System Basics

Hi! This page is supposed to explain phones/laptops/desktops/technology in general. It's obviously completely incomplete.

Monitors, displays etc

Frame rate and Sync technologies

Standard for monitors is 60Hz. That means the monitor can display 60 frames per second. In games you might still want your PC to produce more frames than your monitor's refresh rate can physically display, because you will see a newer frame, which means less latency.12

Source: http://www.blurbusters.com/gsync/preview2/

144Hz monitor - 1000/144 = 6.944ms per frame draw

144Hz G-Sync worst result - 38ms

144Hz No sync worst - 26ms

G-Sync effectively adds about 12ms of input lag over having no sync on 144Hz. That's almost 2 whole frames, which was really noticeable for me when I was testing it on CS:GO.

Having no sync can cause screen tearing . However, while screen tearing is really noticeable on a 60Hz panel, on a 144Hz one it's much less noticeable and not a problem for me, so I play with no sync.

Keep in mind that Windows doesn't seem to like having two or more monitors connected with different refresh rates. It seems to slow down the other monitors to the frequency of the lower monitor when there's a redraw on it, which makes everything stutter/freeze for a little while.

Linux doesn't seem to have that problem.

Passwords

Unless you're using a password manager, your passwords are most likely insecure.

Take the LinkedIn database breach - over 96% passwords were bruteforced already. That means that less than 4% people are probably using decent passwords. Why probably? Because they could still be using one strong password for all sites, but the moment there's a database leak where passwords are in plain text is the moment when their system becomes useless.

Go read the XKCD about bit entropy. Sounds like good advice? It isn't. If we agree that safe password length is 15 characters, and you just used "correct horse battery staple" as a password, you've just effectively made a "4 character" password to someone who has a script that uses whole dictionary words to bruteforce it_(taking in account that the number of commonly used words is much larger than the number of characters in the alphabet(26))_. Even if you start adding numbers here and there, it's still not good as someone could use modified version of that script so it tries replacing letters with numbers, and you also end up having the problem the XKCD itself talks about. Think that any word or any sequence of characters counts as one character.

I've hopefully convinced you that remembering 15 completely different passwords is impossible and using the same or similar password on those 15 sites is stupid. What's the solution then? A password manager.

You should also expect certain things from the password manager.

- It needs to be open source - It will deal with all your passwords, the possibility to check what the program actually does is a need.
- It needs to work cross platform - It needs to be capable of working on all mainstream operating systems and devices.
- It needs to be self-hosted - Dealing with your database somewhere on the web is just another attack vector. Also, you need to rely only on yourself, not on anyone else. What if the site goes down?
- It needs to be secure - This bit is on you too, using autofill adds another attack vector, so I personally avoid it.

I've found KeePassXC to satisfy those needs(Android implementation). You only need to make one strong password for the database file and you're set. Remember to have multiple backups of the database file. Setting a timeout on the clipboard and database to 30~ seconds should be a good idea.

Keep in mind that if you follow all the advice here, you will lose access to your accounts if you lose access to your password database backup. It would be a good idea to encrypt it with 256-bit AES and keep it with someone you trust or somewhere safe offsite. Imagine some very unlikely event like tornado/flood passing through your house, you probably won't have access to anything in it afterwards, so plan ahead.

2 Factor Authentication

2FA is an awesome thing which adds another layer of security. Service you're using has to obviously support 2FA.

There is a lot of implementations since most 2FA is based on an open standard.

2FA works like following: Site gives you a secret key(sometimes in a form of QR code with a small button that reveals it in plain text), which you will supply into your 2FA app on your device(s), and the application on the device(s) will give you a generated code which works for a short time, you will need that code to login to the service after supplying your username and password.
'''Remember to backup the secret key.'''

Your time on the device needs to be correct otherwise you will receive invalid codes.

2FA does not only mean that you have an app with a secret key and timed codes, but can also be a device, or SMS/call codes...

You cannot trust your cell provider, **SO DO NOT** use SMS/call authentication if you can avoid it.

