How to design a password policy

Provide safe and simple access

Taz Wake
Halkyn Consulting Ltd
Security & Risk Management Specialists

How to design a password policy

Passwords are probably the most widely used authentication mechanisms available. They are, on the whole, easy to deploy, easy to explain to users and easy to manage. It is likely that every online application you interact with, from your bank, to social networking, to email, will require you to enter a password to gain access. Passwords are everywhere.

There is a modern trend which tries to describe passwords as "outdated" and too easy for hackers to compromise – this is frequently pushed by providers of alternative solutions such as smart cards and biometrics. While there is an element of truth (advances in computer hardware, make brute force attacks on passwords easier), this simplistic approach ignores some major problems and calls for a simplistic response: Passwords are not "weak" nor are they an outdated. They provide one of the most cost effective, scalable solutions available and enable you to establish some form of authentication with an unknown third party.

Passwords are not perfect, but no security control is. You must always keep in mind the fact that passwords do have weaknesses and should only be deployed as part of an overall security system. Passwords are only there to enable a person to authenticate themselves to a resource. This really is important.

So we can agree that passwords are useful. The problem is that people often get confused as to how to implement them.

The Problem with Password Policies

When you create an account with your bank, or twitter etc., you have to comply with the policies that they have set and frequently this will be along the lines of a minimum number of characters and a minimum mix of upper/lower case letters, numbers and other "special characters."

With these remote services, there is no way for us (as a customer) to determine what (if any) reasoning went into their choice of what makes up a suitable password, nor are we likely to know what other security controls they have in place. This means we don't have any insight into the rationale that leads to the choice of password rules.

However, our daily exposure to the rules of other sites leads to people who are responsible for establishing their own password rules falling back on very simplistic often irrelevant principles. Badly designed password rules have a worrying tendency to either be insufficient for the environment they are in, or so complex they are routinely broken (or result in excessive calls to the helpdesk, wasting time and money).

To make matters worse, often these poorly designed password policies cause so much trouble that the organisation moves towards additional costs and complexity to implement multi-factor authentication. Security has seriously failed if it forces a business to spend huge sums of money for an unnecessary control instead of just planning the password implementation properly.

Remember, passwords are almost the least likely security control for an attacker to breach and almost never act as the entry point for a major breach (for example, not one of the "Best (or Worst) Breaches of 2010" were the result of as password compromise). Do not assume that upgrading your authentication will improve your security. Often it won't – multifactor authentication only works to enhance a good authentication strategy, it won't create one from scratch.

How to Design a "Good" Password Policy

It is surprisingly easy to build a good password policy that is tailored to your business needs. If you are responsible for determining what your organisations passwords should be like then you owe it both to your employer and your own sense of personal pride to design one rather than bolt on the rules someone else uses somewhere else and hope they will be effective.

1. **The first thing is to design the password policy**. By this we mean accept the fact that you are going to spend a little bit of time thinking about how passwords fit into your business and what the best way to use them is. Do not automatically assume the rules given by your bank are applicable.
2. **Assess the situation**. Look at what security controls you have in place and how important is the resource you are controlling authentication to. You really need to understand this if you want to design a good policy.
3. **Assess the threat**. Worryingly, this is the most overlooked step. If your resource doesn't have an internet facing portal, then you don't need to worry about global

hackers so spend time on this stage. Likewise, if you have robust access controls and staff vetting, you might be more relaxed over writing passwords down.

4. **Abide by regulations**. If you are in an industry which is regulated or where there are required controls (such as compliance with PCI-DSS or government standards) then make sure you are aware of what these requirements are. In general, it's always worth exceeding a regulatory requirement where feasible as this helps prevent you being "low hanging fruit."
5. **Design your password requirement.** So that it matches the threat you face, comply with any applicable regulations and suitably protect the resource you want to guard. It sounds simple, because it is.
6. **Implement and educate**. Just as important as making sure the password rules are sensible and properly implemented is making sure your staff are properly educated in why they should comply (or ideally exceed) the minimum requirements. Encourage employee buy-in so they can be part of the solution rather than another security risk.

Things to consider

With passwords, **longer is always better**. Using *R6gY&@pda* might "look" like a super-secure password but all things being equal, a hacker will compromise that faster than *This is my super-secure password*.

Setting rules is a trade-off between encouraging (forcing) good user behaviour which actually makes the password easier to compromise, and risking lazy users selecting easy to break passwords. A rule which says passwords should comprise 8 characters chosen from any letter (upper or lower), number or special character results in a theoretically harder to compromise password than one which says there must be at least 1 number, 1 capital letter, 1 lower case letter and 1 special character. You have to decide how important this trade-off is for you.

Writing down passwords is only a problem if there is a **genuine** way for your threat to get physical access to your facility. In most circumstances if they can do this, there are better ways they can gain access to your systems and other parts of your security are seriously

compromised and should be improved. **Hackers on the other side of the planet are not likely to travel to your building and then try to con their way in** – this is more a hallmark of penetration testers than it is of genuine attackers. If you are worried about distant hackers, use long, random passwords and let your staff write them down. If you are worried about customers or visitors shoulder surfing, then use short, easy to memorise passwords. Defend against **your** threat, not the one the NSA, CIA or GCHQ faces, nor the one the latest ex-hacker is publicising in his book.

Enforce lockouts after a set number of retries. Don't be fooled into thinking there is a "best" number for the retries before lockout, use what is best for your business. However, if you don't lock an account (even if just for a few minutes) after a set number of failed retries **your password will be broken eventually** (even if it takes years). This is probably the single most important thing you can do with your password. If you have a password of 8 letters (single case) a brute force attack can compromise this in about 3.5 minutes, however if you lock the account for 15 minutes after 3 failed attempts, this increases to hundreds of years (a maximum of 12 attempts an hour, 100 billion combinations needed). **This simple control is massively effective**.

Password change intervals should be set to less time than you would expect the password to be compromised by a brute force attack. A password made up from 8 characters, using every character on the keyboard is likely to last less than 84 days under direct attack, so if that was your chosen password length, you might want to implement a password change rule every 45 days.

Remember, passwords are good, cost effective authentication controls which come supplied with every operating system and most applications. Do not discount their usefulness simply because people have got lazy in how they implement them.

Build a sensible password policy that is relevant for your organisation to make best use of the security provided by single factor authentication.

About Halkyn Consulting

Halkyn Consulting Ltd is a specialist security consultancy based in North Wales, experienced in providing security advice to local, national and international clients including government agencies, multinational corporations, small businesses and homeowners.

For more advice on how to properly use passwords or any other questions you may have related to security and protecting your home, business or other assets, you can reach Halkyn Consulting at www.halkynconsulting.co.uk or info@halkynconsulting.co.uk