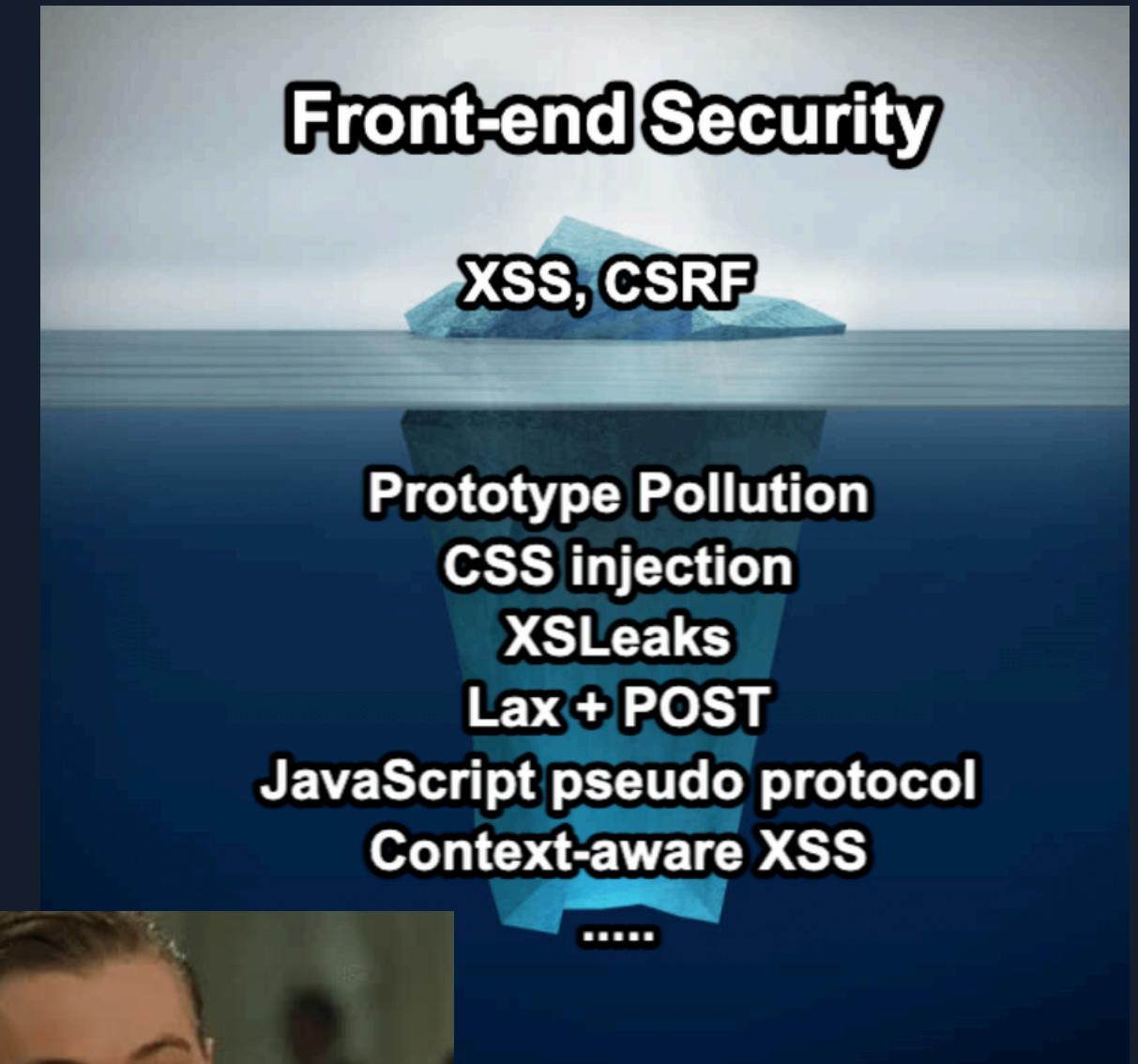


Breaking the Frontend - Client Side Hacking 101

Jesus Lujan (Aka. s4yhii)

**Hack The Box Meetup:
Lima, PE.**



Contenido



HACKTHEBOX



Introducción al Hacking Client-Side

1. Qué es client-side?
2. Web Architecture + Browser Security Model
3. Casos reales
 - (Tampering javascript, Local Storage attack, Cookie attacks, XSS, Postmessage, Source Mapping...)

