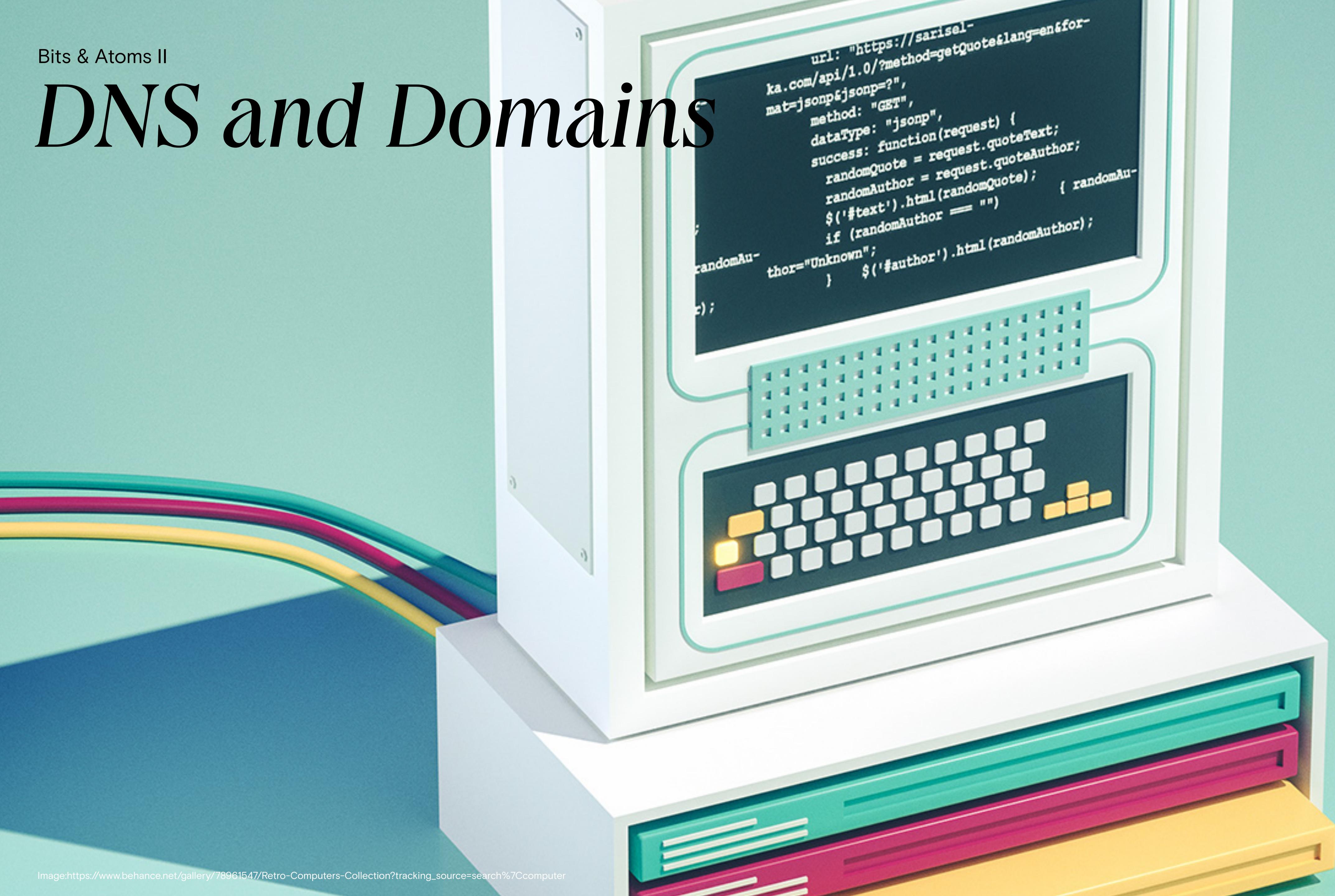


DNS and Domains

```
url: "https://sarisel-
ka.com/api/1.0/?method=getQuote&lang=en&for-
mat=jsonp&jsonp=?",
method: "GET",
dataType: "jsonp",
success: function(request) {
    randomQuote = request.quoteText;
    randomAuthor = request.quoteAuthor;
    $('#text').html(randomQuote);
    if (randomAuthor == "") { randomAu-
        thor="Unknown";
    } $('#author').html(randomAuthor);
}
);
```



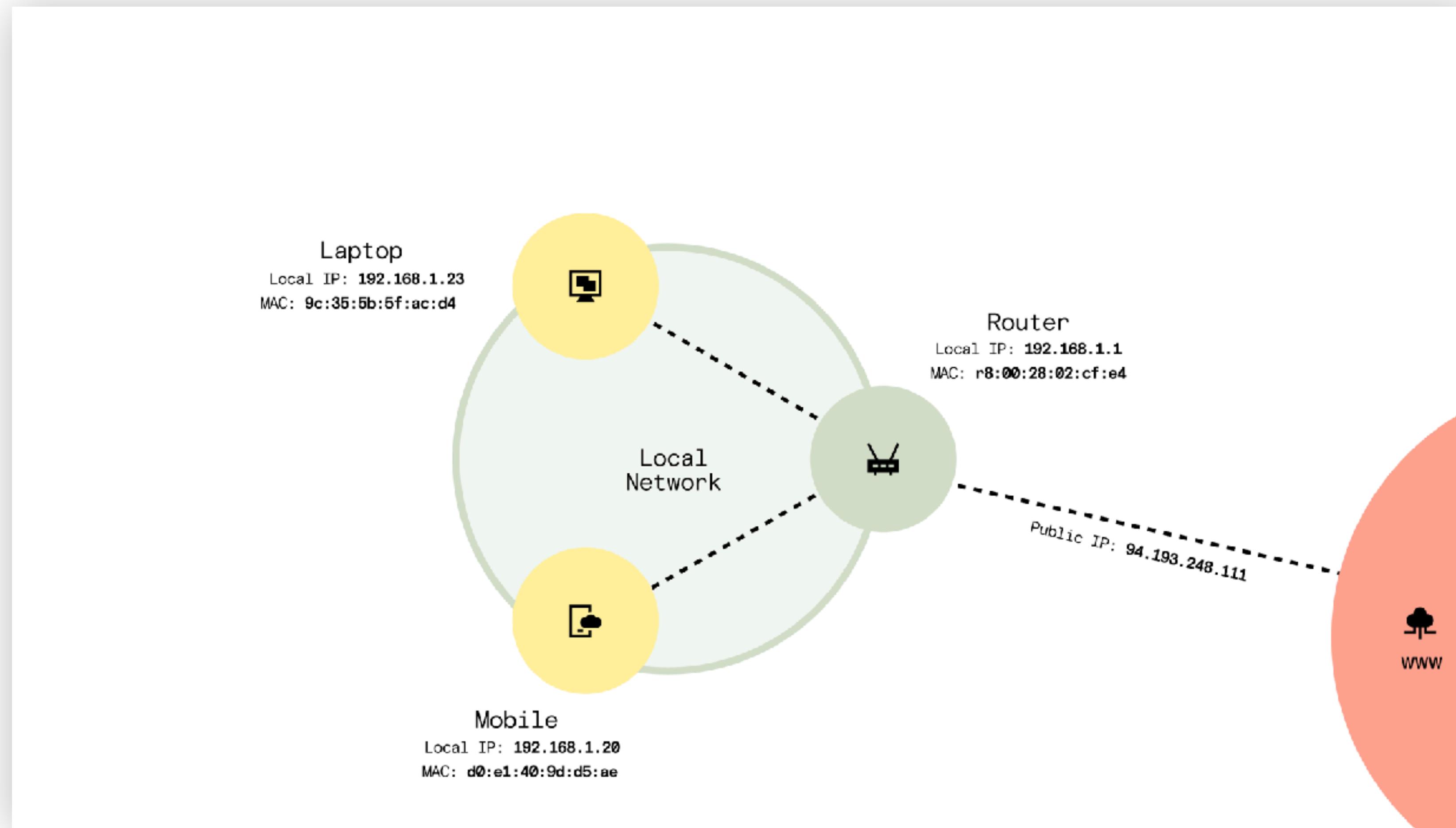
VPN vs. (*Encrypted*) Proxy

- Proxy only “protects” one app at a time (e.g. browser or BitTorrent client)
- VPN “protects” all your traffic

Internet Censorship

- States censor certain pages
 - Iran, China, etc.
- Different content depending on state
 - SRF, Netflix, etc.

NAT (Network Address Translation)



Encryption vs. Quantum Computers

- Every passphrase which would take thousands or millions of years to bruteforce (*trying out every possible combination*) today, could be easily found by using quantum computers

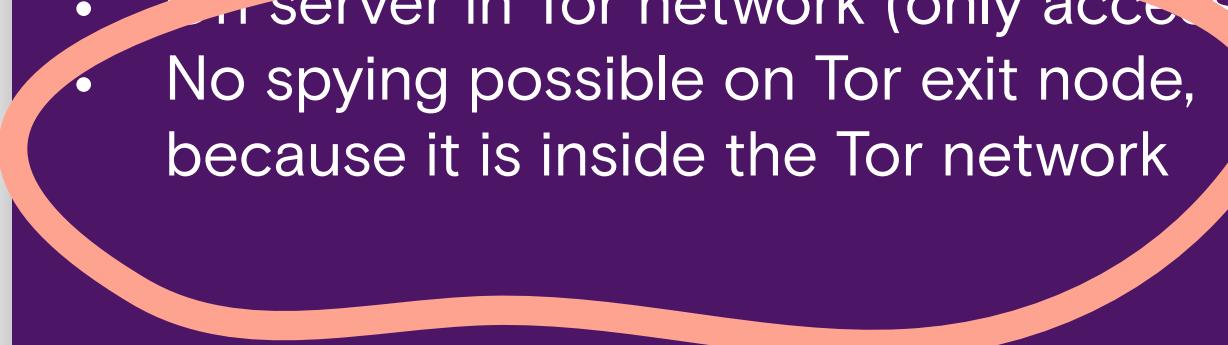
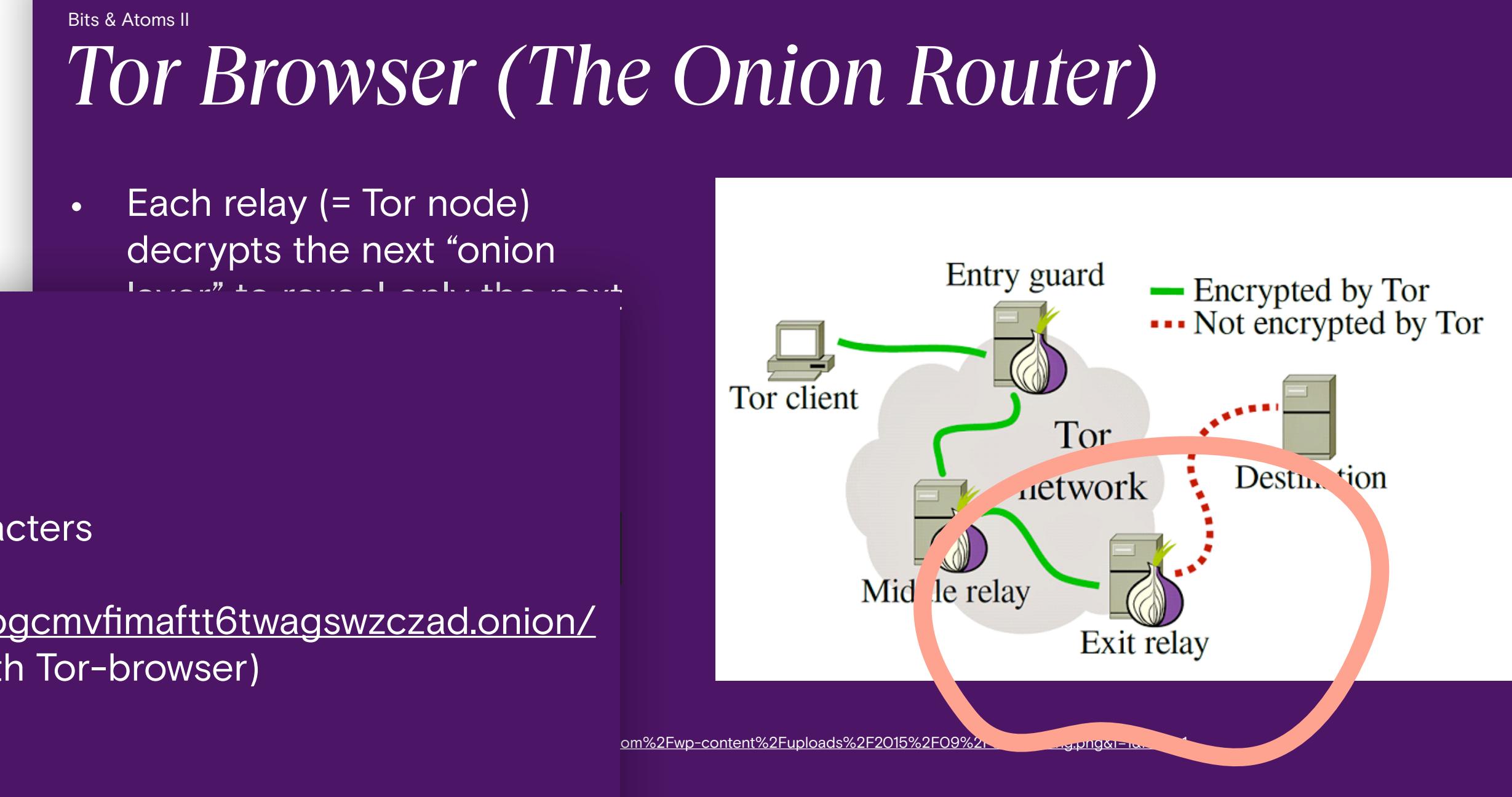
Spying on Exit Node (Tor)

