Global Knowledge®

Expert Reference Series of White Papers

10 Steps to Better, Stronger Passwords

10 Steps to Better, Stronger Passwords

James Michael Stewart, CISSP, ISSAP, SSCP, CPTE, CDFE, Q/SA, Q/EH, CEH, CHFI, Security+, Net+, and A+

Introduction

It has been years since single-password authentication was even potentially a good security idea. You should consider any static password that you can remember as vulnerable. Let's take a look at what makes a good password and then examine ten easy steps you can use to make your password as secure as possible.

Accountability

Security can be defined in many ways. One way is accountability, which is when security is holding people responsible for their actions. In order to hold someone accountable, three elements must be present.

- **Authentication** is the proving or verification that a specific person is who they claim to be. In most cases within a computer network, authentication is used to link a specific person to a specific user account. When a person attempts to log on, they claim an identity, often by typing in a user name, then they must provide authentication factors to prove that they are (or at least are responsible for) the claimed identity.

- **Auditing** of events is the recording of all activities of the system, resources, and users. This creates a log trail of everything that took place within the computer network and to some extent within the organization's facility during a specific period of time.

- **Authorization** is the assignment of rights, permissions, and privileges to users that enables them to accomplish their assigned work tasks. Authorization is also the prevention or denial of access to any resource or activity that is not granted to a user. Thus, authorization is a collection of allows and denies that define the activity and access boundaries for a user. Every user will have their own unique, custom, and focused set of access boundaries.

Authentication

Of these three essential security services, authentication is the most important. Failing to prove a solid and unassailable link between a digital identity (i.e., a user account) and a person prevents us from holding someone accountable for the recorded actions of a user account. Without strong authentication, it is not possible to hold someone accountable.

Unfortunately, authentication is where most systems, services, online sites, and organizations fail in their attempts to provide accountability security. The reason for this failure is passwords. Passwords are the most com-

monly used form of authentication. However, in practical terms and use, they end up being the least effective form of authentication. This is caused by several factors.

- **Most organizations and services rely only on passwords for authentication.** When a single factor authentication mechanism is used, especially when that single factor is just a password, a single successful attack against a user account, person, or password is all a hacker requires to impersonate someone and log in as the victim account.

- **Most organizations leave password selection up to the end user.** Most end users pick passwords that are easy for them to remember. The fact that a password is easy to remember makes it a password that is easier to guess, discover, or crack.

- **Common or "standard" password security policies, guidelines, and training do not help against modern password cracking techniques and tools.** Forcing users to employ one or two uppercase letters, numbers, and symbols, or requiring a specific number of characters (even setting a small range of allowed lengths, such as 8 - 12 characters) actually makes the task of password compromise easier. For example, if a hacker knows your company's password policy, then they can automatically exclude any password that does not fit your requirements, such as anything missing an upper case characters or anything with too few letters.

- **People are lazy.** Even with good recommendations for password creation, most users only perform the minimum requirements in order to skirt the rules. Most users do not truly understand the point and purpose of the rules. Instead, by using the minimums as if they were exclusive requirements, this gives the hacker even more of an edge. If your policy is to require 2 upper case letters, even though there is not a restriction on using more, most users will only employ 2 upper case letters. Hackers study human behavior and use this foible to improve the success of their password attacks.

- **Too many real-world passwords have been hacked.** Hackers have an overwhelming amount of knowledge about general password rules, guidelines, and selections. This insight into how we, as general computer and Internet users, select passwords makes password cracking easier and faster.

Password Cracking

Hackers have compromised hundreds, if not actually thousands, of large company networks and popular online services. Many of these compromises have granted hackers either direct or delayed access to user passwords. Direct access to user passwords occurs when user account credentials are stored in a **cleartext** form. Thus, once the user database is accessed, all of the user passwords are directly available. Delayed access occurs when the user account credentials are stored in some form of hashed, encrypted, or other protected form. This requires the hackers to crack the passwords. Password cracking efforts can sometimes be effective nearly immediately (i.e., within a few hours) or may take considerable time and effort (i.e., several weeks or months). Generally, hackers will continue to crack passwords until something more promising comes along, and they need their resources for something else.