Using an open source authenticator app is highly recommended, it is a layer of your security, you should be able to look at the source code.

Personally I use WinAuth for desktop. It is an open source app made for Windows, with support not only for the traditional Google Authenticator algorithm but also services like Battle.net or Steam. It also works well on Linux with WINE.

For phone I use FreeOTP. Google Authenticator is not open source anymore, so I'd advise against using it.

Remember that 2FA is a **Two** Factor Authentication, having your password and 2FA system saved on the same computer effectively makes 2FA useless.

Backups

Hard drives fail, flash drives fail, DVDs/BDs deteriorate, solid state drives fail, accounts get hacked and the data wiped, data corruption happens... You need a good backup system.

Security through encryption

Wi-fi security

Use WPA2-PSK with AES encryption, disable WPS and use a password that you would not find in a dictionary nor one that can be bruteforced.

That means you shouldn't use WPA/WPA2, just pure WPA2, you should not use TKIP, only AES.

For WPA2 Enterprise networks, do not use MSCHAPv2 protocol.

Here's why

TLS

File encryption

VPN

Privacy

Why? I have nothing to hide

Privacy is a human right. People should have control over and access to the data they produce. If this right is not granted, people give significant power to those who have the access to and control over information. Even though they might be trusted now, no one can predict who will have the access and control in the future. Data about you is a leverage point for predicting and influencing your future actions. It is power over you.[1]

Instant messaging

Telegram

Also known as "Give me 5 digits to reset an account and gain control over it"

TOR

Operating systems and everything around them

There's tons of operating systems and tons of their variations. I'll list some of the most used ones.

Android

You can backup and restore android appdata without root.

```
adb backup -apk -shared -all -f backup-file.adb
adb restore backup-file.adb
```

With root apps should be in /data/app/APPNAME/base.apk and appdata should be in /data/data

Windows

Runs on x86(Windows 11 dropped support) and x64 and ARM CPU architectures.

You can either buy a retail or OEM license, retail is transferable between devices and OEM is tied to one device(unique motheboard and CPU) only. Retail home edition costs 119$.

The main editions that are still supported are 8.1(2023) and new builds of 10. The number in brackets is the year when the system becomes End Of Life, which means it will stop receiving support and should not connected to the internet at that point.

W10 has following editions:

- **Home** - Lacks features like Group Policy(though you can hack them in), meaning you don't have much control over updates
- **Pro** - More expensive than Home, better control over the OS, RAM support up to 512GB from 128GB, and some more features
- **Enterprise** - For businesses with over 250 computers
- **Enterprise LTSB** - Stable version, does not have any of the metro apps including Edge, Store,... with no way to install them.
- **Education** - Student version with basically all features of Enterprise

- ...and some more less significant ones

In addition to that, each version can be either N or KN(Korean version of N), which removes software like Windows media player and a bunch of other mostly useless software.

There are also Server versions

Linux

macOS

It is legal to only run the OS on Apple hardware. macOS running on something else than Apple HW is called a hackintosh, and since actually doing that is very annoying it turns me off the OS, so no more info besides this.

BSD

Partition Tables, BIOS and UEFI

Motherboard has firmware, which used to be IBM BIOS, but is nowadays UEFI. Almost everyone incorrectly refers to UEFI as BIOS, including motherboard manufacturers.

Most UEFI implementations have backwards compatibility with how the old BIOS firmwares used to boot, known as CSM (and sometimes as Legacy or simply BIOS). This mode should not be used and is going to be discontinued on Intel hardware post 2020.

BIOS booting works by loading the first 1024~ bytes (it varied) from a drive. This was never standardized. UEFI booting works by either directly booting an entry off a partition(almost never implemented), or by booting off entries in /boot partition(called ESP or EFI) which is standardized, should be FAT32 (FAT16 also works if your OS supports it, and Apple has added in their FS support into their UEFI implementation).

Using the EFI partition allows for multiple installed operating systems without deleting each other's bootloaders, as it used to be during the BIOS times.