- Because there is no exit node, you cannot spy on it...

Bits & Atoms II

.onion

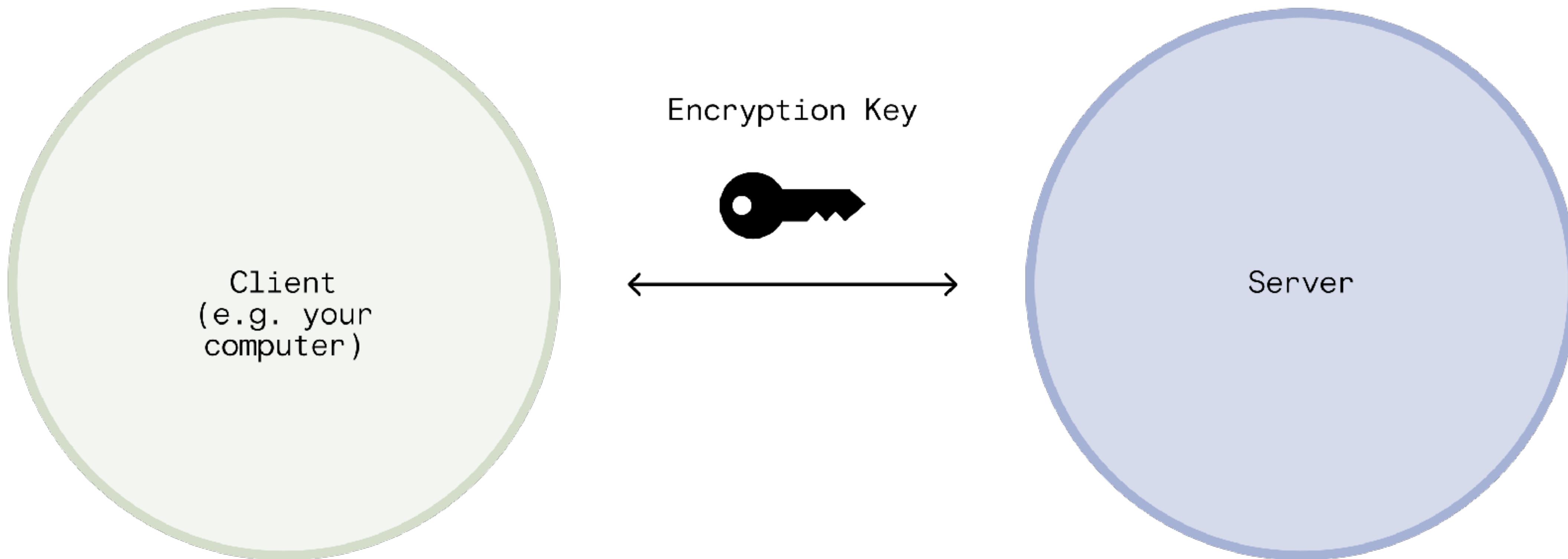
- 56 mostly random letters and numbers characters
- e.g. DuckDuckGo hidden service:
<https://duckduckgo.onion/>
- On server in Tor network (only accessible with Tor-browser)
- No spying possible on Tor exit node, because it is inside the Tor network

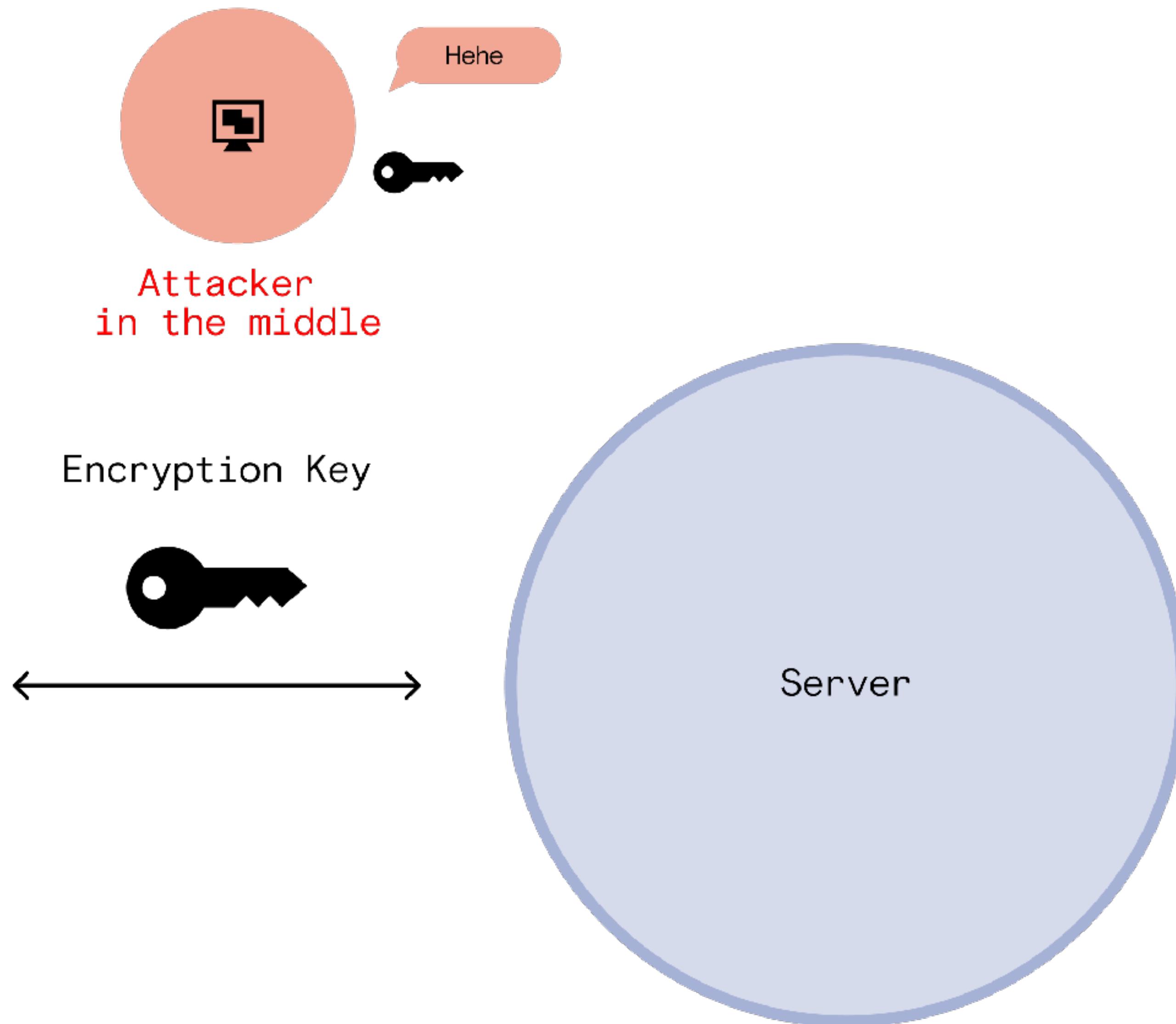
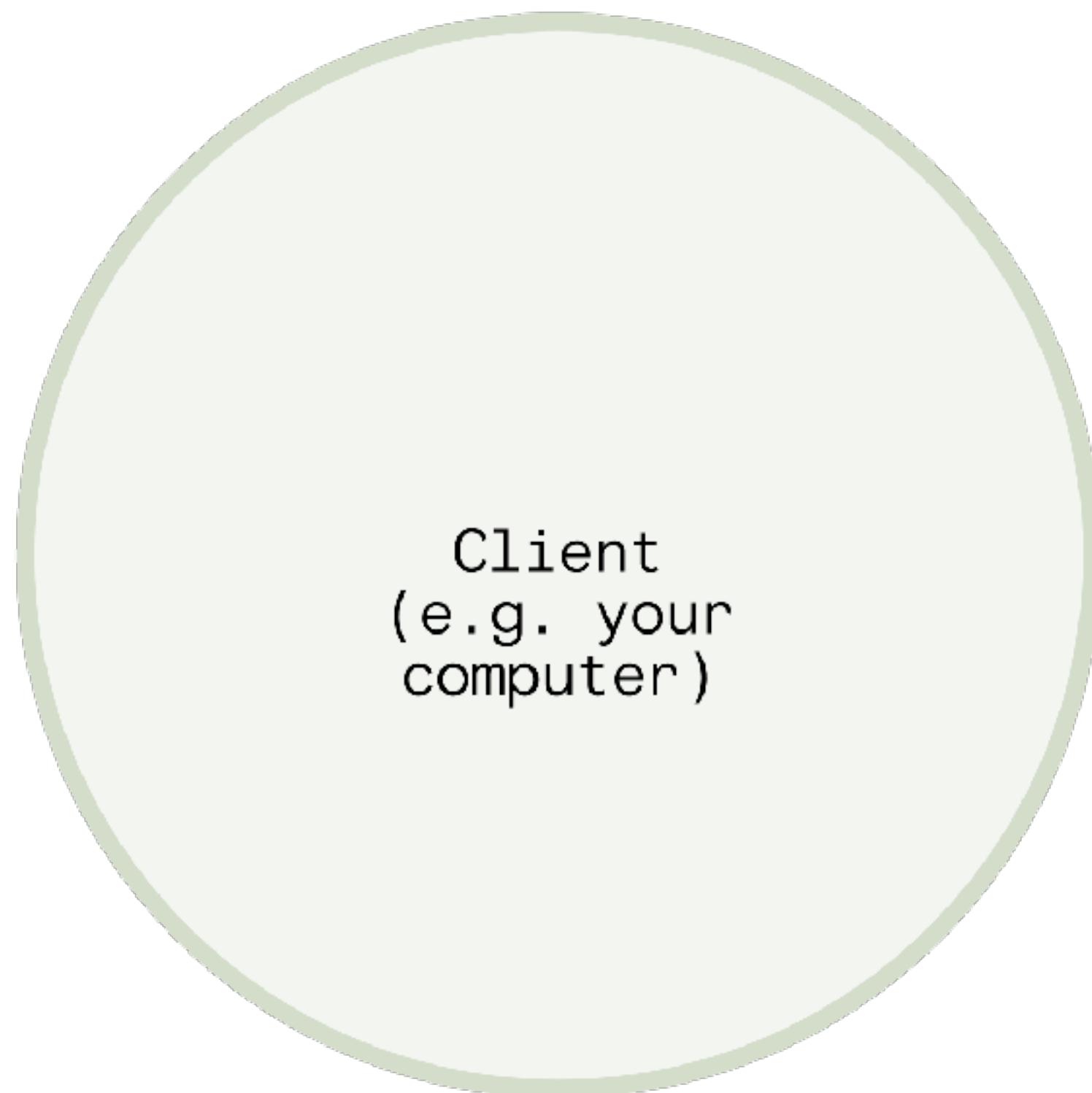
VPN and Evil Twin Discussion

- Problem of eavesdropping by attacker in the middle
- How to establish encryption in an unprotected environment?

