Security

3 Secure your device

3.1 Secure your computer

1. Anti-virus – Avira, AVG, Bitdefender
2. Anti-malware – Malwarebytes
3. Clean up regularly – CCleaner
4. Backups – options – external HD, NAS, cloud
5. Keep systems up-to-date
6. Use strong passwords – see below
7. Remove dodgy software – QuickTime, Adobe Flash, Java
8. Don't disable UAC (User Account Control)
9. Use non-privileged account
10. On PC, show extensions
11. Use a VM (Virtual Machine) – Virtual Box, VMWare
12. Use a VPN (Virtual Private Network)
13. Disable AutoPlay (typically a Windows problem)

3.2 Backing up

- Free backup programs – SyncBackFree
- External hard disk – 2TB ($85 JB Hi Fi, $79 MSY)
- Online – Dropbox (1TB, $10/month), Google (2TB, $12.50/month), Microsoft (1TB, $10/month – includes Office 365)
- NAS (Network Addressable Storage)

3.3 Secure your mobile devices

- Set password or PIN
- Keep OS and apps up-to-date
- Turn off Bluetooth and hotspot when not in use
- Enable the timeout lock
- Backup regularly

3.4 Passwords

Crackers use different dictionaries: English words, names, foreign words, phonetic patterns and so on for roots; two digits, dates, single symbols and so on for appendages. They run the dictionaries with various capitalizations and common substitutions: "$" for "s", "@" for "a", "1" for "l" and so on. This guessing strategy quickly breaks about two-thirds of all passwords.

- Needs to be long – minimum 16 characters
- Use a mixture of characters
- Do not based it on a dictionary word

- Think about a password based on an entire sentence
- Use a password generator and store it in a password manager
- Do not reuse passwords on different sites
- Use a password manager – any password you can remember is unsafe – LastPass, Dashlane, 1Password (not free)

The rules for creating passwords are simple: Use a random combination of numbers, symbols, and mixed-case letters; never reuse passwords; turn on 2FA, and use a password manager.

There's some disagreement on whether you should change passwords regularly. There's a strong case to be made for changing passwords every year or so, if only to avoid being innocently caught up in a database breach.

If you would rather remember passwords the old-fashioned way, then try the book *Moonwalking with Einstein*, by Joshua Foer, for some ideas.

3.4.1 25 Most popular passwords (2019)

1. 123456	2. password	3. 123456789	4. 12345678	5. 12345
6. 111111	7. 1234567	8. sunshine	9. qwerty	10. iloveyou
11. princess	12. admin	13. welcome	14. 666666	15. abc123
16. football	17. 123123	18. monkey	19. 654321	20. !@#$%^&*
21. charlie	22. aa123456	23. donald	24. password1	25. qwerty123

3.4.2 Creating secure passwords – Bruce Schneier

If you want your password to be hard to guess, you should choose something that this process will miss. My advice is to take a sentence and turn it into a password. Something like "This little piggy went to market" might become "tlpWENT2m". That nine-character password won't be in anyone's dictionary. Of course, don't use this one, because I've written about it. Choose your own sentence -- something personal.

Here are some examples:

WIw7,mstmsritt... = When I was seven, my sister threw my stuffed rabbit in the toilet.

Wow...doestcst = Wow, does that couch smell terrible.

Ltime@go-inag~faaa! = Long time ago in a galaxy not far away at all.

uTVM,TPw55:utvm,tpwstillsecure = Until this very moment, these passwords were still secure.

You get the idea. Combine a personally memorable sentence with some personally memorable tricks to modify that sentence into a password to create a lengthy password. Of course, the site has to accept all of those non-alpha-numeric characters and an arbitrarily long password. Otherwise, it's much harder.

Even better is to use random unmemorable alphanumeric passwords (with symbols, if the site will allow them), and a password manager like Password Safe to create and store them. Password Safe includes a random password generation function. Tell it how many characters you want -- twelve is

my default -- and it'll give you passwords like y.)v_|.7)7Bl, B3h4_[%}kgv), and QG6,FN4nFAm_. The program supports cut and paste, so you're not actually typing those characters very much. I'm recommending Password Safe for Windows because I wrote the first version, know the person currently in charge of the code, and trust its security. There are ports of PasswordSafe to other OSs, but I had nothing to do with those. There are also other password managers out there, if you want to shop around.