Over the years, hackers have learned the passwords of billions of user accounts. Many of these are duplicates, like 1234567, or monkey, or princess, or P@ssw0rd. Thus, over 100 million unique passwords are now available to hackers. Just about every possible combination of keyboard characters in shorter passwords is known, but there are some 20+ character passwords in this collection, which is known as a dictionary list. If you happen to have selected a password that is on this list, then if/when a hacker attempts to crack your password, they are just about guaranteed to be successful.

So, why is this large database of passwords so important to us? Well, it reveals just about every conceivable trick or pattern or clever mechanism used by users. Tricks such as shifting every other letter, making a pattern out of the keyboard, using words spelled backwards, replacing certain letters with numbers or symbols, commonly upper-casing first letters, and adding numbers or symbols only at the end of a base word. If user habits of password generation or selection are all known, then it is just about impossible for a typical user to have a chance of picking a password on their own that they can remember.

Mitigating a Multi-factor Authentication System

The best way to solve the problem of relying too heavily on password-only, single-factor authentication is to migrate into a multi-factor authentication system. If this is offered by your company or any online service, be sure to take advantage of it. Today this already includes many well-known Web services such as Google, Facebook, Twitter, Ebay, Paypal, and Dropbox. If a site or service that you use does not already offer a multi-factor authentication option, send the site owners/managers a request to improve their security by adding in multi-factor authentication.

When you are stuck with password-only, single-factor authentication, there are steps you can take to minimize your risk and improve your password security.

Ten Steps to Better, Stronger Passwords

1. DO NOT RE-USE PASSWORDS!

Never, ever, ever use the same password twice – not on the same system and not on different systems. Think of passwords as a consumable item, like the windshield wipers on your car. Once they are used and replaced, you never re-use the old ones again – ever! When you re-use a password, there is the potential that the old use of that password has been compromised in the intervening time frame. Now that you are using it again (or using it someplace else), hackers have a higher chance of discovering that you are re-using your old password. This might be due to the hash being the same, or they happen to be using the dictionary list with your old password in it.

2. Stop trying to be clever.

Hackers have too many examples of clever in their password database. Just about every trick or technique you can come up with on your own – based on your keyboard or mental calisthenics – is already known and often built into password cracking tools. You need to start basing your passwords on randomness.

3. Use a tool.

Specifically, if not prohibited by your organization, consider using a secure password management tool. Even if you are not allowed to use one at work for company systems, at least use one for all private access to Internet

sites and services. You need a password management tool that encrypts your password database and offers random password generation. My personal preference is LastPass, but Keepass, Onepass, and RoboForm are also great options. Some of the password storage tools are tied into a cloud service that can give you access to your securely stored credentials from any Web terminal or via a smart phone app. The most important aspect of a secure password storage tool is that you are the only and sole possessor of the master unlock code to your credential archive. Having your credential archive stored in the cloud, as long as it is encrypted and even the hosting company and software vendor are unable to open your personal password locker, can be considered safe.

If you choose to use a tool, be sure to set the master unlock code to as complex a password as you can muster. Again, this should be as random as possible, but you will need to MEMORIZE this one (see step 4). You will then let the tool generate truly random passwords for all other needs, and let the tool keep track of them for you.

4. If not using a software tool, consider 3 alternatives: passphrases, random password generators, or paper-based tools.

- **A passphrase is a multi-word password.** Instead of picking a single word or a single string of characters as a password, pick 3 to 6 words and use them as a collective. You can put a space or a symbol between each word. Try to avoid picking words that go together in any way. You also want to intentionally stay away fr*o*m crafting understandable sentences. Instead, pick words that might tell you a mental story so you have a chance of remembering it. For example: "purple balloon underground flights intentionally upsetting." This is an exceptionally long password and even as lowercase-only would be improbable for a hacker to figure it out using traditional password cracking techniques. (I'm skipping over the problem of keystroke logging, video cameras, shoulder surfing, social engineering, passwords stored in cleartext, non-encrypted, or poor encryption used for transmission, etc.)

- **Random password generators abound.** There are computer software tools and smart phone apps you could use to generate your passwords. I prefer an online service called Perfect Passwords at https://www.grc.com/passwords.htm, which is GRC's *Ultra High Security Password Generator*. This site generates random 64/63 character passwords in Hex, ASCII, and just alpha-numeric. You can choose to use all or just part of one of these random passwords. Plus, each time you refresh the page, a whole new set will be generated. By the way, Steve Gibson (the man behind GRC – Gibson Research Corp.) does not track what is generated on this page nor who uses it. The only problem you have with a random password generator is remembering it. See my suggestion in step 5.