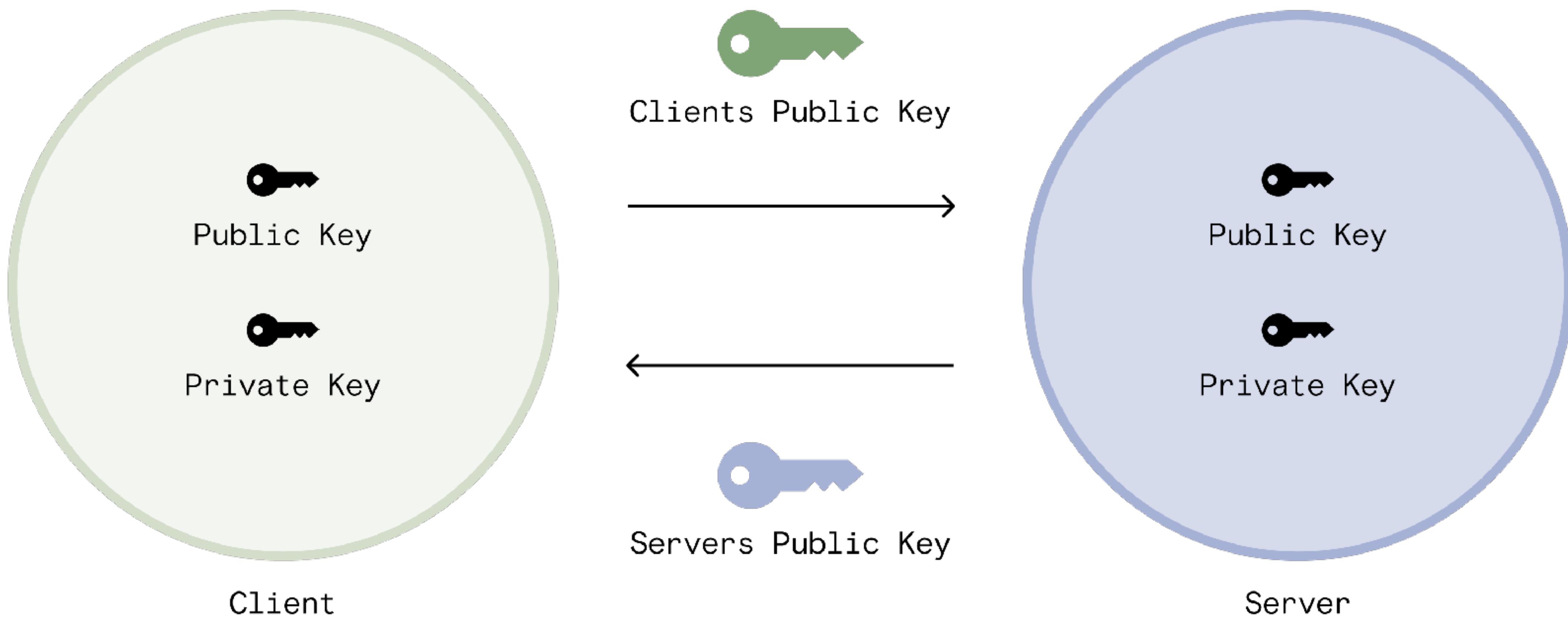
Encryption (Symmetric keys)



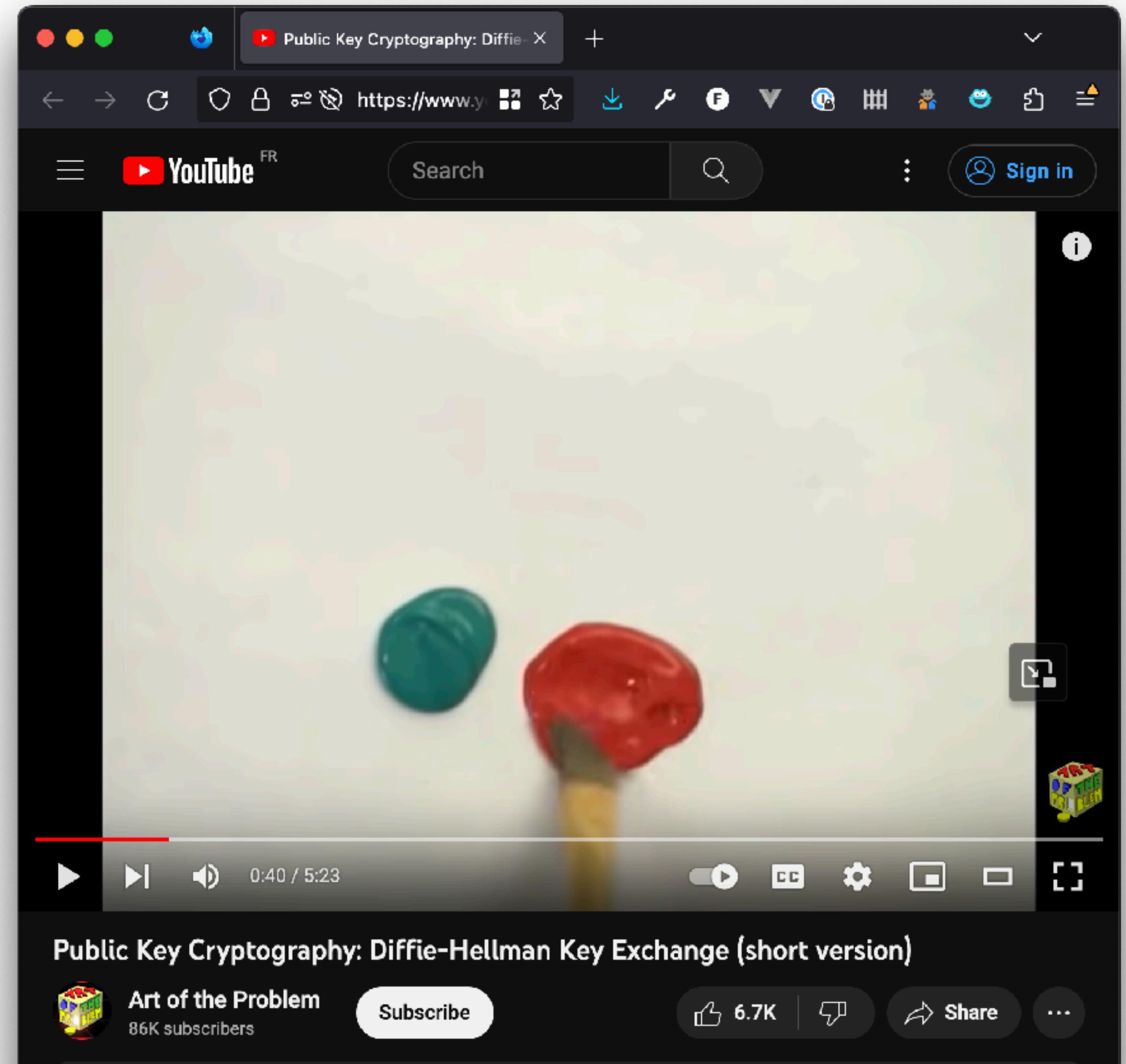
Key Exchange



Keypair (Public and Private Keys)

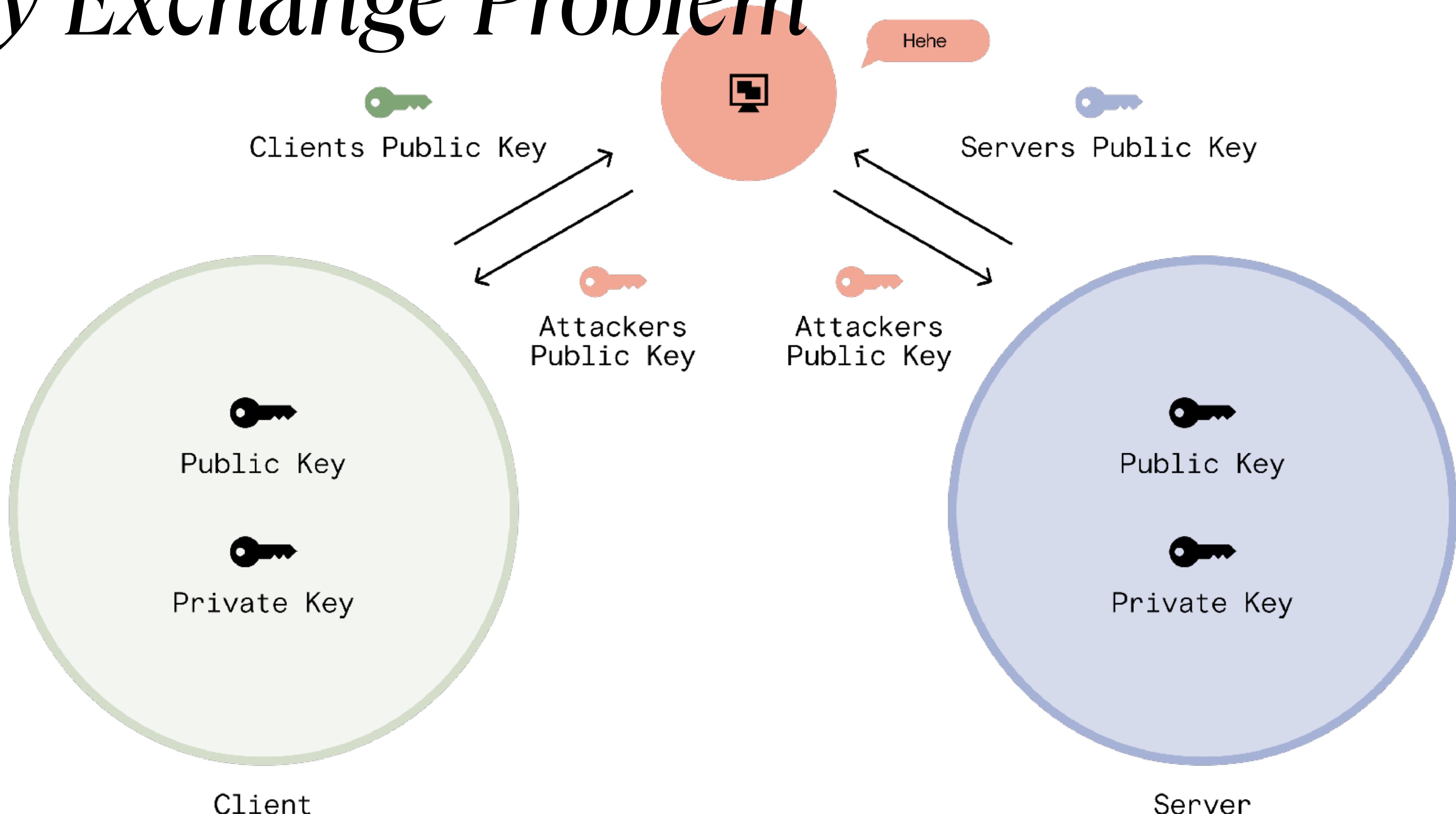


Keypair (Public and Private Keys)