3.4.3 There's more to passwords than simply choosing a good one:

1. Never reuse a password you care about. Even if you choose a secure password, the site it's for could leak it because of its own incompetence. You don't want someone who gets your password for one application or site to be able to use it for another.
2. Don't bother updating your password regularly. Sites that require 90-day -- or whatever -- password upgrades do more harm than good. Unless you think your password might be compromised, don't change it.
3. Beware the "secret question." You don't want a backup system for when you forget your password to be easier to break than your password. Really, it's smart to use a password manager. Or to write your passwords down on a piece of paper and secure that piece of paper.
4. One more piece of advice: if a site offers two-factor authentication, seriously consider using it. It's almost certainly a security improvement.

3.5 Password managers

Reasons for using one:

3.5.1 Browser integration

Most password managers include browser extensions that automatically save credentials when you create a new account or sign in using those credentials for the first time. That browser integration also allows you to automatically enter credentials when you visit a matching website.

Contrast that approach with the inevitable friction of a manual list. You don't need to find a file and add a password to it to save a new or changed set of credentials, and you don't need to find and open that same file to copy and paste your password.

3.5.2 Password generation

Every password manager worth its salt includes a password generator capable of instantly producing a truly random, never-before-used-by-you password. If you don't like that password, you can click to generate another. You can then use that random password when creating a new account or changing credentials for an existing one.

Most password managers also allow you to customize the length and complexity of a generated password so you can deal with sites that have peculiar password rules.

3.5.3 Phishing protection

Integrating a password manager with a browser is superb protection against phishing sites. If you visit a site that has managed to perfectly duplicate your bank's login page and even mess with the URL display to make it look legit, you might be fooled. Your password manager, on the other hand,

won't enter your saved credentials, because the URL of the fake site doesn't match the legitimate domain associated with them.

That phishing protection is probably the most underrated feature of all. If you manage passwords manually, by copying and pasting from an encrypted personal file, you will paste your username and password into the respective fields on that well-designed fake page, because you don't realize it's fake.

3.5.4 Cross platform access

Password managers work across devices, including PCs, Macs, and mobile devices, with the option to sync your encrypted password database to the cloud. Access to that file and its contents can be secured with biometric authentication and 2FA.

By contrast, if you manage passwords in an encrypted file that's saved locally, you have to manually copy that file to other devices (or keep it in the cloud in a location under your personal control), and then make sure the contents of each copy stay in sync. More friction.

3.5.5 Surveillance protection

Password managers generally offer good protection against "shoulder surfing." An attacker who's able to watch you type, either live or with the help of a surveillance camera, can steal your login credentials with ease. Password managers never expose those details.

3.6 Additional protection techniques

- 2-factor authentication (2FA)
- Physical key – RSA, Yubi

3.7 Encryption

Consider the two prime numbers 7901 and 7919 (prime numbers are ones that you can divide by no other numbers than one and themselves). Suppose you multiply them together to get 62,568,019. That's a pretty simple operation, anyone can do it quickly with a calculator.

But what if you were given the number 62,568,019 and told to figure out the two numbers multiplied together to make that number (ie to factorise the number). You'd be there all day!

Now imagine multiplying two prime numbers, each about 40 digits long? How long would it take to factorise that number?

Answer? If you started at the time of the big bang, you would probably be still going (using current technology).

This product of two prime numbers forms the basis of an encryption key.

- A list of the first million prime numbers begins at the number 2 and ends with 15,485,863. That’s a long way short of 40 digits.
- The largest prime number known (discovered in 2017) contains 23 million digits.

3.7.1 Types of encryption

- Symmetric – single key used to both encrypt and decrypt
- Asymmetric – two keys are used, one to encrypt (public key) and the other to decrypt (private key).

