On password management

Katsutoshi Seki | Published: December 25, 2025 | Source: https://sekika.github.io/2025/12/25/passme/

Remembering multiple complex passwords is difficult. For this reason, many people likely use simple, easy-to-remember passwords or reuse the same password across many sites, even though they know it's not a good practice. This article offers some tips on how to manage your passwords.

1.Principles of Password Setting

There are two main principles for setting passwords.

1.1.Principle 1: Use Complex Passwords

There are various levels of "complex passwords," but you should especially avoid using the "extremely simple passwords" that many people use. Attackers trying to steal accounts will attempt to break into sites using lists of these "commonly used passwords" to increase their success rate. Using a simple password raises your risk of being compromised.

Easy-to-remember passwords, such as those using words or names, are "easy-to-guess passwords" for an attacker, making them weak. To make an attack less likely to succeed, you should set a "complex, hard-to-remember password" — in other words, a "strong" password. The longer the password and the more character types you use (e.g., lowercase and uppercase letters, numbers, symbols), the more difficult it will be for an attack to succeed, making your password stronger.

1.2.Principle 2: Do Not Reuse Passwords

The purpose of this principle is to prevent password list attacks. If a password you registered on one site is leaked for some reason, lists of usernames and passwords may circulate. If you have the same password set on multiple sites, an attacker can use the "password leaked from one site" to successfully log into your other accounts, causing the damage to spread. Since you won't necessarily be notified that a leak has occurred, thinking "I'll just change my password when I find out it's been leaked" is not enough.

2.Password Management Methods

Setting "strong passwords" "uniquely for each site" like this makes them impossible for a person to remember, so you'll need a tool or method to help manage them. Here are three methods[1], listed in order from what anyone can easily do to methods for more tech-savvy people. Feel free to choose the method that seems most convenient for you and add your own customizations.

Also, even if you manage your passwords with methods like these, your password can still be stolen if, for example, you fall for a phishing scam. What is written here does not cover all security measures.

2.1.(1) Paper Memos

"You should never write down your passwords on paper" was common advice in the past. This is because if you leave that paper where others can see it, your password could be easily stolen. On the other hand, information written on paper has the advantage over a digital file in that it cannot be stolen over the network by a remote attacker. To enhance the security of this method, it's desirable to not only "store the paper in a place where others are unlikely to see it" but also to "make it so the password cannot be easily guessed even if it is seen." To do this, you create a transformation rule to convert the password written on the paper into the real password, and you memorize that rule. For example, you could have a rule like adding a fixed string such as yaQ to the end[2]. In other words, on paper, you would write:

```
○× Bank
ID: seki
Password: xTL2WLGz0P
```

And then you would append your fixed string yaQ to xTL2WLGz0P to form the password xTL2WLGz0PyaQ. If you use this "yaQ" string in common for other sites as well, a string of this length should be memorable (though you might forget it if you don't use it for a long time, so it's a good idea to devise a way to make it hard to forget). This way, even if a family member sees the paper and tries to log in secretly, they will not succeed.

The main challenge with paper storage is preventing loss. You could consider measures like writing them in a large notebook kept in a designated place, not carrying it with you (which increases the risk of loss), or making a copy and storing it in two or more locations.

2.2.(2) Password-protected Electronic Files

Storing account information electronically on a PC or smartphone creates the risk of that information being stolen if your device gets infected with malware. Assuming you take countermeasures against this, let's consider how to record account information electronically.

Storing a file with your account information in a "plain, readable state" is highly risky. It is preferable to encrypt the file so that it can only be read by entering a decryption password. To create a "password-protected electronic file," you can use methods like password-protected compressed archives, encryption software like GPG, or creat-

[1] This article excludes methods that use password management software. The reasons are concerns such as: Can the software developer be trusted? Even if the developer isn't malicious, could a software bug lead to a password leak? Could development stop, making the software suddenly unusable? While it's fine to trust and use specific software for yourself, I don't believe I can recommend it to others.

ing an encrypted disk image file. Note that to prevent the encrypted file from being cracked by a brute-force attack, the password for file encryption must be sufficiently strong. The word "password" might evoke the length of a single word, so using a longer "passphrase" is often recommended.

2.3.(3) Hash Functions

This method involves generating a password using a deterministic cryptographic hash function from a "memorized common master password" and "site-specific information." This is similar to, but distinct from, method (2).[3] SuperGenPass is a system that makes this easy to use with JavaScript or a bookmarklet, using the public "site domain name" as the "site-specific information."[4] You can try the mobile version to see how it works.

I have released my own password generation program, Passme, which implements this method. Since its public release in 2017, I have migrated from existing password management systems to Passme and have been using it securely through 2025. I currently manage over 100 site passwords with this tool.[5]

3.Two-Factor Authentication

No matter which method you use to manage your passwords, using it in conjunction with other authentication methods is an effective way to further increase security. Here, we'll discuss two-factor authentication (2FA) via mobile phone/smartphone, which is a type of multi-factor authentication. This method authenticates you based on not only something only you know (the password) but also by verifying that you possess something only you have (your phone).

For example, you register your mobile number, and after you enter your password, a confirmation code (a one-time password) is sent to your phone via text message (SMS). You can only log in after entering this code. The code is valid for that one-time use only, so the mechanism ensures that only someone who "possesses" the phone at that moment can log in. The main types of two-factor authentication are:

- Using SMS or voice calls to a mobile phone/smartphone.
- Using a small, dedicated device that displays confirmation codes (token).
- Using a mobile app.

If you set up 2FA with a mobile app, for instance, you won't be able to log in if that app becomes unusable (e.g., you lose your smartphone). Therefore, you need to take countermeasures, such as enabling 2FA via SMS as a backup or printing out backup codes. If you lose your smartphone, you can still receive SMS messages sent to your registered phone number by getting a new phone with the same number.

Note that if you cancel or change the phone number registered for 2FA, you will be unable to log in, so be careful. It's a good idea to make a list of which services have which of your phone numbers registered.

4.Notes

---

[2] You could also add another rule to this, like "start reading from the 6th character (ignore the first 5)." However, making the rule too complex increases the risk of forgetting it.

[3] The key difference between this method and (2) is that if an attacker obtains the site-specific information and tries to guess the master password, they cannot know if the generated password is correct or not without attempting to log into the site. In method (2), if the password-protected file is leaked, the attacker can verify offline whether a master password guess is correct, making a brute-force attack feasible. With this method, however, the attack is limited by the site's tolerance for incorrect password entries. Many systems lock an account after a certain number of failed attempts, making attacks that try tens of thousands of combinations in a short period generally impossible. However, if a password for one site is leaked, the attacker now "knows the correct answer," which creates a risk that the master password could be cracked via a brute-force attack.

[4] Since the "site-specific information" is public knowledge, there is a risk, as noted in footnote 3, that if a password for one site is leaked, the master password could be discovered through a brute-force attack. If the master password is found, passwords for other sites will also be revealed. For this reason, the author explains that it is necessary to set a sufficiently strong master password. Since the site-specific information is public, it doesn't fundamentally enhance the complexity of the master password.

[5] With this program, as long as the file containing the "site-specific information (password seeds)" is not leaked, a brute-force attack is not feasible even if a "password for one site" is known. To perform a brute-force attack, an attacker would need to illegally obtain both the file with the "site-specific information" and "a password for one site." In this respect, it should be more secure than SuperGenPass. However, if this file is lost, you will lose the passwords for all sites managed by this program, so you must manage the file in a way that ensures its security while also preventing its loss.