Devtools for hacking

Devtools 101: **Elements, Console, Sources, Network, Application**

Reverseando Javascript en el navegador

1. Source Mapping: Revelando código original
2. Lazy Loading (Carga diferida) de Javascript
3. Desofuscación de archivos JS

Vulnerabilidades Client-Side

1. Habilitando funciones restringidas
2. Sobreescribir funciones y redefinir eventos
3. Usando Breakpoints para debuggear JS
4. Proximamente... (CSPT, CSTI)

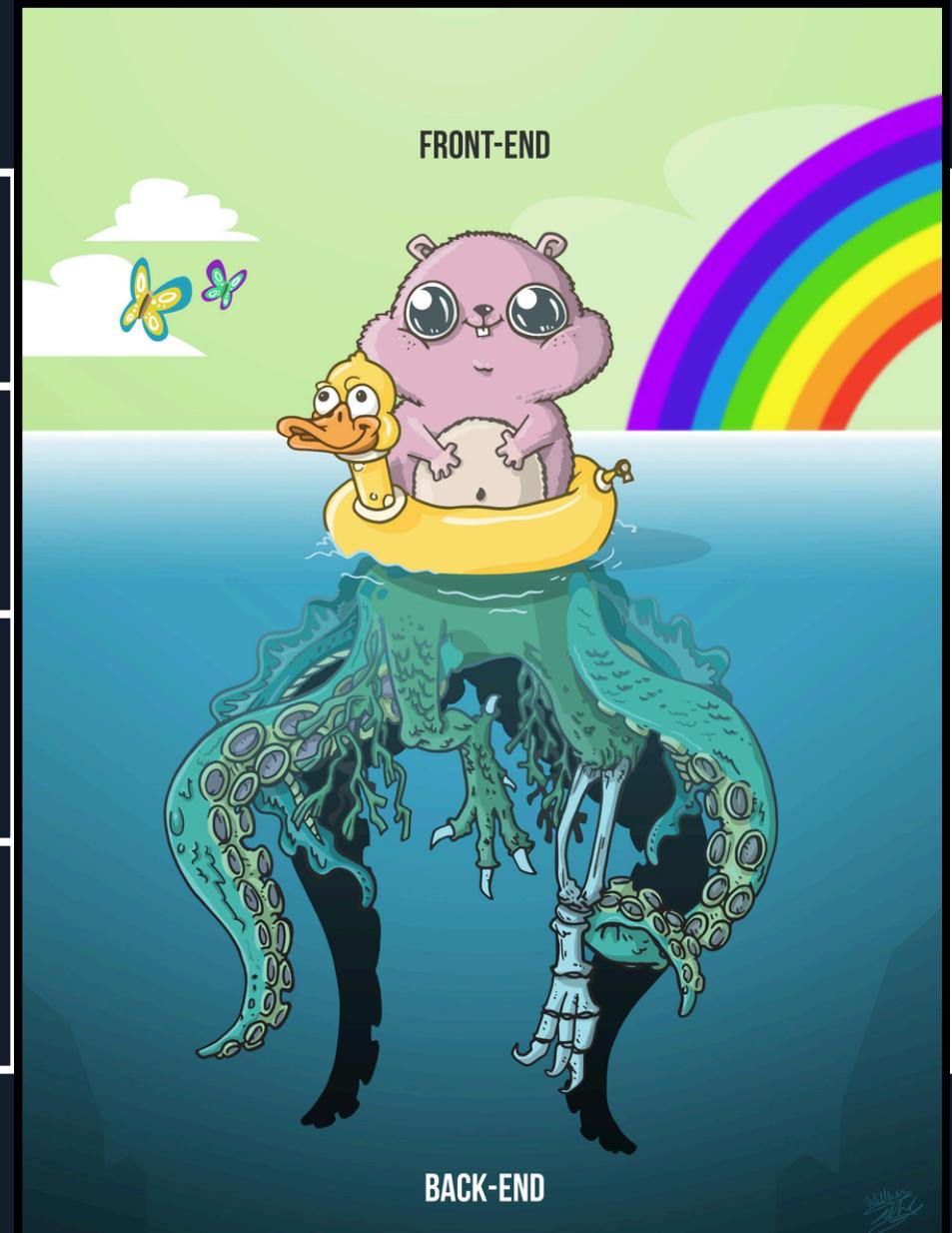
CTF Time (2 Vip+ winners)

Instructor: Jesus Lujan Montufar
<https://www.linkedin.com/in/jesusluj4n/>

Qué es client side?

