology could easily replace the use of passwords if users were open to it’s adoption, it has since been concluded there are many other external factors that will ever prevent this from happening, regardless of the strengths of any new technologies.

Prior opinion placed any blame of poor password management solely on the users. It was interesting to learn about the human cognitive and psychological side of password practices and selection. This resulted in a change of opinion at the conclusion of this work. It has been accepted whilst users may responsible for using poor password practices, it’s not always intentional.

This work evolved significantly over time. The original focus of this work was in-depth research and evaluation of two alternative authentication technologies. Over time there became a greater need to understand the problems with the current technology, both cryptographically and socially in more depth. To propose a new solution, there is a requirement to understand the problems with the current system. This was the justification for this change in direction.

Given more time, additional focus would have been given to break down the each of the barriers to adoption. Whilst it has been concluded that due to the structure of the Internet, these barriers will always exist, further exploration is a future interest area. It was interesting to learn about the many external factors that directly influence the authentication technologies used in modern computing.
During the research phase of this work, it was established there is a lack of recent academic research in this area. Many businesses are investing in the research and development of new authentication technologies, but their motives are financial rather than trying to solve the problems with current technology. This led to the question, are academics unmotivated to find a new authentication solution? Are they out of ideas? Do the barriers of adoption restrict their work? Or are the problems with text-based passwords not as severe as originally believed with no requirement for a replacement solution.

The author has concluded as a result of this work that when used correctly text-based passwords are secure and their use will be prevalent in computing for many years to come.
Bibliography