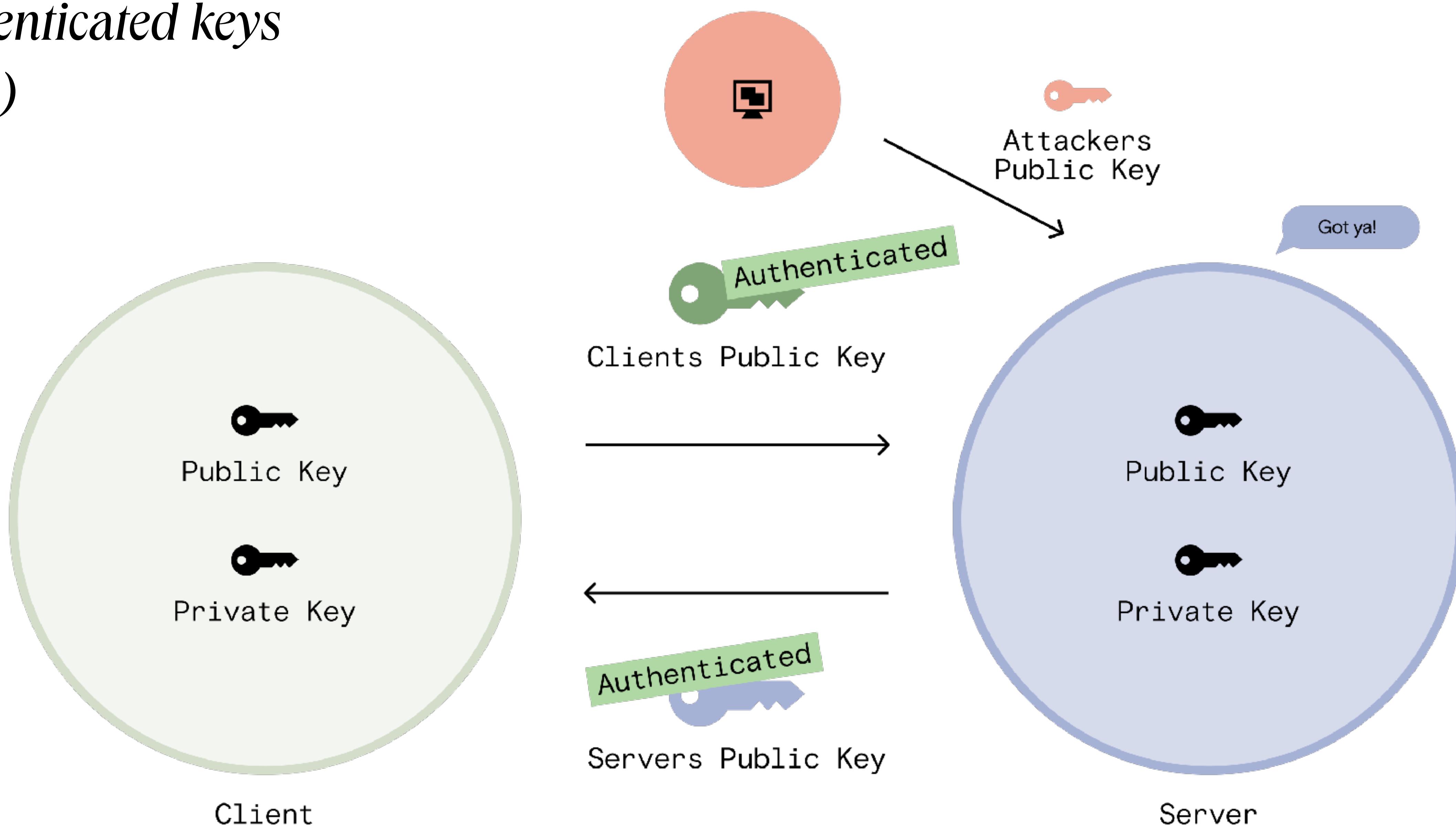


[https://www.youtube.com/watch?
v=3QnD2c4Xovk&ab_channel=ArtoftheProblem](https://www.youtube.com/watch?v=3QnD2c4Xovk&ab_channel=ArtoftheProblem)

Key Exchange Problem



Authenticated keys (RSA)



VPN and Evil Twin Discussion

The screenshot shows a section of the ProtonVPN website's FAQ page. At the top, there is a navigation bar with links for About, Features, Pricing, Blog, Support, For Business, Login, and Create free account. Below the navigation bar, the title "Frequently Asked Questions" is displayed in a large, bold, dark blue font. A specific question, "What encryption settings do you use for OpenVPN?", is highlighted with a light purple background and a small minus sign icon to its right, indicating it can be collapsed. The detailed answer follows, describing the secure nature of the OpenVPN protocol and the specific encryption settings used by ProtonVPN, including AES-256, RSA-4096, and HMAC SHA-384.

What encryption settings do you use for OpenVPN?

The OpenVPN protocol has proven itself secure for over 20 years. It is available in our Windows, Linux, Android, and iOS/iPadOS apps, using the following encryption settings:

The control channel

The control channel establishes a TLS connection between the VPN client and the VPN server. The whole process uses a symmetric key cipher, but the actual key exchange requires an asymmetric encryption system where a public key is used to encrypt the data, which can only be decrypted using a private key. Proton VPN uses AES-256 for its symmetric cipher, RSA-4096 to ensure a secure key exchange, and HMAC SHA-384 hash authentication to verify the TLS certificates. The encryption suite we use also includes a Diffie-Hellman key exchange to provide forward secrecy.

The data channel

Once a TLS connection is established, OpenVPN transfers your actual data over the data channel. This is encrypted with a symmetric cipher (Proton VPN uses AES-256) and verified with a hash function (HMAC SHA-384 in our case).

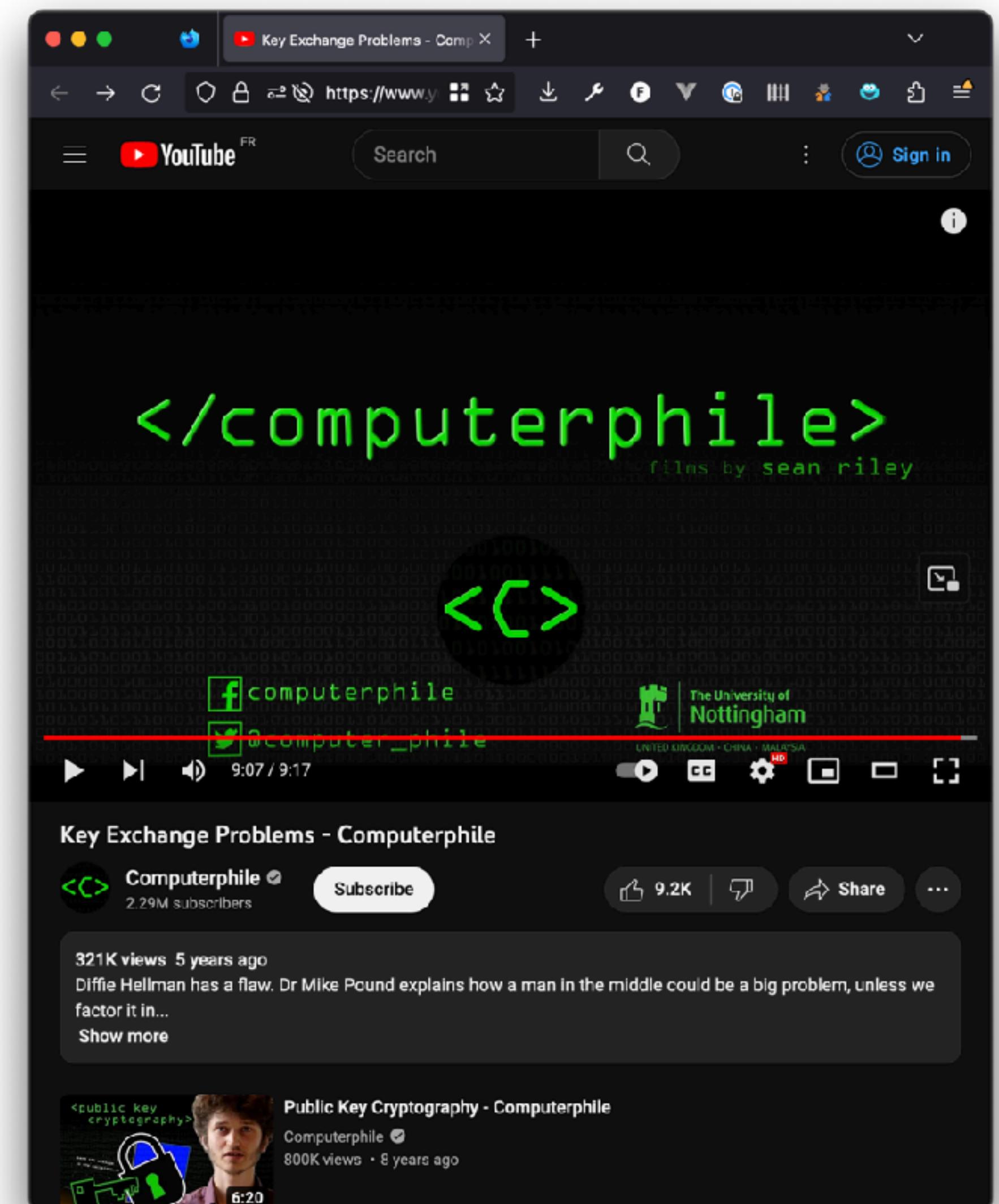
<https://protonvpn.com/secure-vpn/strong-protocols>

Further Reading

- [https://en.wikipedia.org/wiki/RSA_\(cryptosystem\)](https://en.wikipedia.org/wiki/RSA_(cryptosystem))
- https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange

Recommended Youtube Channel

- Computerphile Youtube Channel
 - TLS Handshake Explained
 - Key Exchange Problems
 - Diffi-Hellman Key Exchange



