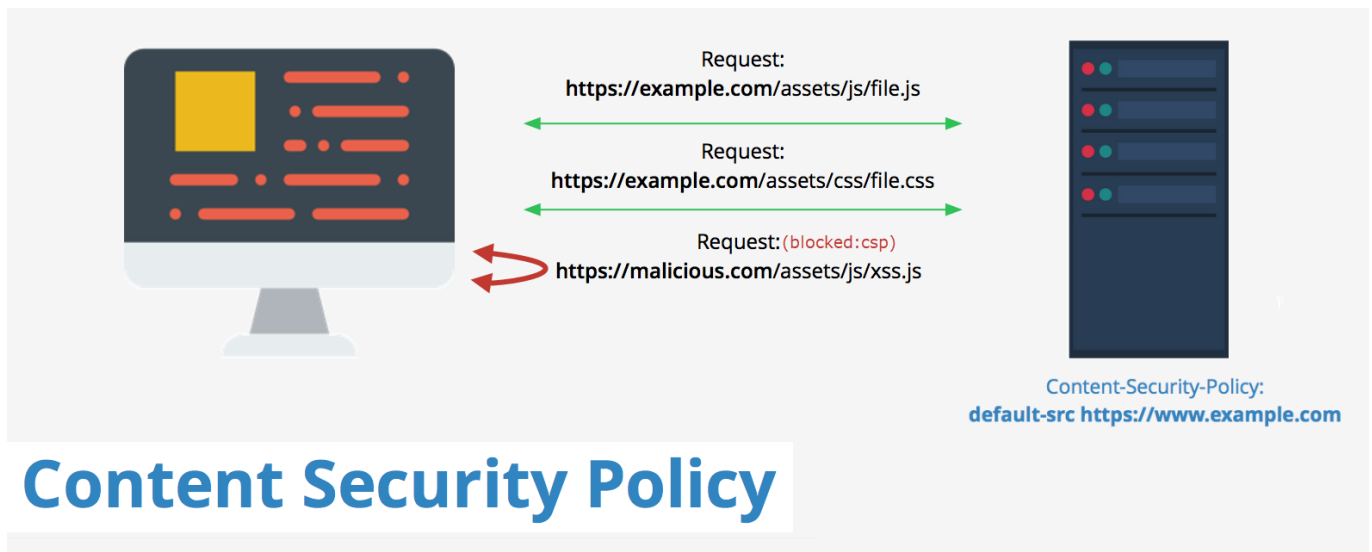


CSP and CORS

1. CSP - Content Security Policy
2. CORS - Cross-Origin Resource Sharing

1. CSP - Content Security Policy

CSP (Content Security Policy), là một cơ chế bảo mật trong web để giúp ngăn chặn các loại tấn công phổ biến như **XSS** (Cross-Site Scripting). CSP cho phép các nhà phát triển xác định các nguồn tài nguyên nào được trình duyệt của người dùng phép tải xuống và thực thi. Bằng cách này, CSP giảm thiểu khả năng một trang web bị tấn công bằng cách chặn việc thực thi mã JavaScript từ các nguồn không tin cậy hoặc không mong muốn.



Source: <https://www.rahulnath.com/blog/http-content-security-policy-csp/>

```
// try to run this line of code on MDN and see
fetch('http://js-post-api.herokuapp.com/api/students?_page=1')
// error: Refused to connect to 'http://js-postapi.herokuapp.com/api/students?_page=1' because it violates the document's Content Security Policy
```