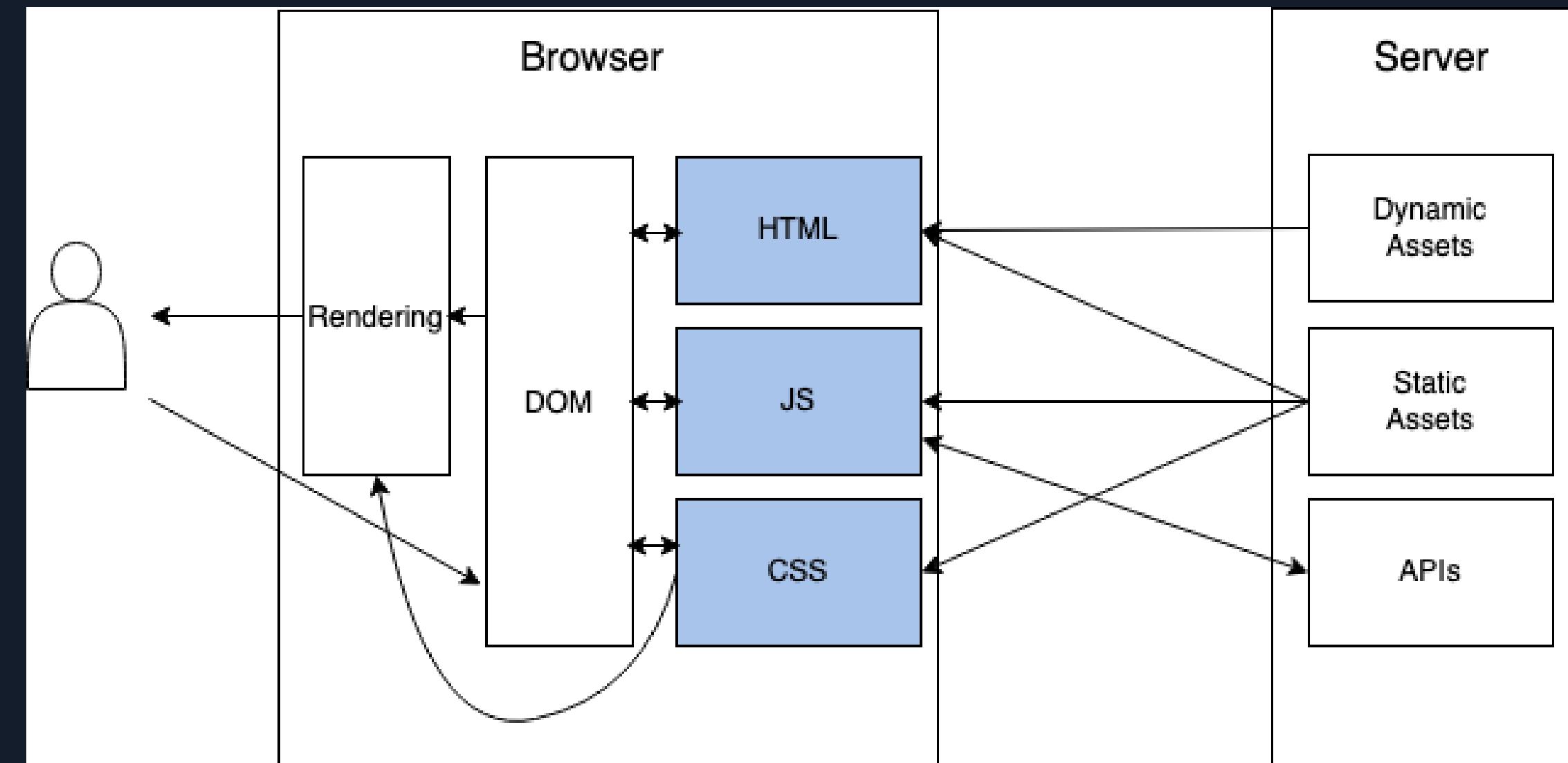


Client-Side
Se ejecuta en el navegador (HTML, CSS, JS)
Modificable por los usuarios (DevTools)
Ejemplos: Botones como click, Validaciones en formularios