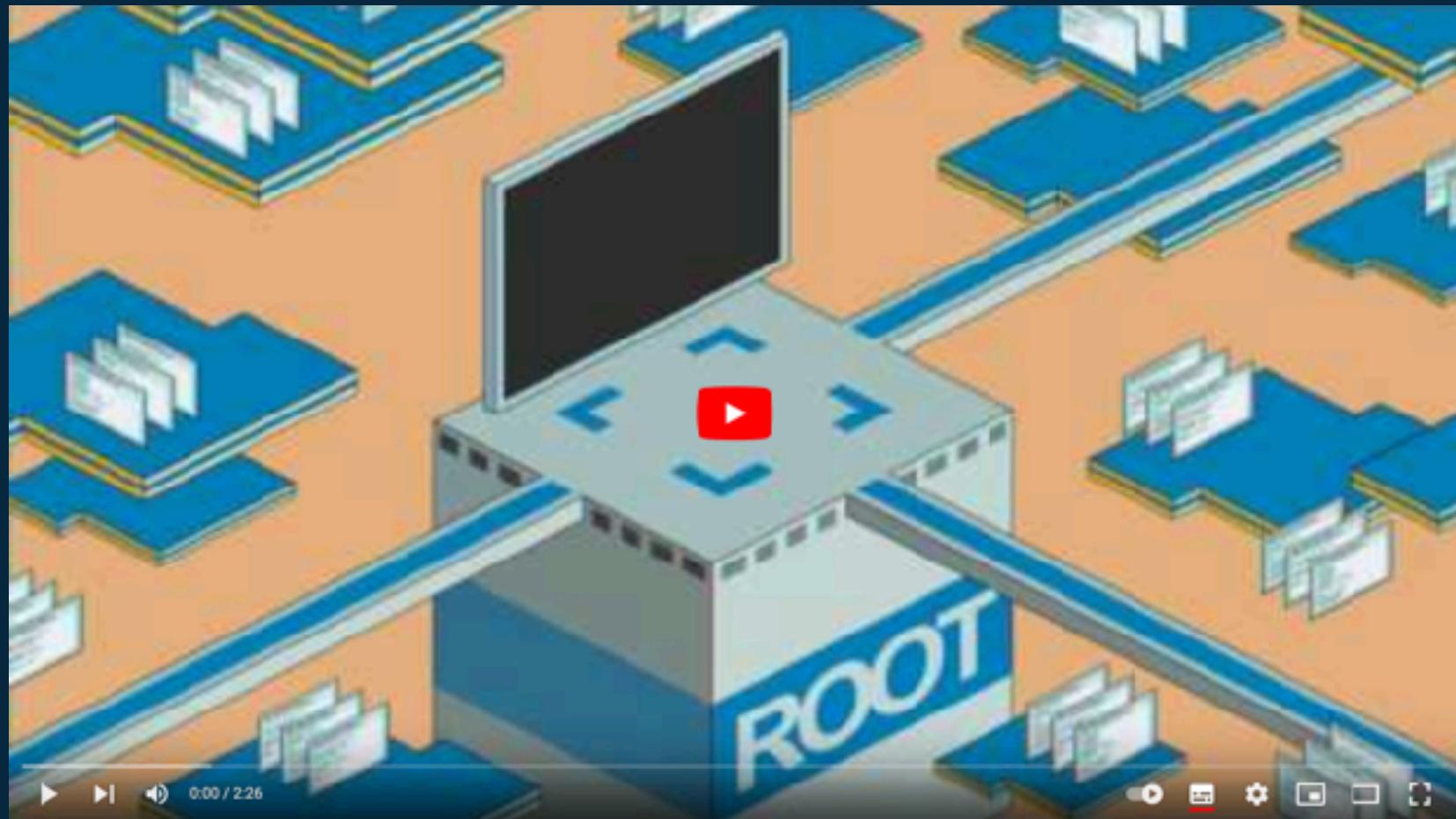
Domains

- duckduckgo.com
- zhdk.ch
- wired.com
- hallointer.net
- ...

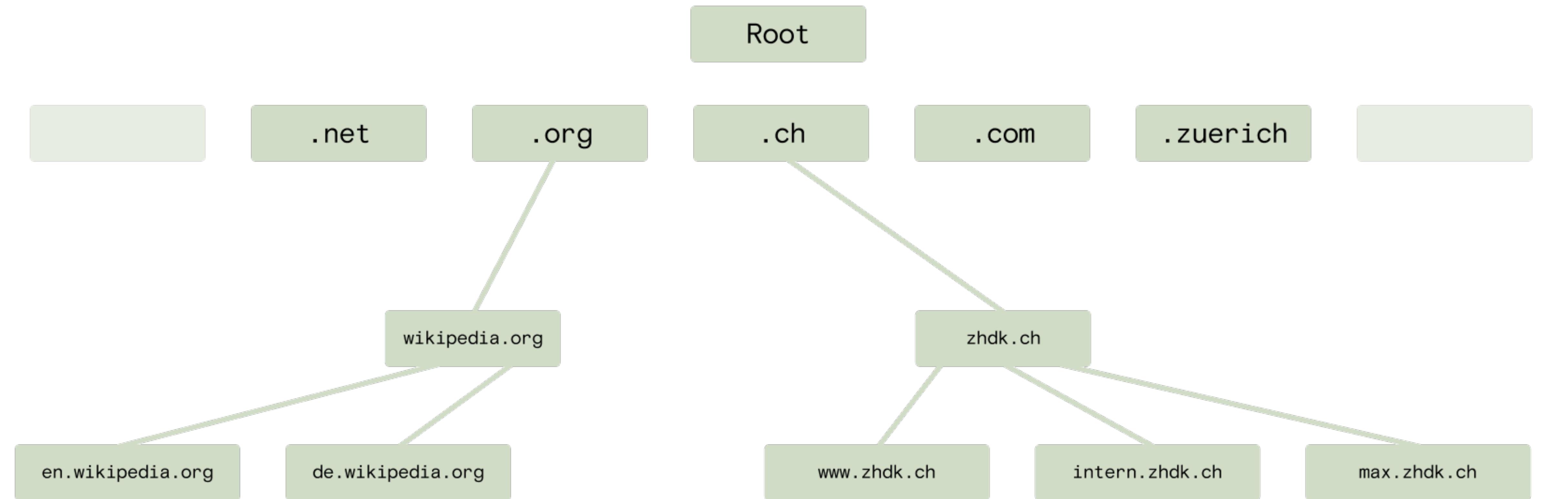
Domain Name System (DNS)

- duckduckgo.com → 40.114.177.156
- zhdk.ch → 195.176.247.145
- wired.com → 151.101.2.194
- localhost → 127.0.0.1

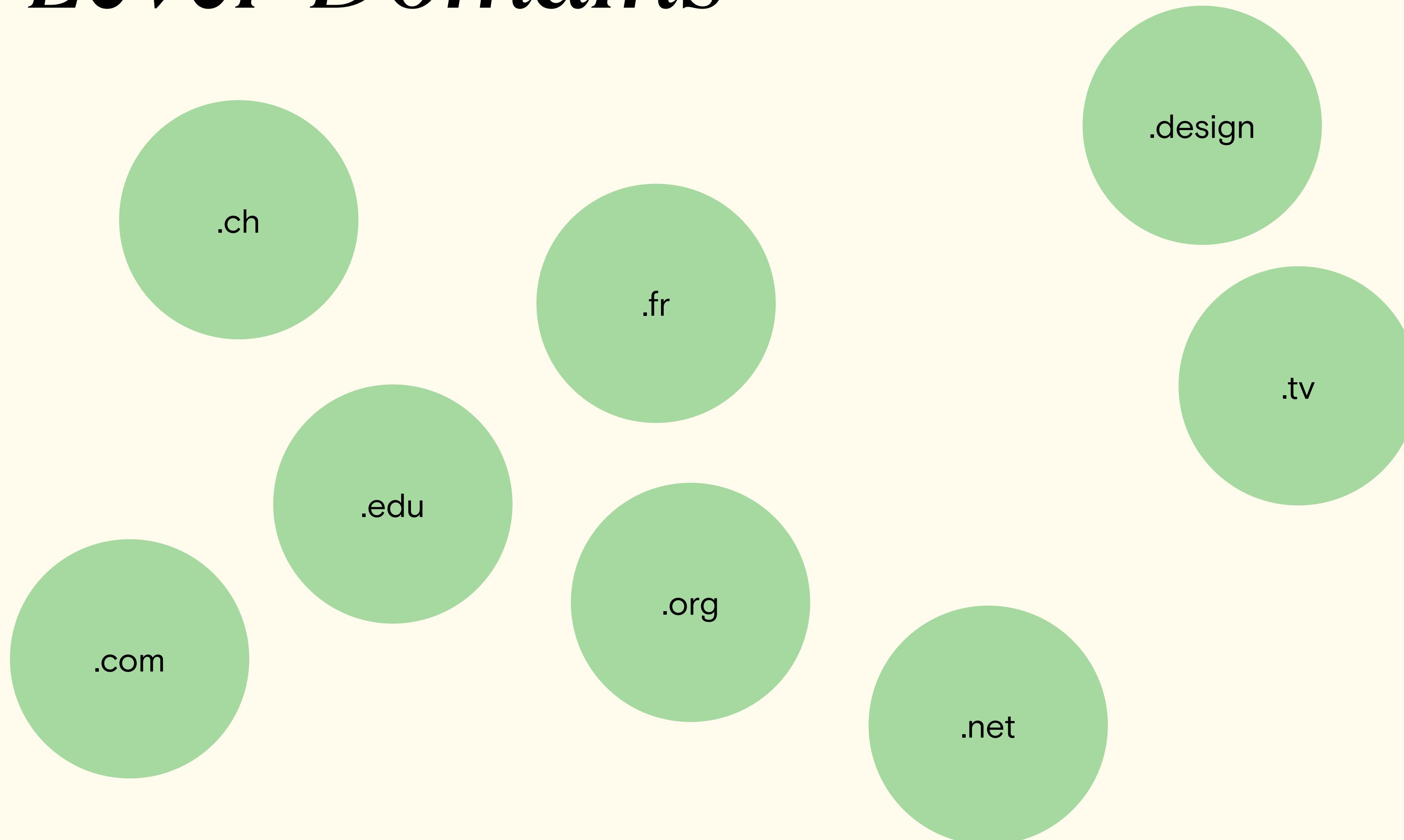
DNS



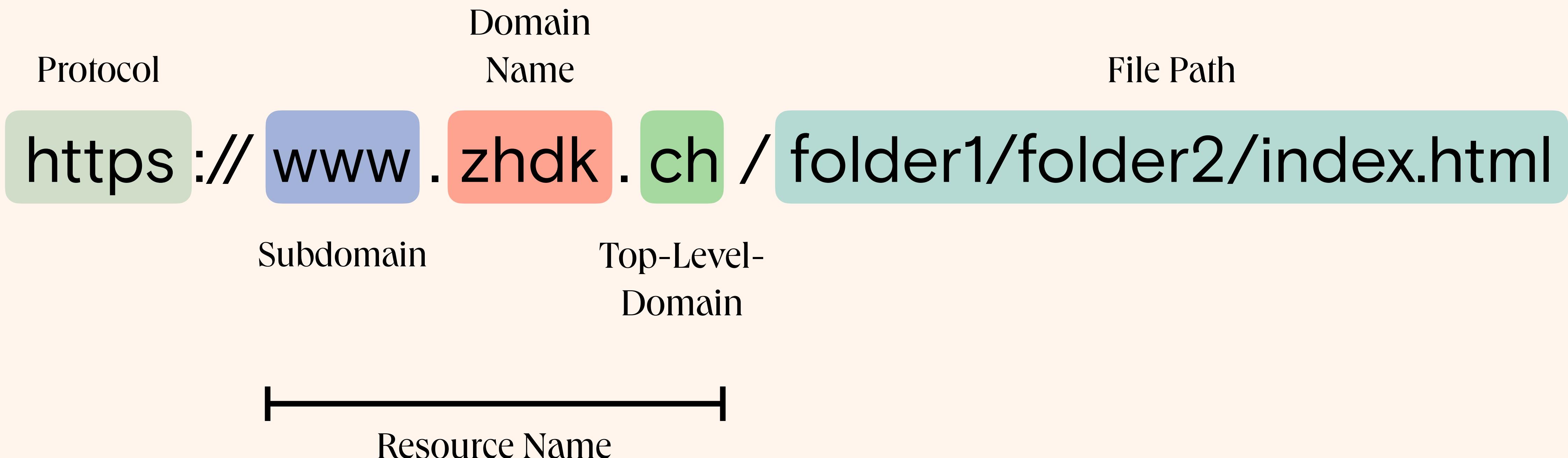
DNS Lookup Hierarchy



Top-Level-Domains



Uniform Resource Locator (URL)



Uniform Resource Locator (URL)

- <http://my-domain.to/path/to/my/file.zip>
- <https://www.chess.com/puzzles/rated>
- <https://goo.gl/maps/hiVtxgPvNzmxvbB56>
- <https://www.youtube.com/watch?v=2ZUxoi7YNgs&t=65>