Key size	Possible combinations
1-bit	2
2-bit	4
4-bit	16
8-bit	256
16-bit	65536
32-bit	4.2×10^{9}
56-bit (DES)	7.2×10^{16}
64-bit	1.8×10^{19}
128-bit (AES)	3.4×10^{38}
192-bit (AES)	6.2×10^{57}
256-bit (AES)	1.1×10^{77}

Key size	Time to crack
56-bit	399 seconds
128-bit	1.0×10^{18} years
192-bit	1.9×10^{37} years
256-bit	3.3×10^{56} years

Age of the universe: about 1.4×10^{10} years

3.8 Cracking Passwords

The general attack model is what's known as an offline password-guessing attack. In this scenario, the attacker gets a file of encrypted passwords from somewhere people want to authenticate to. His goal is to turn that encrypted file into unencrypted passwords he can use to authenticate himself. He does this by guessing passwords, and then seeing if they're correct. He can try guesses as fast as his computer will process them – and he can parallelize the attack – and gets immediate confirmation if he guesses correctly. Yes, there are ways to foil this attack, and that's why we can still have four-digit PINs on ATM cards, but it's the correct model for breaking passwords.

There are commercial programs that do password cracking, sold primarily to police departments. There are also hacker tools that do the same thing. And they're really good.

The efficiency of password cracking depends on two largely independent things: power and efficiency.

Power is simply computing power. As computers have become faster, they're able to test more passwords per second; one program advertises eight million per second. These crackers might run for days, on many machines simultaneously. For a high-profile police case, they might run for months.

Efficiency is the ability to guess passwords cleverly. It doesn't make sense to run through every eight-letter combination from "aaaaaaaa" to "zzzzzzzz" in order. That's 200 billion possible passwords, most of them very unlikely. Password crackers try the most common passwords first.

A typical password consists of a root plus an appendage. The root isn't necessarily a dictionary word, but it's usually something pronounceable. An appendage is either a suffix (90% of the time) or a prefix (10% of the time). One cracking program I saw started with a dictionary of about 1,000 common passwords, things like "letmein," "temp," "123456," and so on. Then it tested them each with about 100 common suffix appendages: "1," "4u," "69," "abc," "!," and so on. It recovered about a quarter of all passwords with just these 100,000 combinations.

Crackers use different dictionaries: English words, names, foreign words, phonetic patterns and so on for roots; two digits, dates, single symbols and so on for appendages. They run the dictionaries with various capitalizations and common substitutions: "$" for "s", "@" for "a", "1" for "l" and so on. This guessing strategy quickly breaks about two-thirds of all passwords.

Modern password crackers combine different words from their dictionaries:

What was remarkable about all three cracking sessions were the types of plains that got revealed. They included passcodes such as "k1araj0hns0n," "Sh1a-labe0uf," "Apr!l221973," "Qbesancon321," "DG091101%," "@Yourmom69," "ilovetofunot," "windermere2313," "tmdmmj17," and "BandGeek2014." Also included in the list: "all of the lights" (yes, spaces are allowed on many sites), "i hate hackers," "allineedislove," "ilovemySister31," "iloveyousomuch," "Philippians4:13," "Philippians4:6-7," and "qeadzcwrsfxv1331." "gonefishing1125" was another password Steube saw appear on his computer screen. Seconds after it was cracked, he noted, "You won't ever find it using brute force."

Last year, Ars Technica gave three experts a 16,000-entry encrypted password file, and asked them to break as many as possible. The winner got 90% of them, the loser 62% -- in a few hours. It's the same sort of thing we saw in 2012, 2007, and earlier. If there's any new news, it's that this kind of thing is getting easier faster than people think.

Pretty much anything that can be remembered can be cracked.

3.9 References

- **The Cracking Codebook**, Simon Singh
- **Introduction to Cryptography, The unbreakable Cipher, Mathematical Cryptosystems, The RSA Encryption Algorithm**
  Youtube: Eddie Woo, encryption series (Google: eddie woo encryption)
  https://www.youtube.com/playlist?list=PL5KkMZvBpo5CdoOxa3dqll2n6KsXqerYO
- https://boingboing.net/2014/02/25/choosing-a-secure-password.html
- https://open.buffer.com/creating-a-secure-password/
- Schneier on Security: https://www.schneier.com/
- https://medium.com/@stuartschechter/before-you-use-a-password-manager-9f5949ccf168