Clayden Law

Clayden Law, Prama House 267 Banbury Road, Oxford OX2 7HT
E: info@claydenlaw.co.uk | T: 01865 339640

CYBERSECURITY SERIES: PASSWORDS

The use of passwords and their management is not a new subject, but they continue to have such an impact on the likelihood that an organisation will suffer a data protection breach that they deserve particular attention.

Passwords are an imperfect example of IT security, but they are likely to continue to be the first line of security in the majority of systems we use. With that in mind, this article: presents an overview of what passwords are intended to achieve; why they are so hard to use safely; why they have yet to be superseded by new technologies (e.g. biometric authentication); and what users and service providers can do to make them more secure.

In a privacy context, implementing a strong password requirement is a well-recognised means of ensuring compliance with the seventh data protection principle of the Data Protection Act: preventing unlawful access to personal data by using a technical measure. However, crucially, the protection must govern not only the requirement to *have* a strong password, but also implementing appropriate security measures to *manage* any user-entered password as well as the *integrity* of the server side verification mechanisms for password validation and storage. This principle will continue to be important under the new data protection regime, the General Data Protection Regulation, from May 2018.

WHAT DO PASSWORDS ACHIEVE AND WHY ARE THEY SO DIFFICULT TO USE?

Passwords are used where an application needs to provide a way for a user to prove who they are, their right to access specific data or just to ensure that they are a human being. A password is a short string of characters, typically chosen by the user and composed of a combination of alphanumeric and special characters, depending on the requirements of the particular application.

Theoretically, passwords are a simple and effective solution to prove that people have the right to access something – passwords as a concept existed prior to the advent of computing technology, making them accessible to even the most novice IT user. More importantly, they are pervasive because they are 'free' for the developer to implement and don't need the customer to invest in additional technologies, such as card readers or biometric systems. Unfortunately, all of the cost of this security measure then falls on the user[1]

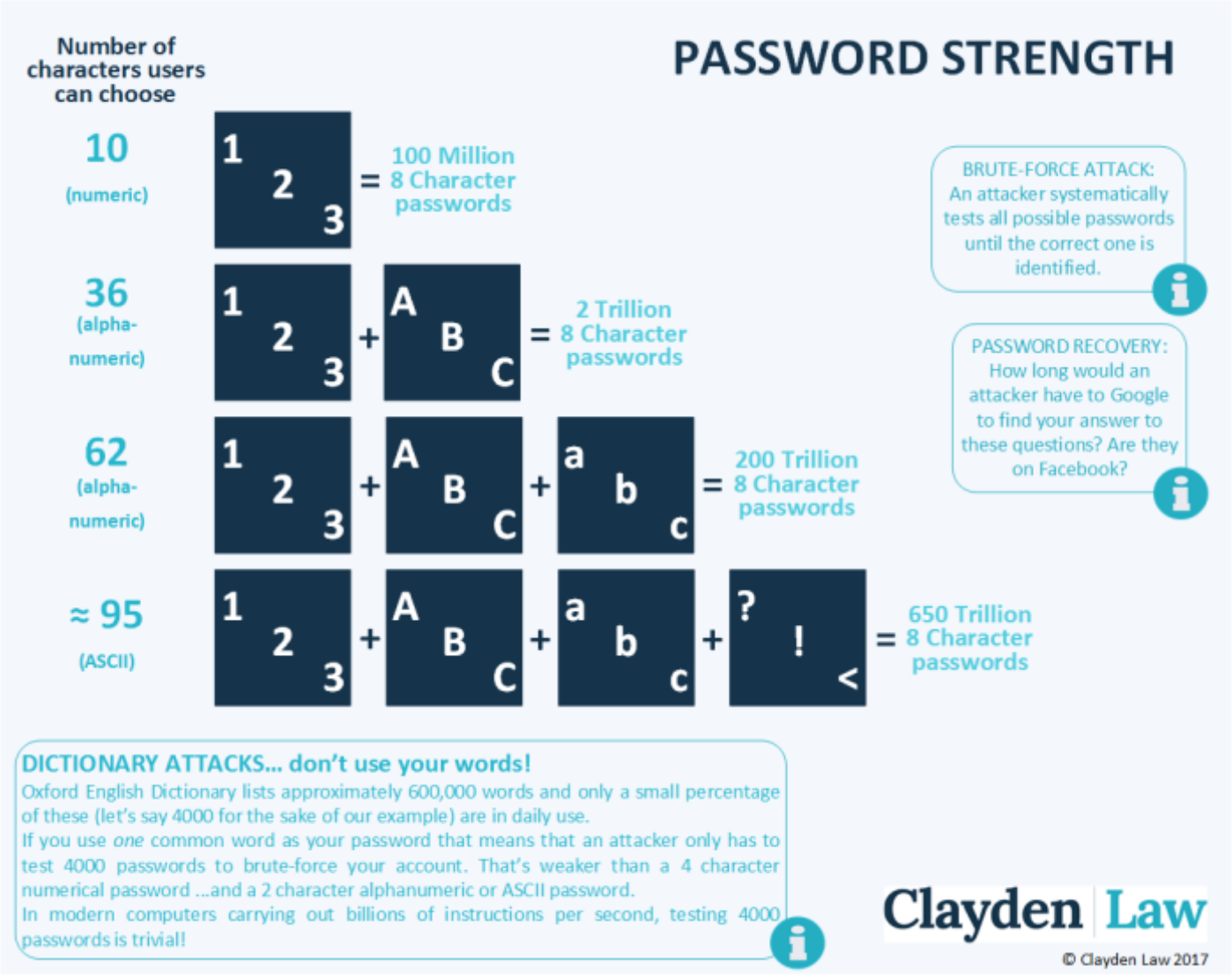
The low cost perpetuates the use of passwords, but the level of complexity of having quite so many passwords perpetuates their associated security flaws.