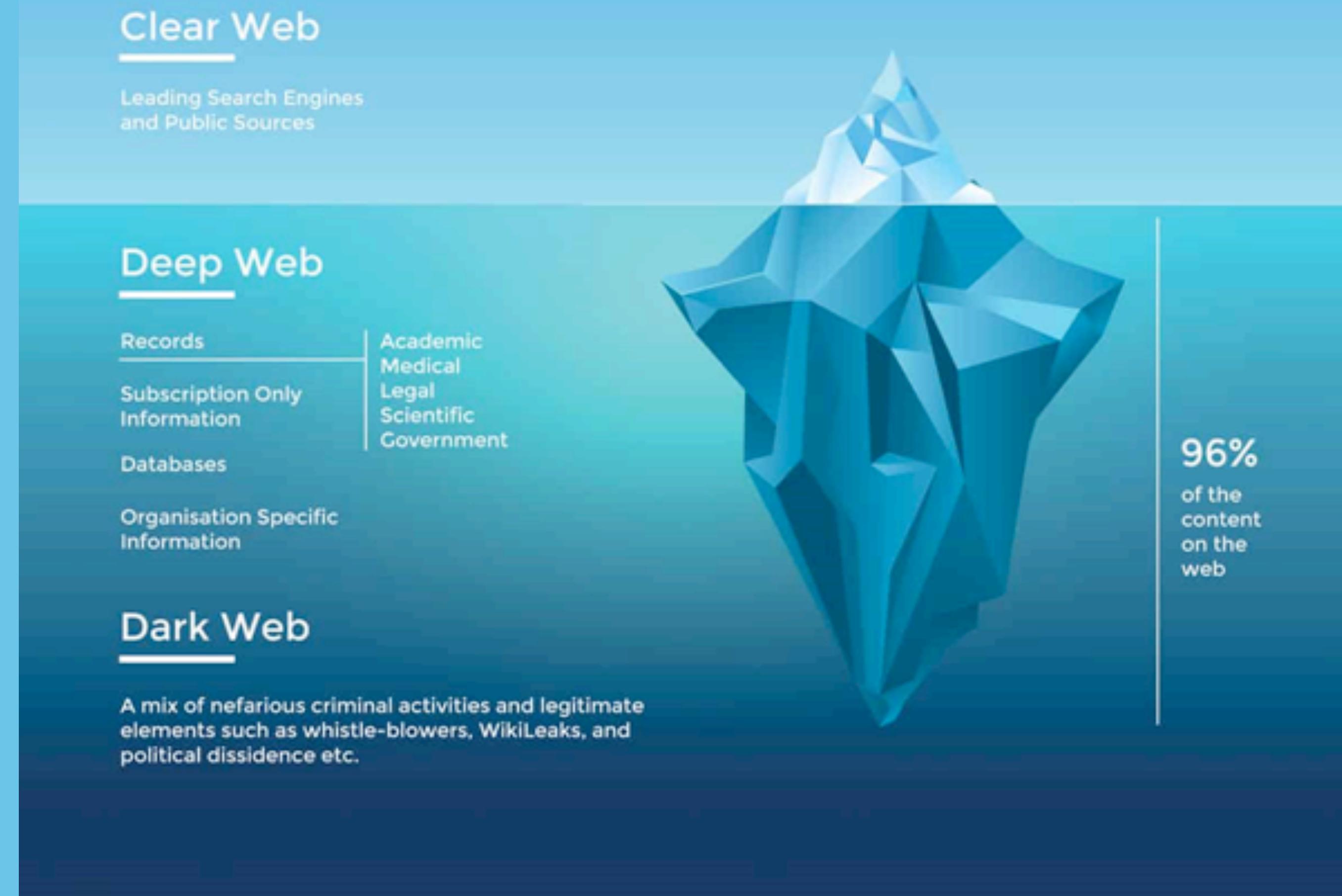
URL Query parameters

<https://www.sudpol.ch/programm/?jahr=2022&monat=3&kategorie=installation&post=pro-senectute-tanznachmittag-4-11>

URL Query parameters

`https://www.sudpol.ch/programm/
?jahr=2022
&monat=3
&kategorie=installation
&post=pro-senectute-tanznachmittag-4-11`

The Web



Languages

- HTML 5
- CSS
- JavaScript
- PHP
- Go
- Xml
- ...

HTML (*Hypertext Markup Language*)



```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Document</title>
8  </head>
9  <body>
10     An HTML document.
11  </body>
12  </html>
```

CSS (Cascading Style Sheet)

```
● ● ●

1
2 html {
3   font-family: var(--sans-font);
4   scroll-behavior: smooth;
5 }
6
7 body {
8   color: var(--text);
9   background: var(--bg);
10  font-size: 1.15rem;
11  line-height: 1.5;
12  display: grid;
13  grid-template-columns:
14    1fr min(45rem, 90%) 1fr;
15  margin: 0;
16 }
17
18 body > * {
19   grid-column: 2;
20 }
21
22 body > header {
```

JavaScript

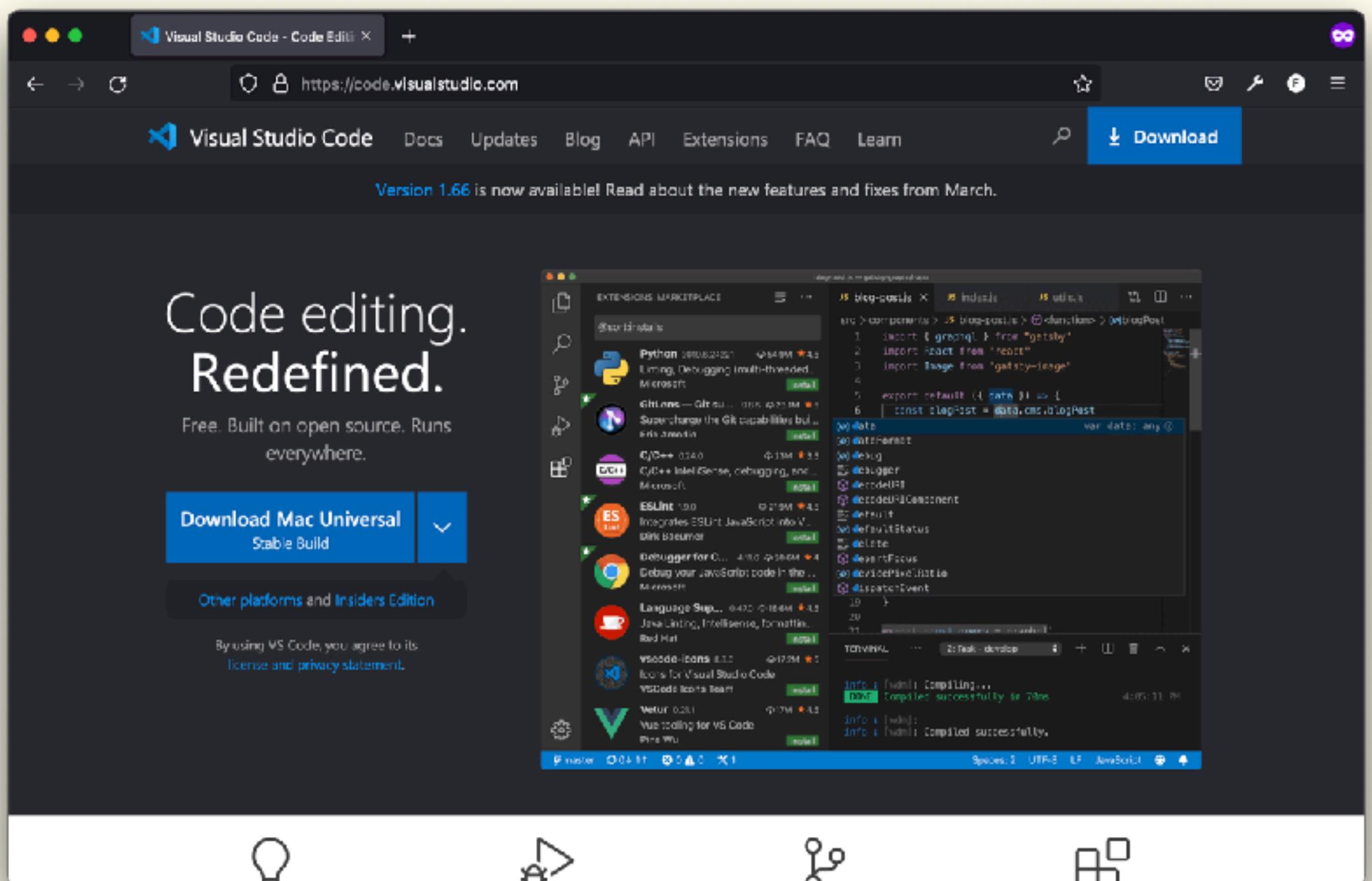
```
● ● ●

1  function registerPartials(directory, type, component) {
2      var files = fs.readdirSync(directory);
3
4      files.forEach(function (file) {
5          var nextComponent = component;
6          if (file == 'atoms' || file == 'molecules' || file == 'organisms') {
7              type = file;
8          } else if (!file.includes('.')) && file != 'views') {
9              nextComponent = file;
10         }
11
12         if (fs.statSync(directory + '/' + file).isDirectory()) {
13             registerPartials(directory + '/' + file, type, nextComponent);
14         } else {
15             var matches = /^([^.]+).hbs$/.exec(file);
16             if (matches) {
17                 var name = `views/${type}/${component}/${matches[1]}`;
18                 var template = fs.readFileSync(directory + '/' + file, 'utf8');
19                 hbs.registerPartial(name, template);
20             }
21         }
22     });
23 }
```