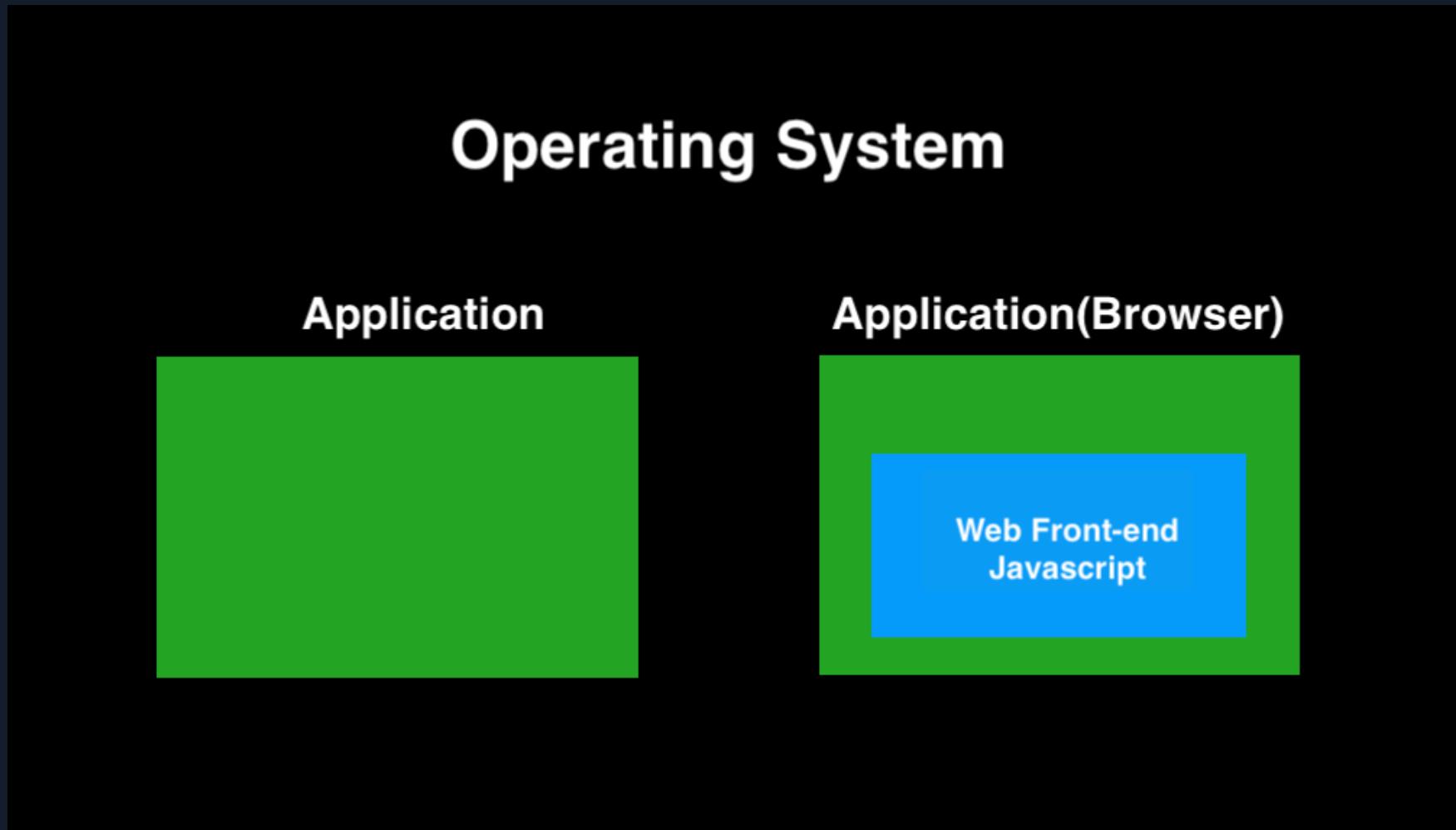


Server-Side
Se ejecuta en el servidor (PHP, Python, Node)
No es visible por los usuarios (Lógica de la app)
Ejemplo: Consultas a la bd, autenticación?

Web Architecture



Browser Security Model



Limitaciones del navegador

- Archivos locales
- APIs del sistema
- Contenidos de otras páginas web

Importante??



Client-side RCE via symlink
following in Google Web
Designer for macOS/Linux: CVE-
2025-1079

Using YouTube to steal your files

Hacking Starbucks and Accessing Nearly 100 Million Customer Records

Sat Jun 20 2020

Links de referencia al final



Devtools for hacking 101



- Panel Elements:

Modificar el DOM (editar formularios, botones, campos ocultos)

- Panel Console:

Ejecutar JavaScript en tiempo real, sobreescribir variables, usar debugger

- Panel Network:

Interceptar llamadas a APIs, modificar peticiones fetch

- Panel Application:

Manipular cookies, localStorage, sessionStorage, IndexedDB

- Panel Sources:

Depurar la ejecución de JavaScript, establecer puntos de interrupción (breakpoints)

Lab URL : <http://localhost:3000>



CTF TIME (10 min)



Ingresá a:
<http://spooky.bugcrowd.zw.ink/kloon.cfm>

Y envíame Screenshot por Disc del Secret password



Reverseando JS

Instructor: Jesus Lujan Montufar
<https://www.linkedin.com/in/jesusluj4n/>

Source Mapping, que es?



Herramienta que permite
debugear código más fácil.

The diagram illustrates the Source Mapping ecosystem, showing the flow from various preprocessors and frameworks through build tools to final output files. A dashed arrow points from the build tools section to the right, where the resulting HTML, CSS, and JavaScript files are listed. Below this, a screenshot of a browser developer tools interface shows a file tree on the left and the source code for 'index.7808df6e.js' on the right. A line of code at the bottom of the source code editor is highlighted with a red box, indicating a sourceMappingURL directive.

