UNIVERSITÀ DELLA CALABRIA
il Campus per eccellenza

Cross-origin resource sharing (CORS) + Clickjacking

Mario Alviano

Main References

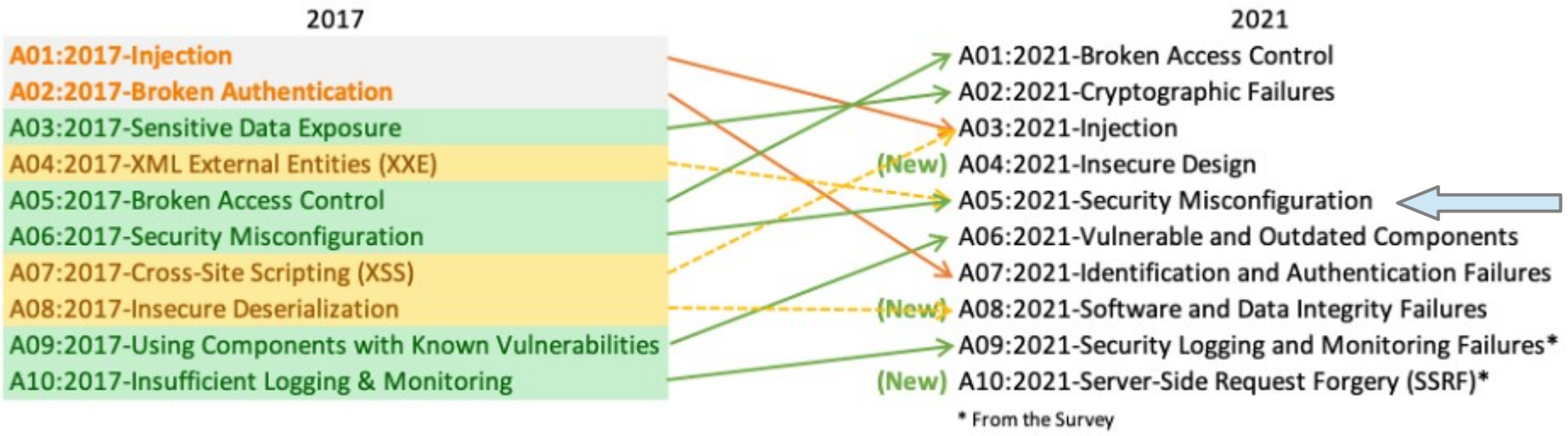
Bug Bounty Bootcamp – Chapters 8 and 19

https://portswigger.net/web-security/cors

https://portswigger.net/web-security/clickjacking

OWASP Top Ten

A broad consensus about the most critical security risks to web applications

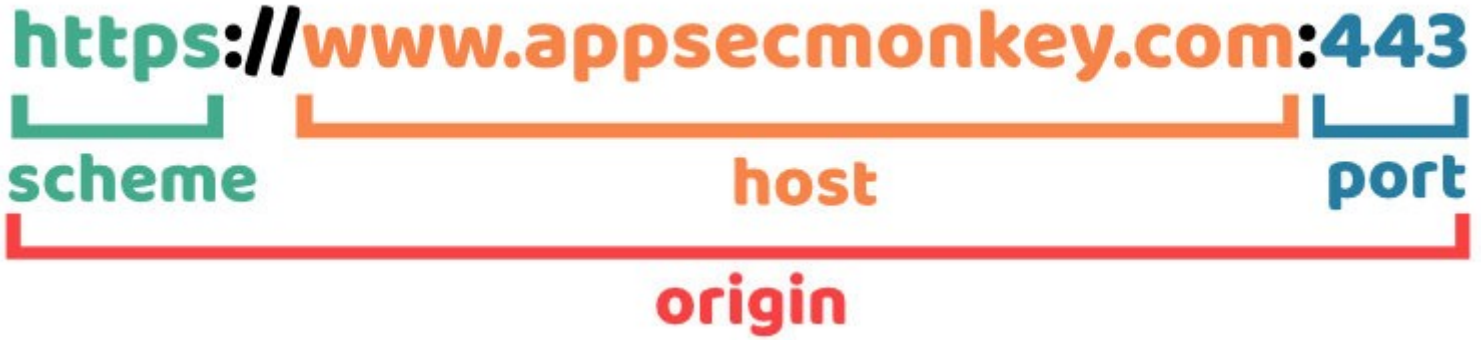


Cross-Origin Resource Sharing (CORS)

CORS relaxes the same-origin policy (SOP). Misconfigured CORS policies lead to information leaks, authentication bypass and account takeover.