WHY ARE PASSWORDS HARD TO USE SAFELY IN PRACTICE?

When combined, two established pieces of security advice have had the unintended consequence of ultimately reducing password security:

1. that passwords are **complex** (so that they cannot be guessed) and **unique**; plus
2. that passwords are **never written down** and **changed regularly.**

[1] https://www.microsoft.com/en-us/research/publication/so-long-and-no-thanks-for-the-externalities-the-rational-rejection-of-security-advice-by-users/



Thankfully the second piece of advice has been proven by security research to increase mistakes, and is no longer required in most security policies. Many organisations have even concluded that a set of short and easy to remember responses to questions is more effective than one difficult to remember password.

However, passwords *do* need to be complex, even if this makes them difficult to recall without an 'aide-memoire'. In addition, it is still poor physical security to write a password down on a post-it left in an easy-to-find location. This combination of requirements reduces usability to a point that even cybersecurity experts are likely to find hard to manage!

Ultimately, passwords are keys and – like physical keys – there are some we need once a year and some we use several times a day. We are more likely to remember the passwords we use most frequently, while the rest belong written down in our wallets, a safe or a locked draw next to all of the real-world keys that we keep, but rarely use.

While new mechanisms for securing IT applications are a common occurrence, passwords are not going away anytime soon. The most effective course of action in the meantime appears to be: *to prevent the escalation in quantity of passwords in daily use*; *help users choose strong passwords* (and

to offer advice as to *why* this is important) and *encourage application providers to fortify their applications at the server side with appropriate controls* (more about this below).

WHAT IS A STRONG PASSWORD AND WHY IS IT SO IMPORTANT TO USE ONE (AND ONLY ONCE)?

Many applications have requirements to ensure a certain level of complexity in the passwords that users create. So much so that people struggle in the first few attempts to create one that meets the criteria, as these requirements will be different in each application they interact with.

The danger of these differing password requirements is that users are likely to end up choosing the lowest common denominator, that is, the password based on the ***most restrictive set of rules***, which fits all of their applications producing an inevitably less secure password. This is often true even when users select different passwords for each application they encounter, as many people have a pattern or system for creating passwords, for example using the name of the application alongside a random string of characters that is reused in all passwords. Ultimately, the greater the number of characters that can be used in a password, and the longer it can be, the more secure it can potentially be.

One reason IT standards may expect people to change passwords regularly is that it's often so hard to work out when there has been a breach that this reduces the length of time an attacker has access. As with other security measures, implementing multiple layers of security is likely to reduce the impact of any personal data breach.

Our infographic provides some examples of the differing strengths of different types of passwords.

WHAT DO SERVICE PROVIDERS HAVE DO ON THEIR SIDE TO PROTECT OUR PASSWORDS?

A personal data breach is likely to result in enforcement action by the ICO where basic precautions have not been not taken to protect passwords. Such a breach may arise due to an oversight, or because the organisation deemed it to be unnecessary, or too costly. A proper consideration at the design stage of the risk involved in failing to secure users' passwords would highlight the potential vulnerabilities and justify the expense of investing in extra security. This will also help an organisation to achieve compliance with new obligations under the GDPR for *data protection by design*.