CSS preprocessors: Sass, PostCSS, LESS

JavaScript frameworks: Angular, React, Vue, Svelte

TypeScript: TS

Transpiler & compressors: BABEL, terser

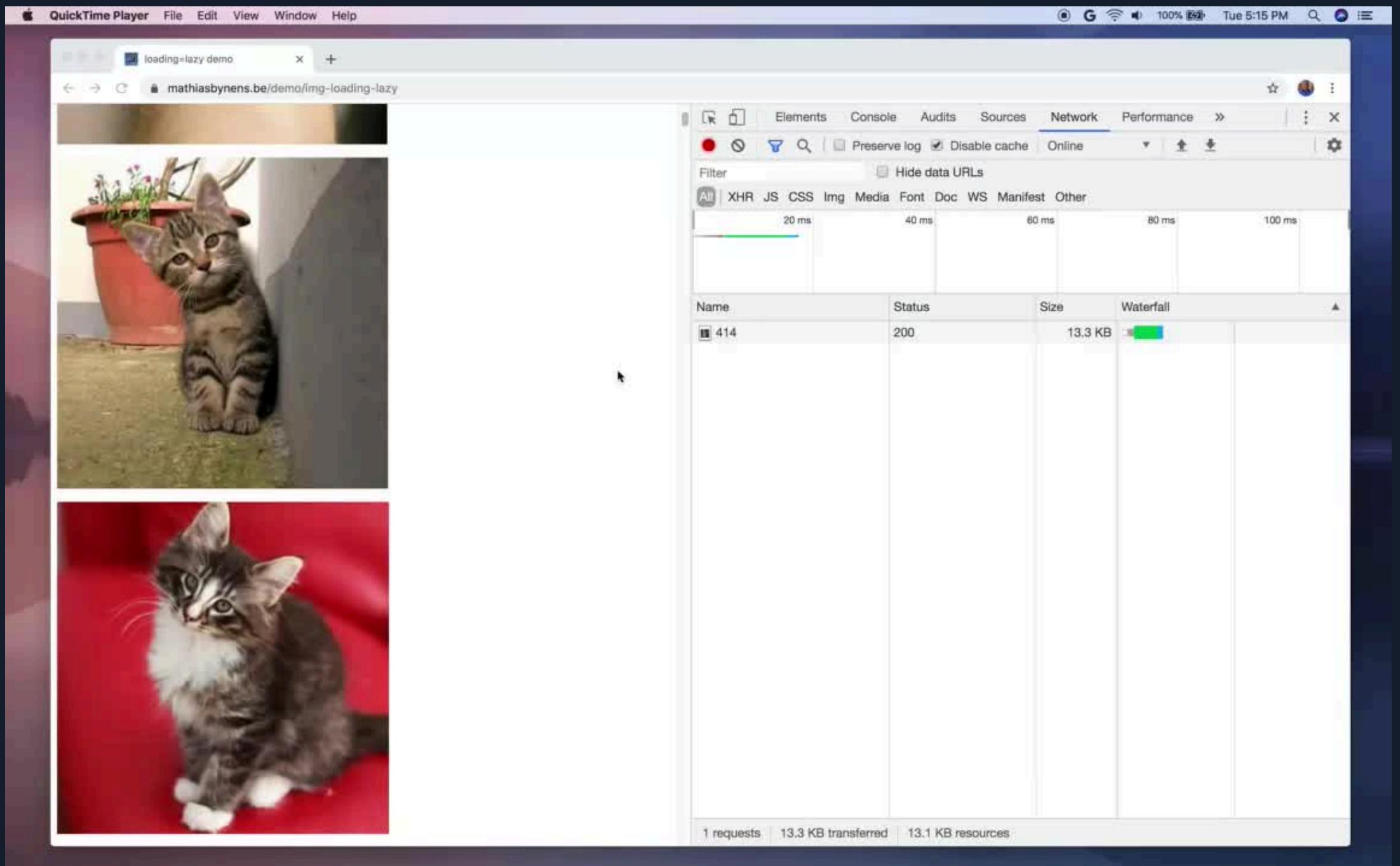
Build tools: Vite, Rollup, Webpack, ESLint, Browsersync

Output: HTML, CSS, JavaScript

Source code (index.7808df6e.js):

```
1 document.querySelector("button")?.addEventListener("click", ( () => {
-   const e = Math.floor(101 * Math.random());
-   document.querySelector("p").innerText = `Hello, you are no. ${e}!`;
-   console.log(e)
- });
- )):
2 //# sourceMappingURL=index.7808df6e.js.map
3
```

Lazy Loading o Carga diferida



Vulnerabilidades Client Side 101



Lógica Client Side



Navegador
- “Lo que tu
digas, mr API!”

DOM Renderizado
SUPER ADMIN, MEGA
USER!

100MB de código JS
minificado

Página HTML
<script src=...

BTW, que rol tiene este usuario?
← TODOS LOS ACCESOS, SUPER ADMIN,
TODAS LAS FUNCIONES, MEGA USER!