- **Use a paper-based tool.** A paper-based tool is a means to generate random passwords on the fly that you can re-create when needed. But do so in a way that cannot be deciphered or predicted by others. I've seen several attempts at this, but the only one I have seen survive real-world use and intensive security community scrutiny is another brainchild of Steve Gibson called *Off The Grid* (https://www.grc.com/offthegrid.htm). This is a page that generates a random 26 x 26 Latin square. A Latin square is a special square array where each value (i.e., letters, numbers, and symbols) appears only once in any row or column. (Sudoku is a form of a Latin square). To use *Off The Grid* for passwords, you generate and print off a random Latin Square. Then using your printed matrix, you generate passwords – passwords that are based on random values but that you can re-produce by repeating your generation method and maintaining possession of your printed Latin square. Please see Steve's excellent documentation, the feedback, and even listen to the podcast discussing this tool to gain a

fuller understanding of this technique and its benefits. If you are unable to use a software password tool at work, a paper-based tool could be the best option.

5. If you write it down, write it down securely.

If you have passwords that are so long and complicated that you are unable to remember them, then you may have to write them down. We all know the rule "never write down your password." However, that rule focuses on writing your password down in a blatantly obvious form and potentially in an easily discoverable location. Always follow your company policy on this, but consider discussing the following ideas with your security manager as potential exceptions to the "do not write" rule.

- **Don't ever write down a password in such a way that it seems like it is a password.** For example, don't write the word "password" or "P:" in front of it.
- **Don't ever keep or leave the written password near the computer.** To be even more secure, don't keep written passwords on the same desk or in the same workspace as the computer.
- **Always write down passwords in code.** Consider writing down in alternating shifting patterns. Such as if your real password is "PassWOrd12" then write down "pAsSwOrd!2" where I inverted the "shifting" or case of each letter as I wrote it on the paper. You could also write in reverse order, such as "21drOWssaP". You could write inside-out, meaning write down the first letter, then put the second letter in front of the first, then the third letter behind the first, then the fourth in front of the second, etc., such as "2dOsaPsWr1". You can use a number of different patterns or encoding techniques. If you can make sense of it, you might even try combining two or three techniques.
- **Never write down a whole password in one location.** In spite of my previous examples, try not to write down the whole password in one spot. Instead, break up your password into 3 or more sections and record each section someplace unique. Maybe one section is on the back of your insurance card in your wallet, another on the back of a house key, a third inside the case of your cell phone, and a fourth on the back of the 8th from the last page in the small notebook you carry in your backpack or purse. This means that even if someone stumbles across one of the section of your password, they will not know the whole password or what order the sections go in.
- **Consider dropping letters.** Instead of recording every letter in your written version, consider always dropping every 3rd letter or the last 2 letters. You will have to be sure you can remember what you are not writing down, but it is another option.
- **Consider having a standard "padding" phrase.** Using the previous suggestions in the section, record the main portion of your password, but always add in padding material. You could either define your padding material as part of the real password, such as "B1u79e$", maybe in the middle or at the end of every password. Or you can include it only in the written version of your passwords, but you must remember to skip the padding material when you type in the password.

There may be even more techniques for writing down passwords in a secure manner than these few I've presented. With a little searching you can find other suggestions. However, ALWAYS review the concept or idea with your security manager before using these writing down techniques at work.

6. Stop replacing your password every ?? days/weeks/months.

If you have a secure password, there is no need to change it periodically. If you have a weak, guessable, or easily crackable password, you should change it to a secure password. Changing passwords just because time

has passed does not benefit the typical user. In fact, by making users change their passwords on a fixed time interval, they are encouraged to pick another new easy-to-remember password, which defeats the whole purpose of authentication in the first place. If you are forced to change your password by company or service policy, then follow that policy. But if you are not forced, keeping a truly secure password is a good practice. The only exception to this is if the site, network, or service is compromised and user account credentials were accessed by a hacker. Then, it would be a good idea to change your password.