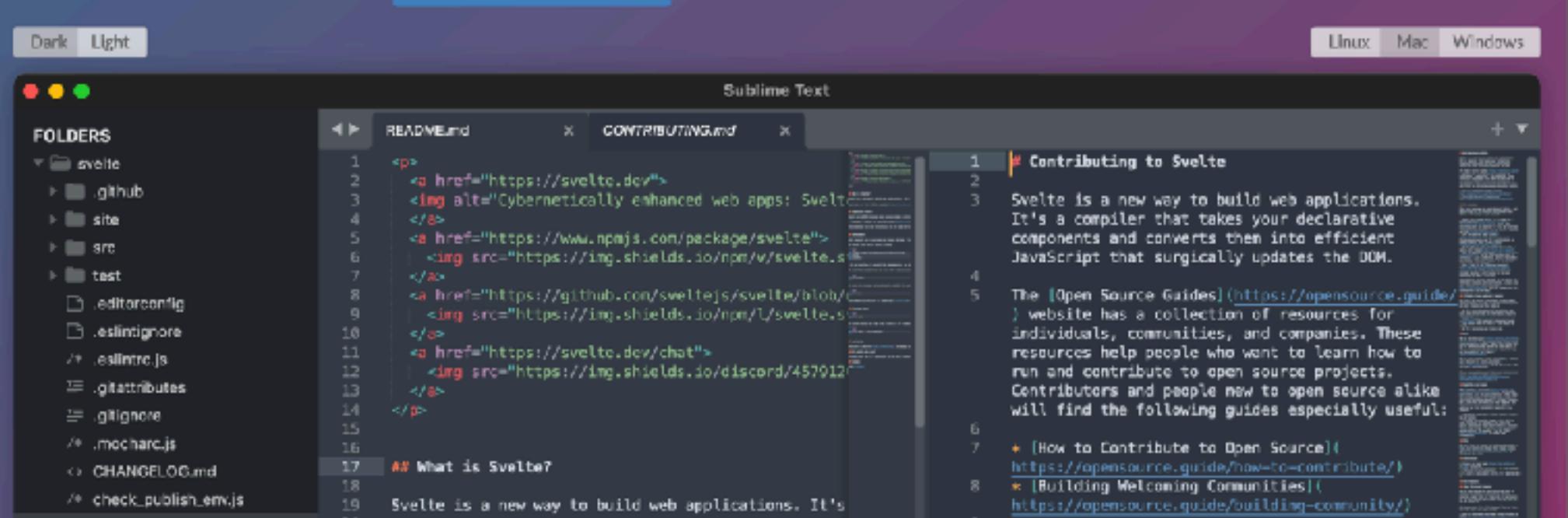
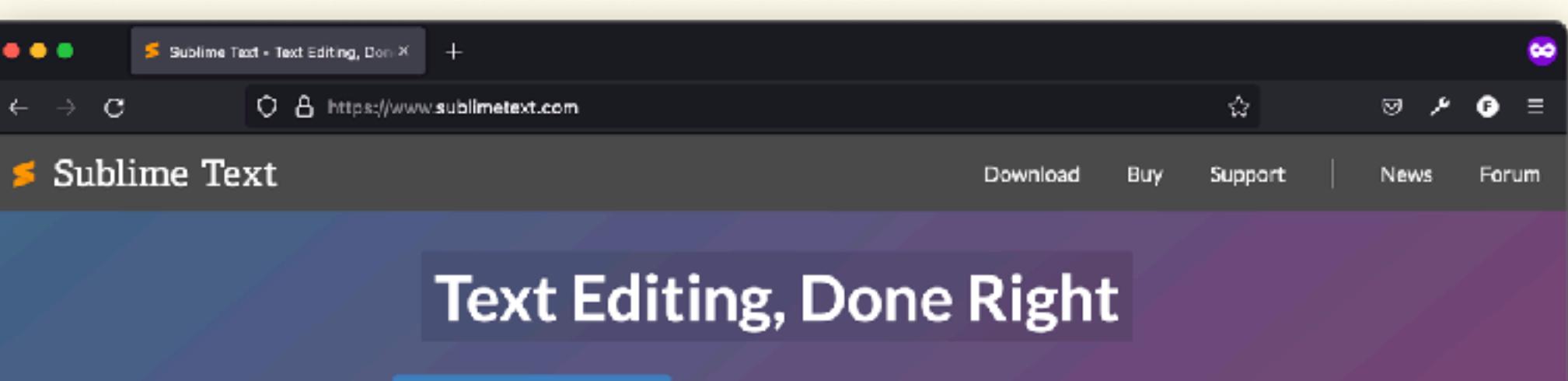
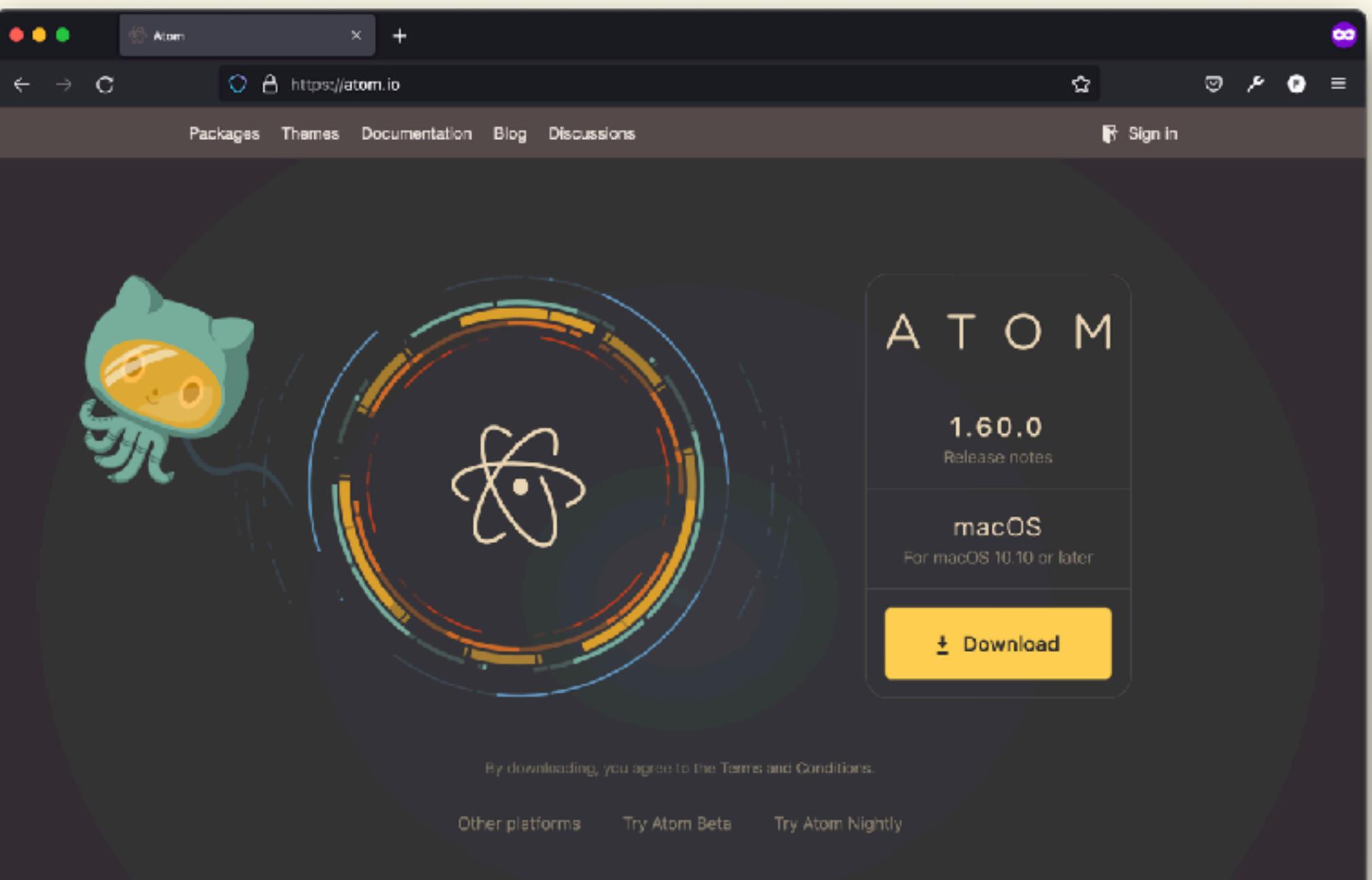
Tools

- Code-Editor (VS Code, Atom etc.)
- Browser (Firefox, Brave, Chrome, Safari, etc....)
 - Developer Tools
- Git Version Control (Github, Gitlab, Bitbucket...)
- Forums and Web-Resources (Stackoverflow, Mozilla Dev Network, W3C etc.)