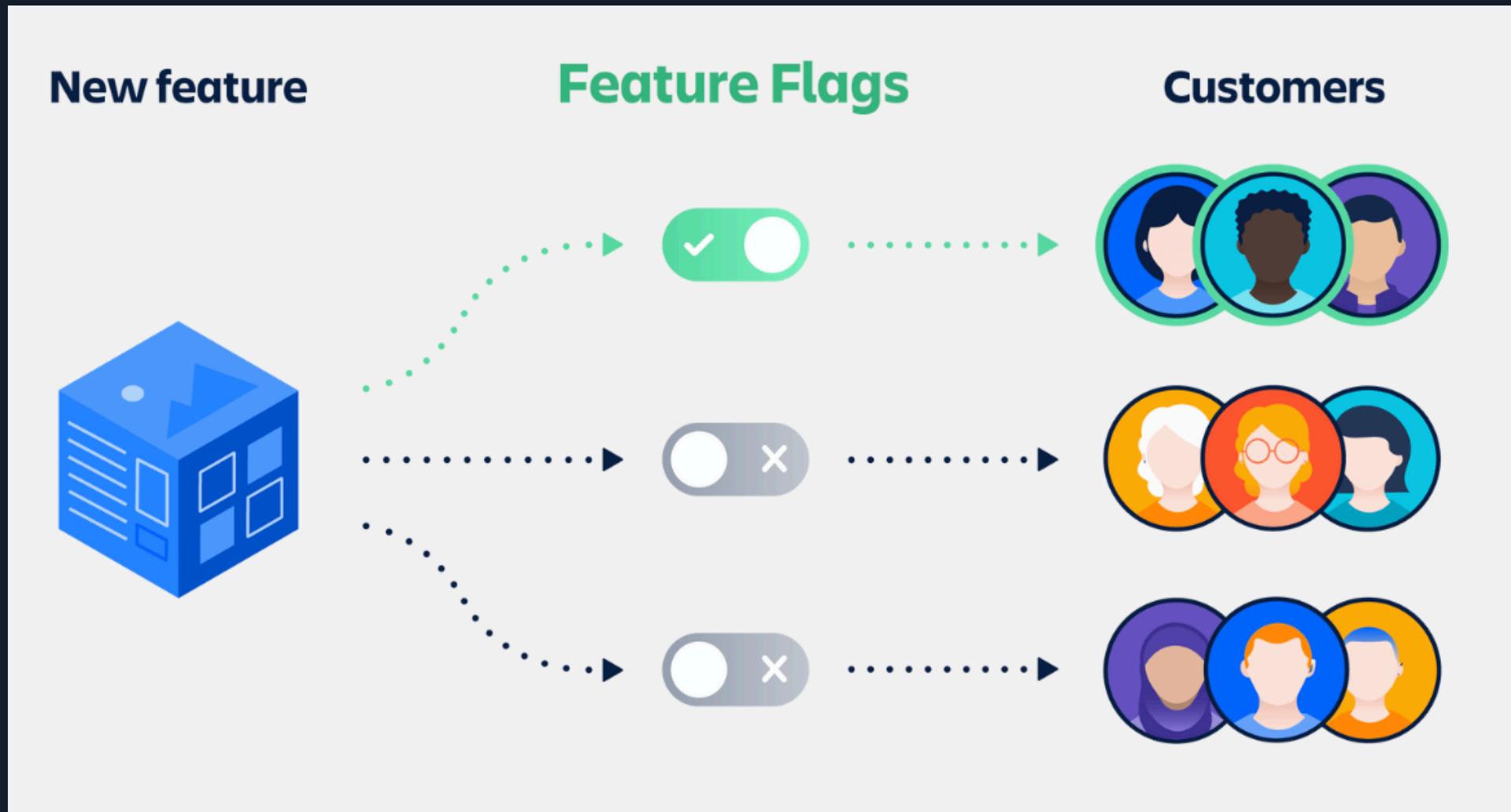
Obviamente no soy Caido API.
Normal API



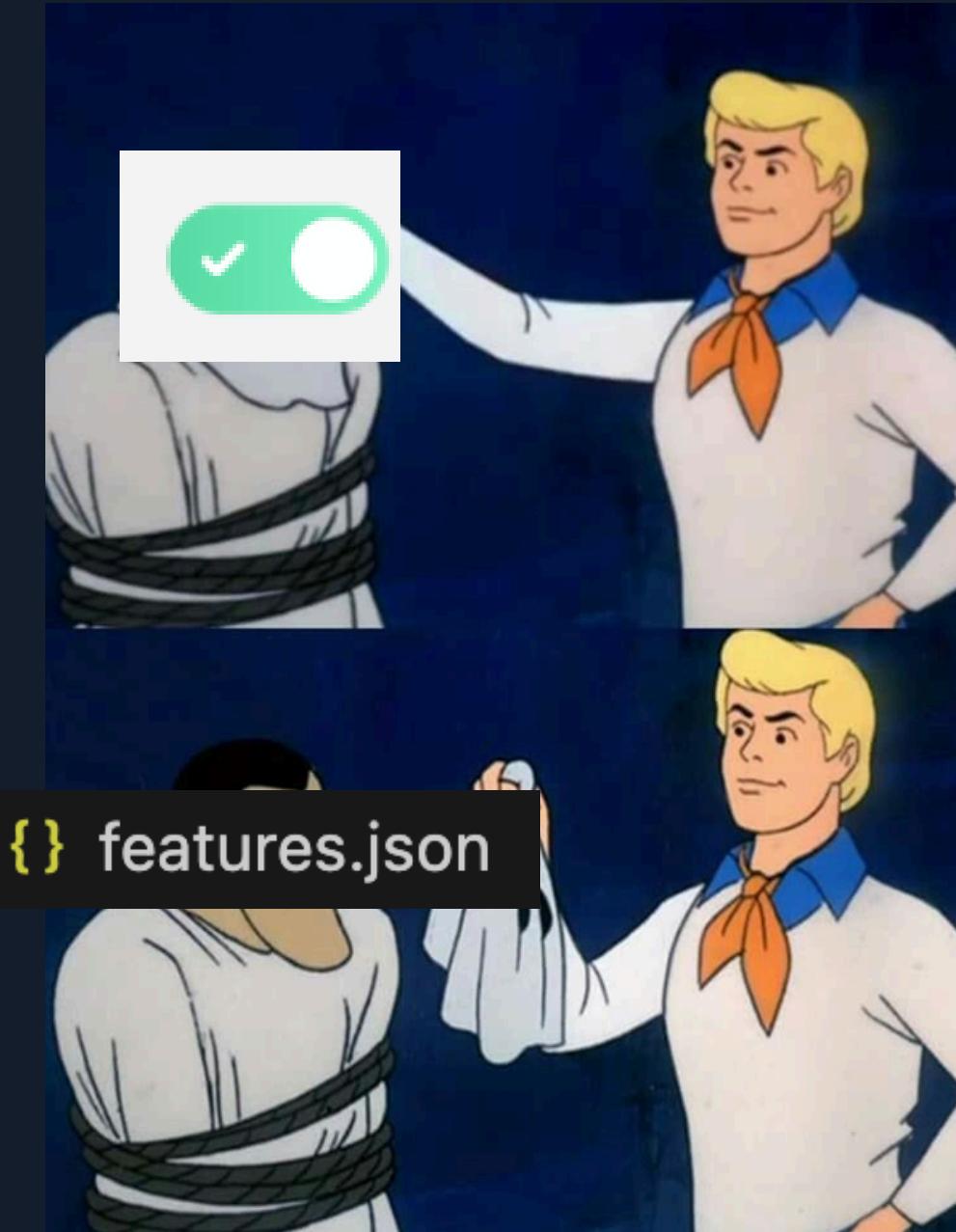
```
if(kl.mE==w.tF){O,w,HY)(Z)}
```

```
if(user.role==="MEGA_ADMIN"){Enseña las funciones ocultas  
del UI}
```