An example to illustrate the point above is the LinkedIn data breach in June 2012. This led to attackers gaining access to 6.5 million passwords, all because of a combination of using a weak hashing algorithm (SHA-1) and not salting passwords. In 2016, LinkedIn became aware of an additional 100 million accounts that were compromised in the earlier breach and had to invalidate those passwords, requiring users to re-configure the security on their accounts.

But what is hashing?

Cryptographic hash functions are integrated into systems for security, providing a combination of confidentiality and integrity. They are a particular type of cryptography, which irreversibly but systematically 'jumble' content. When text is hashed the resulting string is the same every time, but the algorithm used makes it almost impossible to guess what the original text was.

Cryptographic hash functions are frequently used to ensure the accuracy (integrity) of data sent across insecure channels. As well as being encrypted, the data is *signed* using a hash function, so that when it arrives at its intended destination, the recipient can check that the information has not been manipulated by an attacker, or otherwise corrupted.

Additionally, in the case of passwords, cryptographic hashing is also about confidentiality – if hashed passwords are released to an attacker in a security breach they do not provide the attacker with the information they need to break into a user's account.

Salting passwords before encryption adds extra security. The process adds extra characters to a password before it is hashed, protecting against *rainbow table attacks*. Hashing algorithms can be given any length of string (zero to a very large number of characters), but the resulting digest is always the same length. This means that for each hash there are a large number of potential passwords. Rainbow tables are lists of unsalted passwords that correspond to a certain hash, but by adding extra random data the rainbow tables become too unwieldly for an attacker to use.

When using cryptographic hashing to comply with data protection regulations there are a number of elements to consider in the implementation of the password management system:

***Do* use a hashing algorithm:** passwords should never be stored in plain text by an organisation;

Strength of hashing algorithm: it is important to check that the hashing algorithm used is generally considered strong (currently SHA-2 or SHA-3) and not weak or antiquated, e.g. SHA-1;

Salt before hash: ensure that the password is salted (randomly generated redundant characters are added to the plaintext password) before hashing.

If possible, encrypt the hashed passwords: encrypting the hashed passwords will provide an additional layer of security.

MULTI-FACTOR AUTHENTICATION

Passwords are something a user *knows*. There are other mechanisms available for authenticating users, which typically fall under the categories of something a user *has* (their bank card and card reader, or their mobile phone for example) or something a user *is* (typically biometrics).

On their own these methods are not often any more secure than a password. This can be pictured as follows: if someone can steal a password from a post-it on a desk then it may be equally straightforward to take a bank card from a purse or wallet found next to it.

Many biometric systems have been shown to be vulnerable to attacks involving reproductions of biometrics (photos etc.) that can be used at any distance from the victim. However, the field of biometric authentication is still rapidly evolving and improvements may yet appear in these systems to increase their overall security.

Some service providers are offering users the *option to activate a second layer of authentication* that entails codes sent via email or text message, or the use of a card reader to generate a code. This second layer, often only activated for a limited set of low-frequency activities, does two things:

1. requires the attacker to put significantly more effort into obtaining full access to an account since they have to obtain multiple credentials across different platforms; *and*

2. the victim usually receives notifications as soon as one of these actions takes place in multiple forms – text, email etc. Even if the attacker succeeds in compromising the account, the victim swiftly notified. They often have the chance to mitigate or prevent loss following the compromise.

Multi-factor authentication has gained traction with application providers, including banks, as the second layer reduces the risk of data breach and the potential harm that may occur as a result. In the case of high-sensitivity or high value personal data it is becoming good practice to provide users the option of a second factor of authentication, but usability issues means that they are not often compulsory.

ROUNDING UP

Using physical security as a metaphor, passwords are our keys to locked doors in the virtual world, and with the increasing value of the data we store online, there are clear benefits to treating passwords with a similar level of security as our house keys. As well as the challenges passwords present to users, passwords are challenging for service providers to implement, especially justifying investing in the level of behind-the-scenes protections needed to fortify a password system.

Privacy legislation, certainly once the GDPR comes into effect, will expect **greater accountability** (this is in fact a new data protection principle underpinning the GDPR) and enforce substantially greater penalties on organisations that do not invest in suitable technology to safeguard password security, especially where a personal data breach can lead to data subjects suffering financial loss. This is aside from the reputational cost that may be inflicted, along with any other losses and claims.

Organisations, if they have not already done so, would be prudent to review their password technology from user and server sides prior to the implementation of the GDPR. The use of multi-factor authentication in particular may be worth investing in, especially for more valuable personal data.