Code-Editor



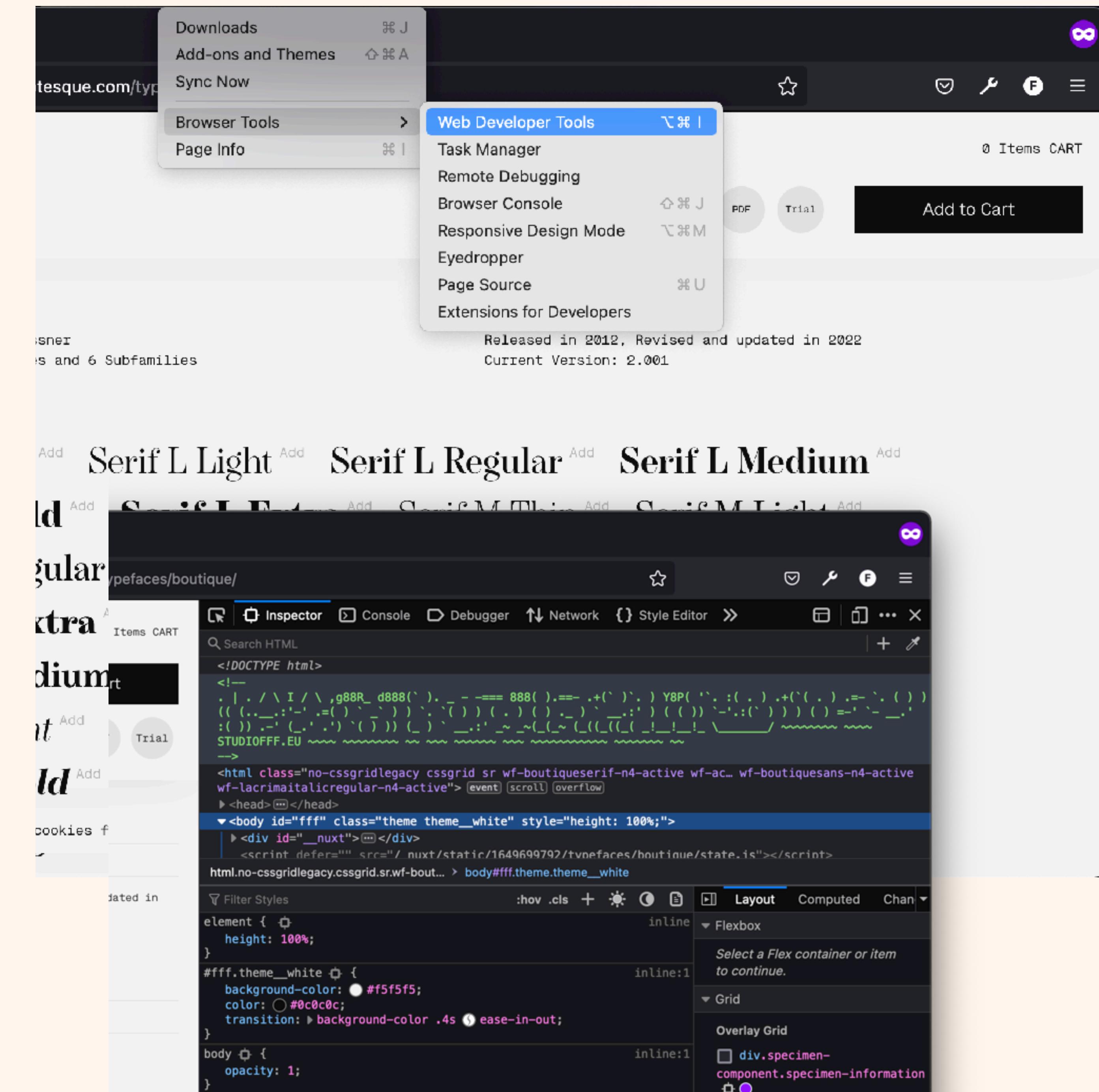
<https://code.visualstudio.com/>



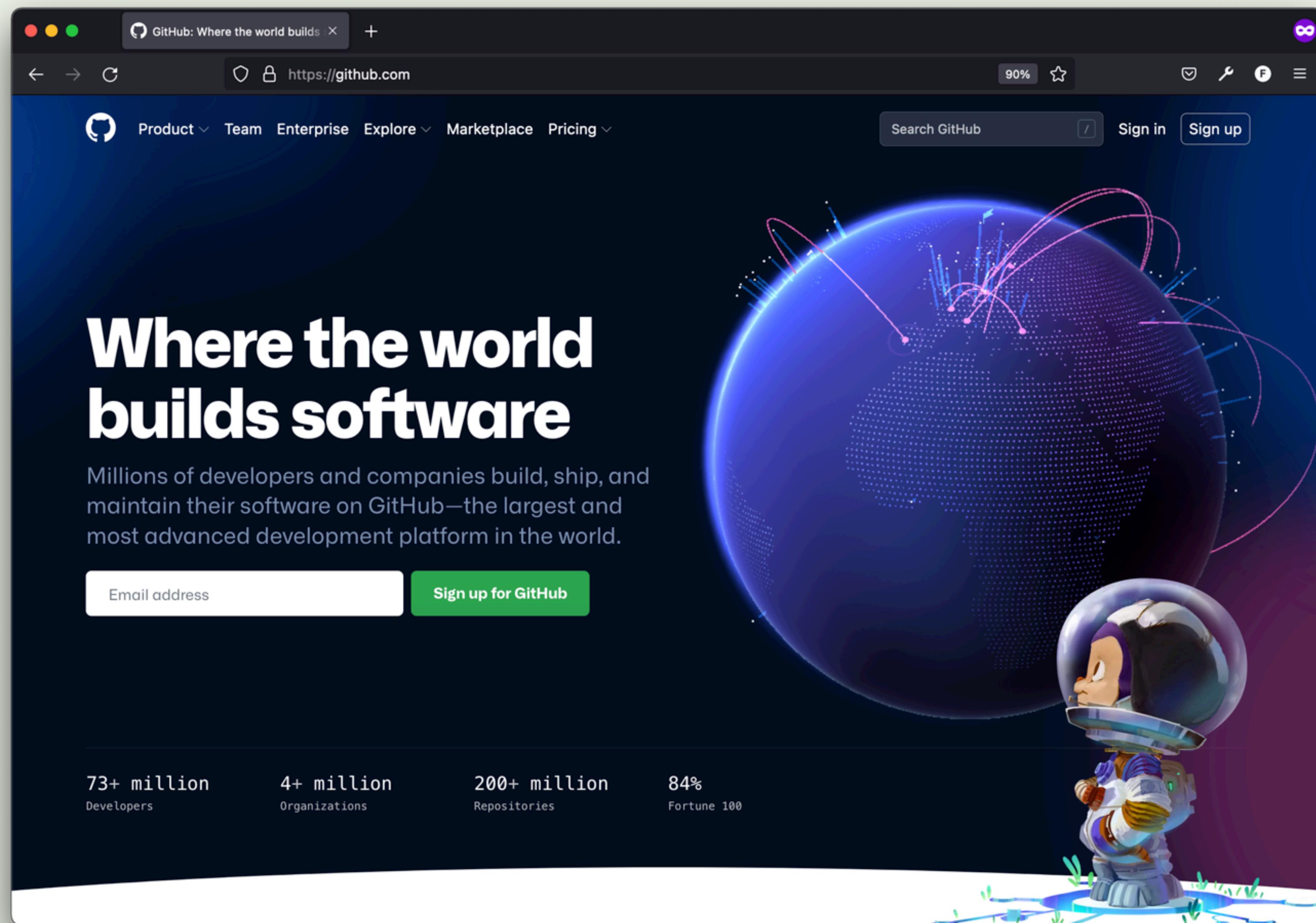
Developer Tools

Every browser is a bit different.
 Look for terms like «Inspector»,
 «Web Developer Tools» or
 «Console».

Shortcut:
 Mac: Option + Command + I
 Windows: F12



Git



Help

- stackoverflow.com
- Mozilla Dev Network
- caniuse.com
- GitHub issues
- Search engines
- Validation W3C
- Framework/Library Documentations
 - e.g.: nuxt.js
 - e.g.: lodash

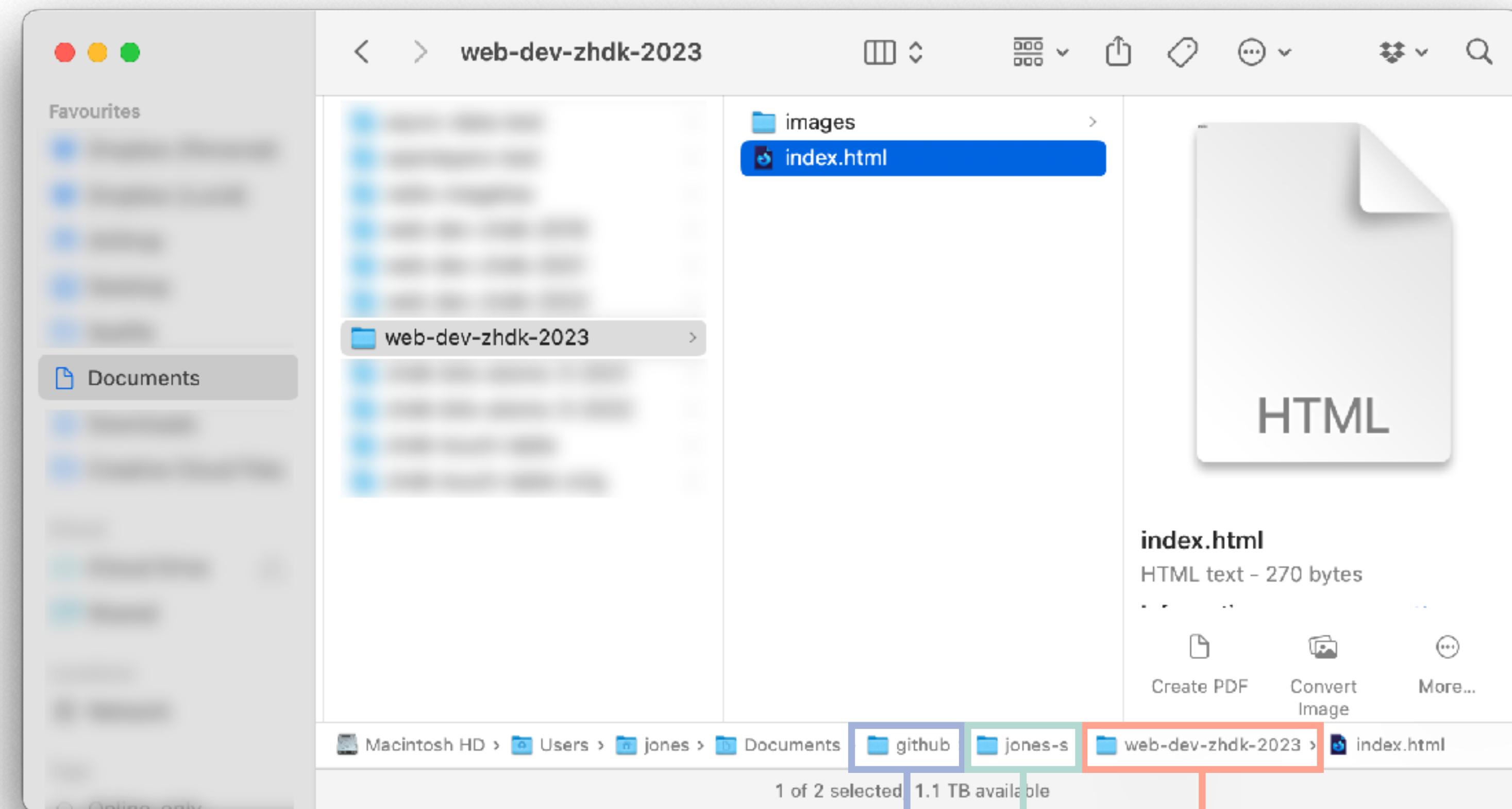
Hands-On

- Download code editor 
- Open dev tools in browser of your choice
- Create your first `.html` file.

Folder Structure and File Names

- Don't use uppercase letters
- Don't use special characters
- Don't use spaces. Use dashes or underscores
- Keep everything in one place

Example



All Github
projects

Project
folder

Github user

Tags and Attributes

Tag

```
<html>...</html>
```

Opening tag

Class attribute

Closing tag

```
<p data-attribute="content" class="paragraph">...</p>
```

Attribute

index.html



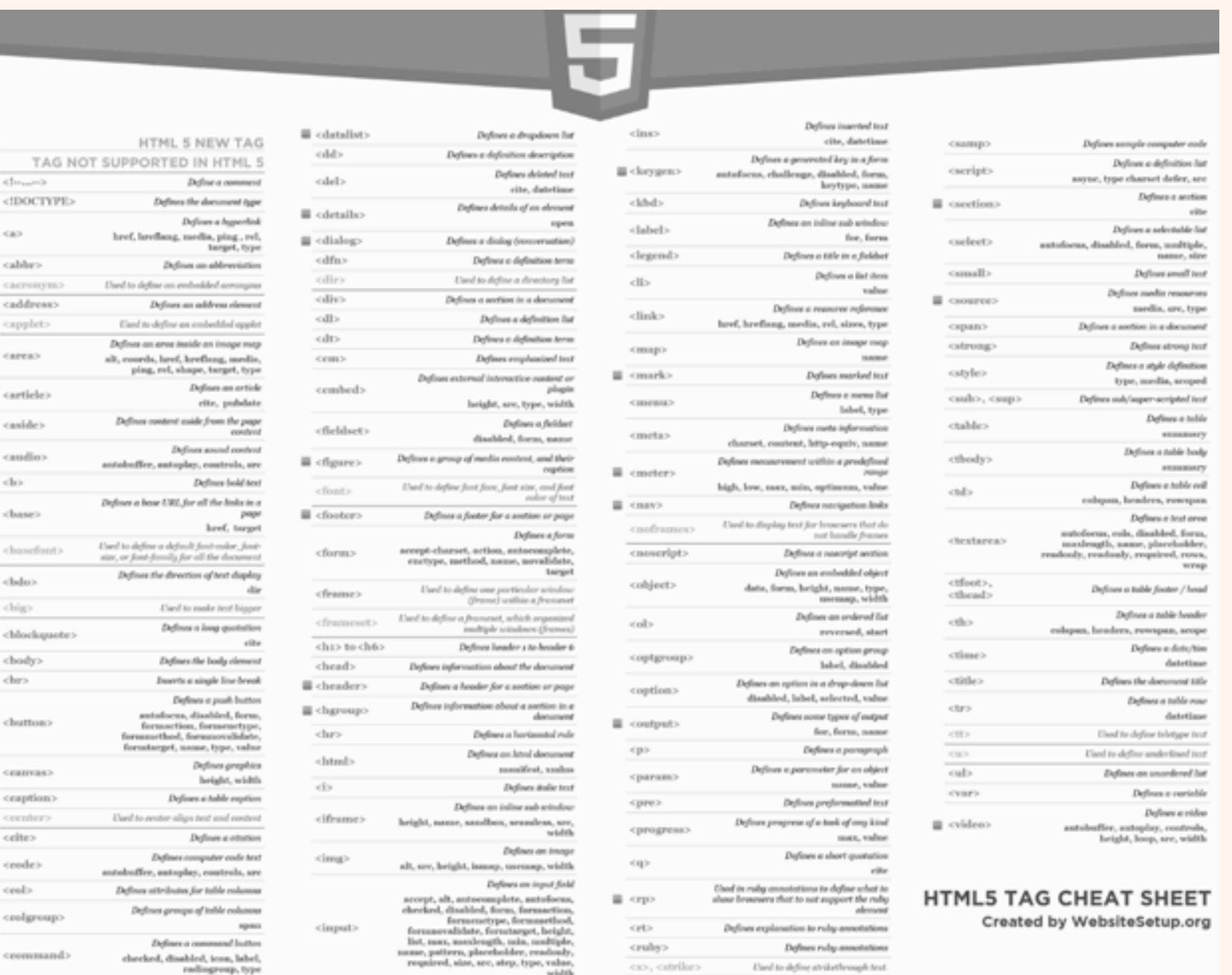
```
1  <html>
2    <head>
3      <title>Bits and Atoms II</title>
4    </head>
5
6    <body>
7      Hello
8    </body>
9  </html>
```

index.html

Selection of important tags:

<p>
<h1> - <h6>
<a>

/
<input>
<table>
<div>



<https://websitesetup.org/wp-content/uploads/2014/09/html5-cheat-sheet.png>

Task (15min)

- Create a document featuring
 - A title
 - A paragraph

Coding together

Task 2 (10min)

- Add an image
- Add a link (<a>)