Habilitando Feature Flags



```
{ } features.json {"enableNewHelpDeskBot":false"true}
```



Bienvenido Match & Replace



The screenshot shows a configuration interface for a "Match & Replace" rule named "adminntrue".

Update rule adminntrue

Name: adminntrue

Strategy: Response Body

Search term: enable(["]*")":false

Replace term: enable\$1":true

Condition: Enter an HTTPQL query...

Before:

```
1 HTTP/1.1 200 OK
2 Content-Type: application/json
3 Content-Length: 15
4
5 {"enableSuperAdminMode":false}
```

After:

```
1 HTTP/1.1 200 OK
2 Content-Type: application/json
3 Content-Length: 15
4
5 {"enableSuperAdminMode":true}
```

A red box highlights the "Search term" and "Replace term" fields, and another red arrow points from the "Before" response body to the "After" response body, indicating the scope of the replacement.



Sobreescribiendo funciones



Usando breakpoints



CTF TIME (10 min)

Accede a http://localhost:80 y obtén la flag.

Premio: HTB voucher vip±

Hint: Existe + de una solución

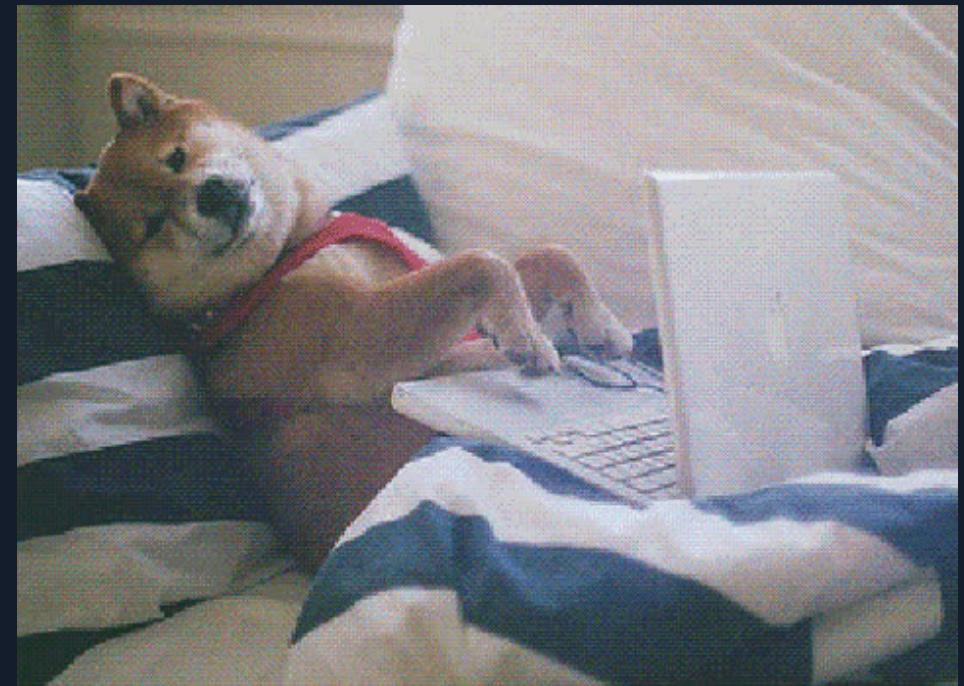
Recursos 403



Must!: <https://aszx87410.github.io/beyond-xss/en/>

Links blogs interesantes

- <https://balintmagyar.com/articles/google-web-designer-symlink-client-side-rce-cve-2025-1079>
- <https://lyra.horse/blog/2024/09/using-youtube-to-steal-your-files/>
- <https://samcurry.net/hacking-starbucks>
- <https://web.dev/articles/browser-level-image-lazy-loading>