Poorly configured CORS policy: it allows requests originated from malicious-website.com to be processed and to carry credentails on

Same-Origin Policy (SOP)



A script from page A can access data from page B only if the pages are of the same origin.

SOP protects cookies (like session cookies).

CORS adds flexibility to SOP. Must be understood and properly configured.

Why relaxing SOP?

To access REST APIs from other domains.

B requests user_info to A

b.example.com

a.example.com/user_info

A respond with JSON (if it accepts requests from B)

```
{"username": "vickieli", "account_number": "12345"}
```

SOP blocks the access to JSON

Browsers add Origin headers to requests

b.example.com

Origin: https://b.example.com

a.example.com/user_info

A processes the request if the origin is in the allow list

Access-Control-Allow-Origin: b.example.com

A adds this header to its response

```
{"username": "vickieli", "account_number": "12345"}
```

The browser allows to read the JSON content

Common mistakes

```
Access-Control-Allow-Credentials: true
```

Problems arise when access to cookies and authentication headers is granted

```
Access-Control-Allow-Origin: null
```

Essentially disables CORS. Any origin can specify **Origin: null**, including **data:** scheme.

```
Access-Control-Allow-Origin: https://attacker.com
```

Don’t reflect the Origin value here without validation. This also essentially disables CORS.

```
Access-Control-Allow-Origin: https://www.example.com.attacker.com
```

Don’t use weak regexes like
Origin starts with www.example.com

```
Access-Control-Allow-Origin: *
```

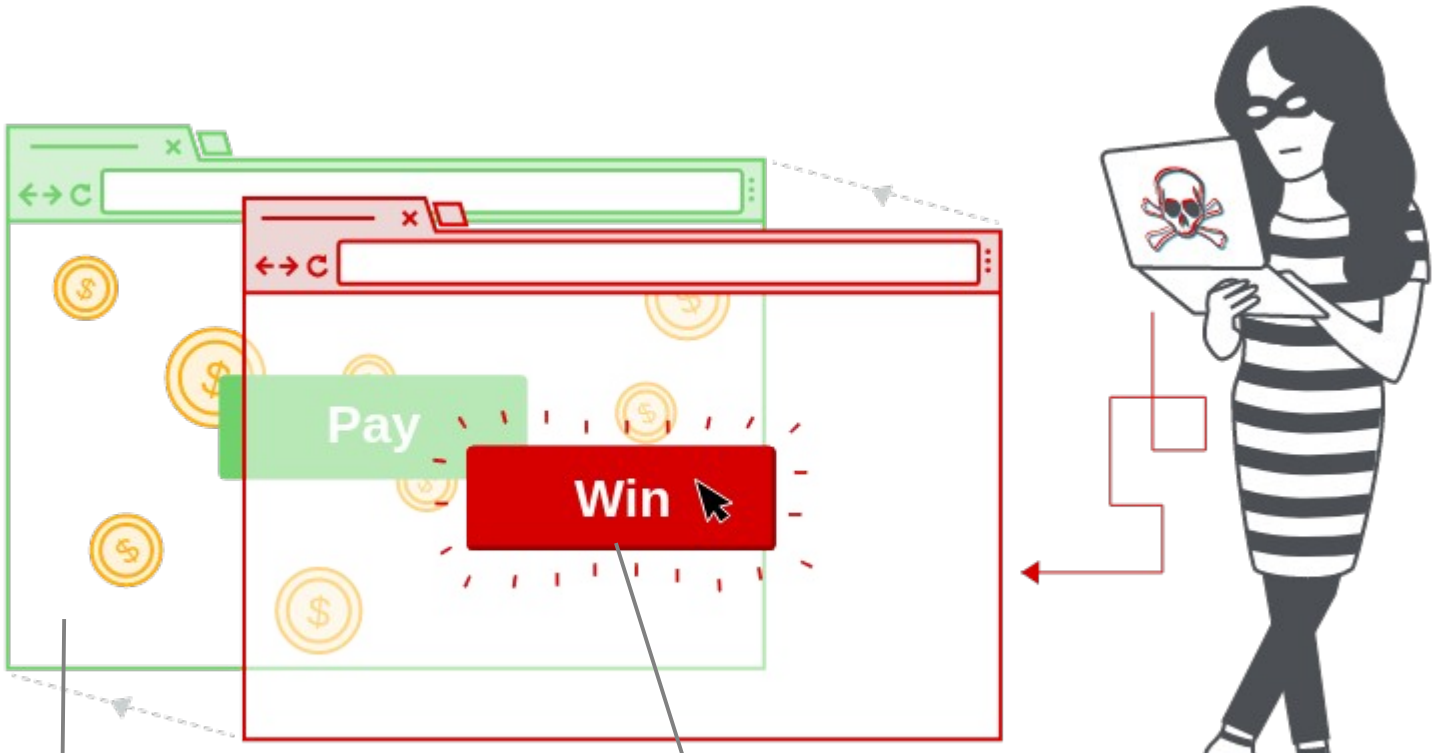
This is OK because it disables access to credentials