MBR - Obsolete drive partitioning standard. It does not support more than 4 partitions per drive or drives larger than 2TB. Whole MBR is located on 512 bytes on the first sector of a drive, it contains the bootloader and information about partitions. Since this size is extremely small for any modern bootloader, it usually contains enough code to load a bootloader stored on an actual partition. For example if you have a Linux/Windows dual-boot, GRUB as a boot manager, and decide to wipe the

partition with Linux, you will not be able to boot properly since GRUB can't load.

GPT - Current partitioning standard, use it if you have the option.

BIOS - First thing that loads when you start up you PC. Obsolete, only supports MBR(or rather, Windows will refuse to work using BIOS+GPT or UEFI+MBR).

UEFI(previously **EFI**) - Successor of BIOS, supports GPT, usually has backwards compatibility to allow BIOS booting(and you can use GPT partitioning with that).

Motherboards with UEFI firmware have been the standard for years now, so unless you're working with hardware that's more than few years old it's likely using UEFI.

To update BIOS or UEFI go to your motherboard/laptop manufacturer's website. Looking in the downloads section should give you downloads and documentation on how to update. 1

If you want to read more about this, here's a handy site.

CPU architectures

ARM - weaker CPU architecture used in smartphones and such. It is not capable of running x86/x64 code

x86 - 32 bit architecture - obsolete, 32bit desktop CPUs were last made a decade ago. Cannot run x64/ARM code

x64 - 64 bit architecture - current standard for desktop PCs. Can run x86 code, cannot run ARM.

Creating a bootable flashdrive

Instructions here

Installing an OS

Windows

Components

Motherboard

PSU

Find some wattage calculator, get a PSU that has higher wattage than that and make sure it is not shit by getting a Tier 1 PSU from this post.

CPU

CPU cooler

GPU

RAM

How to check for failure

Under Arch Linux EFI install:

```
$ trizen -S memtest86-efi
# memtest86-efi --install
```

Storage Drives

HDD

SSD

S.M.A.R.T and Badblocks

S.M.A.R.T - is a monitoring system included in computer hard disk drives (HDDs) and solid-state drives (SSDs) that detects and reports on various indicators of drive reliability, with the intent of enabling the anticipation of hardware failures.

Windows: CrystalDiskInfo - open source tool

Linux: Arch Wiki

Badblocks is a program to test storage devices for bad blocks.

sudo badblocks -wsv /dev/$drive - Perform a **DESTRUCTIVE** test on the device. Tests with 4 patterns, so 4 passes which can take a while on an HDD. Useful for new drives or drives which have useless data on them.

sudo badblocks -nsv /dev/$drive - Perform a **NON-DESTRUCTIVE** test on the device. Single pass test.

Benchmarking

SSD

HDD

Media

Images

Today's widely used formats waste a crap ton of space - it is the reason why Dropbox made the lossless Lepton format for JPGs(saves about 22% space on average).

One of the formats you can convert your images to is WebP.

You can use a simple script and ImageMagick to mass convert files - It is not perfect (images will end up being named image.**jpg**.webp) but it does the job.

```
for file in *; do; convert $file $file.webp; done
```

Video

One of the formats you can convert your videos/GIFs/whatever is WebM.

You can use a simple script and ffmpeg to mass convert files - It is not perfect (images will end up being named image.''mp4''.webm, usually only runs on a single thread) but it does the job.

```
for file in *; do; ffmpeg -i $file $file.webm; done
```

Mouse Polling rate

Max mouse polling rate is 1000Hz, as in 1ms response time. Surprisingly I'm having a hard time finding mice that actually do have 1ms response time even though it says so on the spec sheet.

How to check your polling rate:

Windows: Download

Linux : #TODO

List of mice I've tried so far:

A4 Tech XL-750BK - says 1000Hz, actually is 1000Hz 3600 DPI laser mouse. I'm using this one.

SteelSeries Sensei RAW NaVi Edition - says 1000Hz, is actually 500Hz~.

Revision #1
Created 29 June 2021 11:51:11 by C0rn3j
Updated 29 June 2021 11:58:36 by C0rn3j