Links Laboratorios para practicar

- <https://verified.capitalone.com/>
- <http://spooky.bugcrowd.zw.ink/klown.cfm>

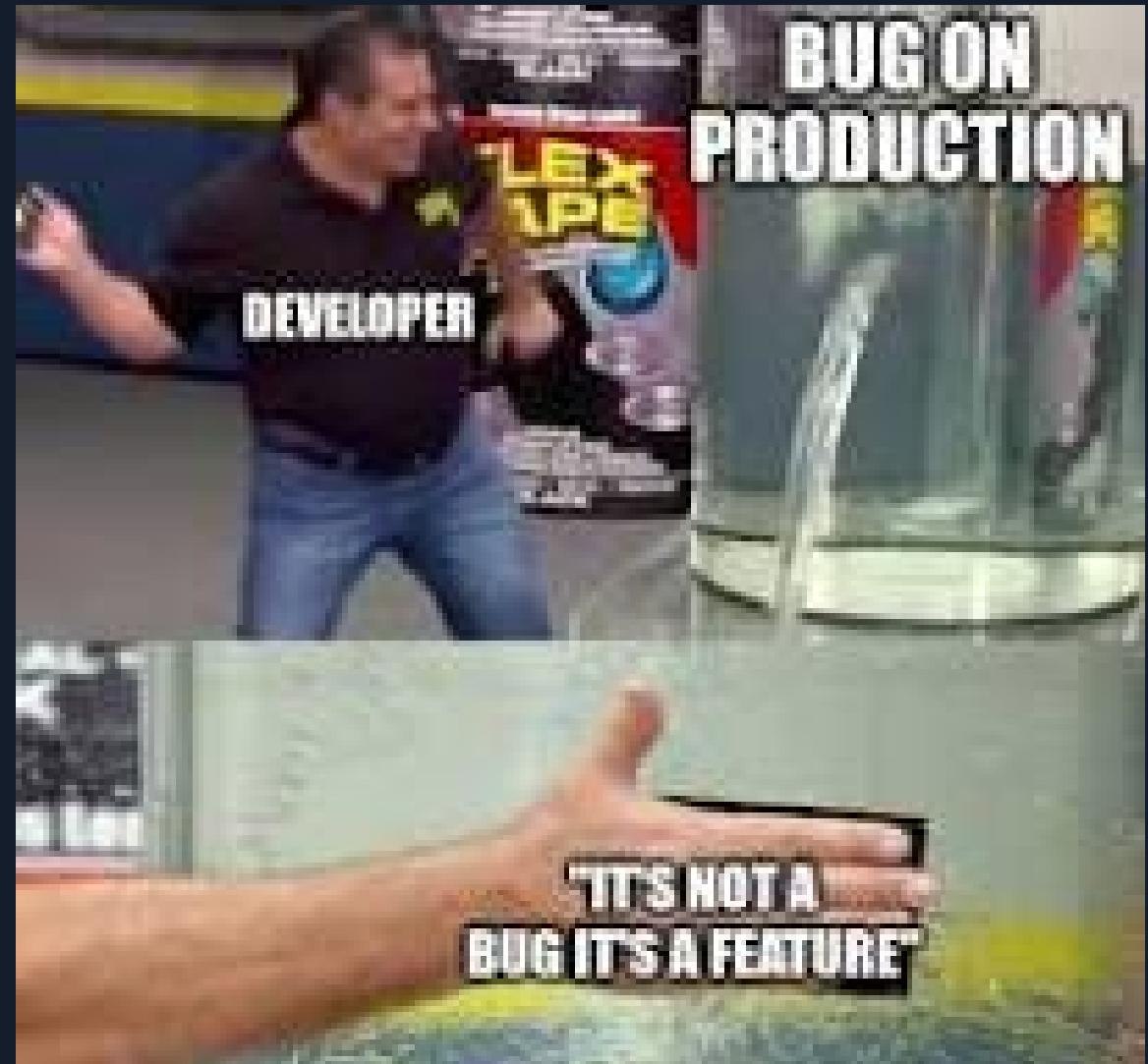
Proximamente



- CSPT (Client Side Path Traversal/si, pero en el cliente)
- Prototype Pollution
- DOM Clobbering
- DOM Logger ++
- Json Injection

O quieren ver lado del servidor

- SQLi
- OS Command Injection
- IDORs, etc ...



Conectemos.



Instructor: Jesus Lujan Montufar
<https://www.linkedin.com/in/jesusaluj4n/>



Graciass

Instructor: Jesus Lujan Montufar
<https://www.linkedin.com/in/jesusaluj4n/>