Prevention

- Specify the **Access-Control-Allow-Origin** header if the response contains sensitive data
- Think if you really need **Access-Control-Allow-Origin: ***
- Double think if you really need **Access-Control-Allow-Credentials: true**
- Use allow lists
- Don’t allow the origin **null**

Clickjacking (aka UI redressing)

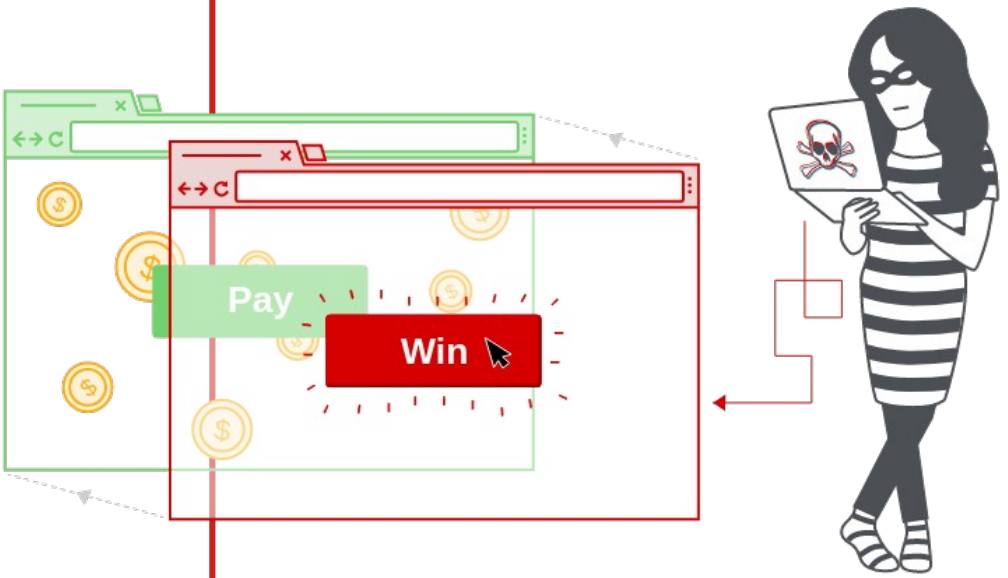
Tricks users into clicking a malicious button that has been made to look legitimate.



Vulnerable website

Catchy button in the same position of the Pay button

```
<head>
        <style>
                #target_website {
                        position:relative;
                        width:128px;
                        height:128px;
                        opacity:0.00001;
                        z-index:2;
                        }
                #decoy_website {
                        position:absolute;
                        width:300px;
                        height:400px;
                        z-index:1;
                        }
        </style>
</head>
...
<body>
        <div id="decoy_website">
        ...decoy web content here...
        </div>
        <iframe id="target_website" src="https://vulnerable-website.com">
        </iframe>
</body>
```



Vulnerable website is included as an iframe, made transparent and put above the catchy button

Prevention

```
X-Frame-Options: DENY
X-Frame-Options: SAMEORIGIN
```

Add headers to disable use in frames

```
Content-Security-Policy: frame-ancestors 'none';
Content-Security-Policy: frame-ancestors 'self';
```

```
Content-Security-Policy: frame-ancestors 'self' *.example.com;
```

Use CSP to disable use in frame (or finegrained control)

```
Set-Cookie: PHPSESSID=UEhQUOVTUOlE; Max-Age=86400; Secure; HttpOnly; SameSite=Strict
Set-Cookie: PHPSESSID=UEhQUOVTUOlE; Max-Age=86400; Secure; HttpOnly; SameSite=Lax
```

Use SameSite flag for cookies

Questions