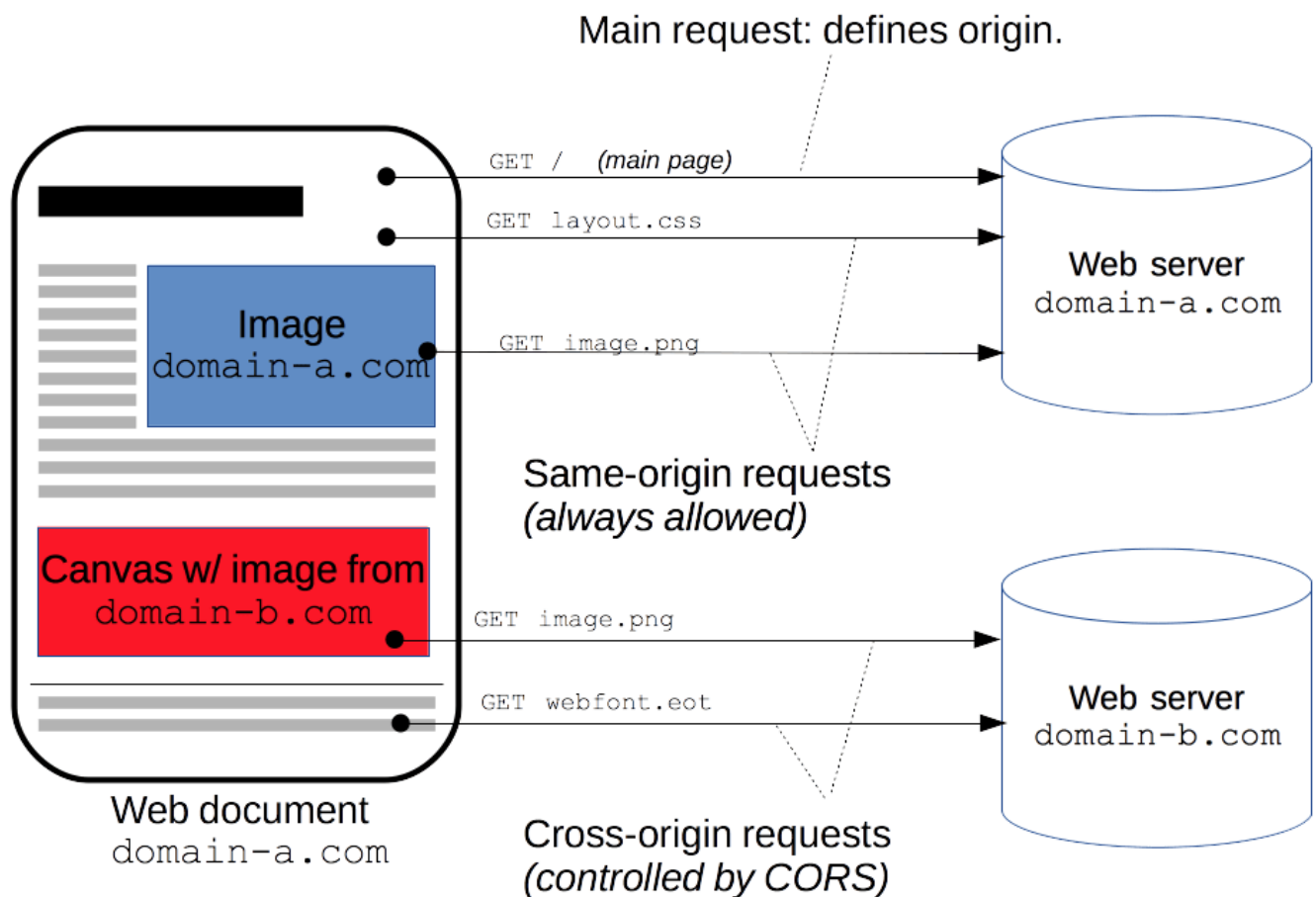
Source: <https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP>

2. CORS - Cross-Origin Resource Sharing

CORS, hay Cross-Origin Resource Sharing, là một cơ chế trong web cho phép một trang web yêu cầu tài nguyên từ một nguồn tài nguyên khác (origin) ngoài tên miền của nó. Origin bao gồm giao thức (http, https), tên miền và cổng (nếu có).

✗ Access to fetch at 'https://joke-api-strict-cors.appspot.com/r/localhost/:1/andom_joke' from origin '<http://localhost:3000>' has been blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present on the requested resource. If an opaque response serves your needs, set the request's mode to 'no-cors' to fetch the resource with CORS disabled.

Source: <https://medium.com/@dtkatz/3-ways-to-fix-the-cors-error-and-how-access-control-allow-origin-works-d97d55946d9>



Source: <https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>

```
// access this from google.com console
fetch('https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS/bcd.json')
// Access to fetch at
'https://developer.mozilla.org/enUS/docs/Web/HTTP/CORS/bcd.json' from origin
'https://www.google.com' has been blocked by CORS policy: No 'Access-Control-
Allow-Origin' header is present on the requested resource. If an opaque response
serves your needs, set the request's mode to 'no-cors' to fetch the resource with
CORS disabled.
```

Client

Server

Preflight request

OPTIONS /doc HTTP/1.1
Origin: https://foo.example
Access-Control-Request-Method: POST
Access-Control-Request-Headers: X-PINGOTHER, Content-type
...

HTTP/1.1 204 No Content
Access-Control-Allow-Origin: https://foo.example
Access-Control-Allow-Methods: POST, GET, OPTIONS
Access-Control-Allow-Headers: X-PINGOTHER, Content-Type
Access-Control-Max-Age: 86400
...

Main request

POST /doc HTTP/1.1
X-PINGOTHER: pingpong
Content-Type: text/xml; charset=UTF-8
Origin: http://foo.example
...

HTTP/1.1 200 OK
Access-Control-Allow-Origin: http://foo.example
Vary: Accept-Encoding, Origin
Content-Encoding: gzip
Content-Length: 235
...