Misc.

ABC CSE Winter School

Keyboard remapping

This usually involves some software that is listening and, whenever a certain key is pressed, it intercepts that event and replaces it with another event corresponding to a different key.

ex. Remap Caps Lock to Ctrl or Escape.

You can also map keys to arbitrary commands of your choosing.

ex. Inserting some specific text, e.g. your long email address or your MIT ID number.

Keyboard remapping

There are even more complex modifications you can configure:

ex. Remapping sequences of keys, e.g. pressing shift five times toggles Caps Lock.

ex. Remapping on tap vs on hold, e.g. Caps Lock key is remapped to Esc if you quickly tap it, but is remapped to Ctrl if you hold it and use it as a modifier.

macOS - karabiner-elements, skhd or BetterTouchTool

Linux - xmodmap or Autokey

Windows - Builtin in Control Panel, AutoHotkey or SharpKeys

Backups

Any data you own that you haven’t backed up is data that could be gone at any moment, forever. Here we will cover some good backup basics and the pitfalls of some approaches.

The 3-2-1 rule is a general recommended strategy for backing up your data. It state that you should have: (Read more)

- at least 3 copies of your data
- 2 copies in different mediums
- 1 of the copies being offsite

Backups

A common pitfall when performing backups is blindly trusting whatever the system says it’s doing and not verifying that the data can be properly recovered. Toy Story 2 was almost lost and their backups were not working, happily they got everything back.

APIs

Most services online will have “APIs” (**A**pplication **P**rogramming **I**nterface) that let you programmatically access their data. For example, the US government has an API that lets you get weather forecasts, which you could use to easily get a weather forecast in your shell.

Most of these APIs have a similar format. They are structured URLs, often rooted at api.service.com, where the path and query parameters indicate what data you want to read or what action you want to perform.

APIs

For the US weather data for example, to get the forecast for a particular location, you issue GET request (with curl for example) to https://api.weather.gov/points/42.3604,-71.094

APIs

Some APIs require authentication, and this usually takes the form of some sort of secret token that you need to include with the request. You should read the documentation for the API to see what the particular service you are looking for uses, but “OAuth” is a protocol you will often see used. At its heart, OAuth is a way to give you tokens that can “act as you” on a given service, and can only be used for particular purposes. Keep in mind that these tokens are secret, and anyone who gains access to your token can do whatever the token allows under your account!

Common command-line patterns

Command-line tools vary a lot, and you will often want to check out their man pages before using them. They often share some common features though that can be good to be aware of:

- Most tools support some kind of *--help* flag to display brief usage instructions for the tool.
- Many tools that can cause irrevocable change support the notion of a “dry run” in which they only print what they would have done, but do not actually perform the change. Similarly, they often have an “interactive” flag that will prompt you for each destructive action.

Common command-line patterns

- You can usually use --version or -V to have the program print its own version (handy for reporting bugs!).
- Almost all tools have a --verbose or -v flag to produce more verbose output. You can usually include the flag multiple times (-vvv) to get more verbose output, which can be handy for debugging. Similarly, many tools have a --quiet flag for making it only print something on error.
- Possibly destructive tools are generally not recursive by default, but support a “recursive” flag (often -r) to make them recurse.
- The special argument -- makes a program stop processing flags and options (things starting with -) in what follows, letting you pass things that look like flags without them being interpreted as such: rm -- -r

Window managers

There are tools that manage how your windows behave. The one that you are 99% likely using (I am using them myself) is called “floating” window manager.

Alternative to that is “tiling” window manager.

With a tiling window manager, the screen is always filled by whatever windows are open, arranged according to some layout. If you have just one window, it takes up the full screen. If you then open another, the original window shrinks to make room for it

VPN

A VPN, in the best case, is really just a way for you to change your internet service provider as far as the internet is concerned. All your traffic will look like it’s coming from the VPN provider instead of your “real” location, and the network you are connected to will only see encrypted traffic.

While that may seem attractive, keep in mind that when you use a VPN, all you are really doing is shifting your trust from you current ISP to the VPN hosting company. Whatever your ISP could see, the VPN provider now sees instead. If you trust them more than your ISP, that is a win, but otherwise, it is not clear that you have gained much.

Markdown

There is a high chance that you will write some text over the course of your career. And often, you will want to mark up that text in simple ways. You want some text to be bold or italic, or you want to add headers, links, and code fragments. Instead of pulling out a heavy tool like Word or LaTeX, you may want to consider using the lightweight markup language Markdown.

Markdown

Emphasis (*italics*) is added by surrounding a word with *. Strong emphasis (**bold**) is added using **. Lines starting with # are headings (and the number of #s is the subheading level). Any line starting with - is a bullet list item, and any line starting with a number + . is a numbered list item. Backtick is used to show words in `code font`

Markdown is easy to get started with, and you can use it nearly everywhere.

ex. Telegram inherited some markdown features such as code font (not sure about kakao talk)

Booting & Live USBs

When your machine boots up, before the operating system is loaded, the BIOS/UEFI initializes the system.

For example, your computer may say something like “Press F9 to configure BIOS. Press F12 to enter boot menu.” during the boot process. You can configure all sorts of hardware-related settings in the BIOS menu. You can also enter the boot menu to boot from an alternate device instead of your hard drive.

Booting & Live USBs

Live USBs are USB flash drives containing an operating system. You can create one of these by downloading an operating system (e.g. a Linux distribution) and burning it to the flash drive.

If you break your existing operating system installation so that it no longer boots, you can use a live USB to recover data or fix the operating system.

Virtual machines

Virtual machines and similar tools like containers let you emulate a whole computer system, including the operating system. This can be useful for creating an isolated environment for testing, development, or exploration (e.g. running potentially malicious code).

Vagrant is a tool that lets you describe machine configurations (operating system, services, packages, etc.) in code, and then instantiate VMs with a simple vagrant up. Docker is conceptually similar but it uses containers instead. (container vs virtual machine)

Virtual machines

You can also rent virtual machines on the cloud, and it’s a nice way to get instant access to:

- A cheap always-on machine that has a public IP address, used to host services
- A machine with a lot of CPU, disk, RAM, and/or GPU
- Many more machines than you physically have access to

Popular services include Amazon AWS, Google Cloud, Microsoft Azure, DigitalOcean.

Notebook programming

Notebook programming environments can be really handy for doing certain types of interactive or exploratory development. Perhaps the most popular notebook programming environment today is Jupyter, for Python (and several other languages). Wolfram Mathematica is another notebook programming environment that’s great for doing math-oriented programming.

GitHub

GitHub is one of the most popular platforms for open-source software development. Many of the tools we’ve talked about in this class (vim), are hosted on GitHub. It’s easy to get started contributing to open-source to help improve the tools that you use every day.

GitHub

There are two primary ways in which people contribute to projects on GitHub:

- Creating an issue. This can be used to report bugs or request a new feature. Neither of these involves reading or writing code, so it can be pretty lightweight to do.
- Contribute code through a pull request. This is generally more involved than creating an issue. You can fork a repository on GitHub, clone your fork, create a new branch, make some changes (e.g. fix a bug or implement a feature), push the branch, and then create a pull request. After this, there will generally be some back-and-forth with the project maintainers, who will give you feedback on your patch. Finally, if all goes well, your patch will be merged into the upstream repository.