7. Use all possible character types supported by the system.

In most cases, lowercase, uppercase, and numbers are universally supported as valid password characters. If you can use symbols, higher order ASCII characters, or even foreign language characters, do so. The wider the potential character set that you choose from, the more potential complexity a hacker must overcome in order to crack your password. Most password cracks focus on the low hanging fruit. There is almost always someone within your organization who has just barely qualified their passwords based on construction rules, but still ended up with an easy to remember and guess password. You want to avoid being a low hanging fruit.

8. Make your passwords longer.

Length is the primary factor in a password's strength. Complexity is a close second. If you only used lowercase letters, then a password 20 or more characters would be a fairly secure password. Using a long password that is also character -complex is even more so. Always try to maximize the length of your selected password based on length limitations of the site or service. If you are limited to passwords under 12 characters, be sure to inquire of the site owners and managers why there is a length restriction at all, much less such a small one. If a site is actually hashing your password, it does not matter how many characters you type in, as a hash produces a fixed length representation of your imputed password. If a site is using a hashing mechanism that produces a 256-bit hash, it doesn't matter whether you type in 8 or 16 or 1000 characters for your password – as the hash of that password would still be 256 bits.

9. Go beyond the minimum requirements.

As discussed earlier, hackers understand that human nature often leans toward laziness, and barely complying with construction rules is a common characteristic of compromised passwords. Try to go beyond the minimum requirements whenever possible. If you are required to have 2 of a character type, try using 4 or 5. If you are required to have at least 10 characters, why not have 20 or 30? If you are asked to define a password, why not construct a lengthy multi-word passphrase?

10. Go random.

You are human. Your cleverness and ability to craft memorable passwords is not unique. In fact, it is very common, typical, and predicable. Don't try to invent a new clever trick to create a pattern or construct a memorable password. Assume anything you can think of the hackers have already seen thousands of others do, and they can recognize it when you try. Your only hope is to change the rules of the game and that is to use random passwords rather than memorable passwords. And, yes, because random passwords are very hard to remember, you will need a secure software tool, a secure means of writing them down, or a secure means of generating/regenerating them.

Summary

The primary point and purpose of this white paper is to help you to understand that easy-to-remember passwords are no longer considered a secure form of authentication. In fact, it has probably been several years, maybe even a full decade, since single-password authentication was even potentially a good security idea. You should consider any static password that you can remember as vulnerable. Even static passwords that are random are still vulnerable to some extent – it just takes much longer for a password cracking attack to be successful, and the likelihood of that success is inversely proportional to the length of the password. There is always a chance; you are just reducing that chance by having a long and random password.

Do the best that you can within the dictated rules, but try to implement as many of my suggestions as possible whenever you are limited to single-factor, password-only authentication. It is also up to you to be the squeaky wheel and inform others about the insecurity of passwords and to encourage site owners and managers to provide more secure multi-factor authentication options.

Learn More

To learn more about how you can improve productivity, enhance efficiency, and sharpen your competitive edge, check out the following Global Knowledge courses:

CISSP Prep Course

Certfied Ethical Hacker v7

Security+ Prep Course (SYO-301)

Security+ Boot Camp (SYO-301)

Visit **www.globalknowledge.com** or call **1-800-COURSES (1-800-268-7737)** to speak with a Global Knowledge training advisor.

About the Author

James Michael Stewart has been working with computers and technology for nearly thirty years. His work focuses on security, certification, and various operating systems. Recently, Michael has been teaching job skill and certification courses, such as CISSP, ethical hacking/penetration testing, computer forensics, and Security+ for Global Knowledge. He is the primary author on the CISSP Study Guide 6th Edition, the Security+ Review Guide 2nd Edition (SY0-301), and Network Security, Firewalls, and VPNs. Michael has also contributed to many other security focused materials including exam preparation guides, practice exams, DVD video instruction, and courseware. In addition, Michael has co-authored numerous books on other security, certification, and administration topics. He has developed certification courseware and training materials as well as presented these materials in the classroom. Michael holds variety of certifications, including: CISSP, ISSAP, SSCP, CPTE, CDFE, Q/SA, Q/EH, CEH, CHFI, Security+, Net+, and A+. Michael graduated in 1992 from the University of Texas at Austin with a bachelor's degree in Philosophy. Despite his degree, his computer knowledge is self-acquired, based on seat-of-the-pants hands-on "street smarts" experience. You can reach Michael by e-mail at michael@impacton-line.com.