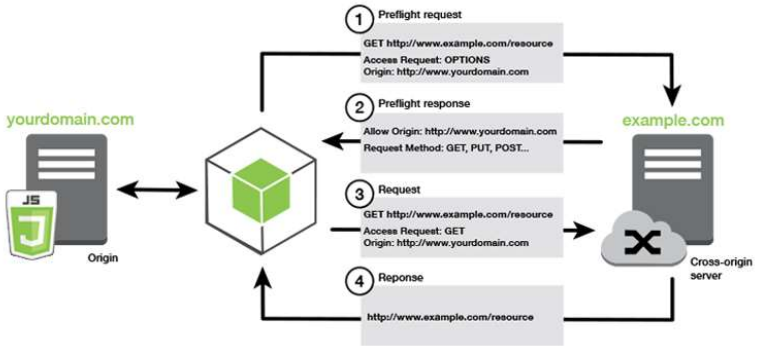
CORS & CSRF

Was ist CORS?

- **C**ross-**O**rigin **R**esource **S**haring
- Die Kommunikation bleibt standardmässig auf dieselbe Domain beschränkt. Abweichungen davon muss der Server explizit akzeptieren.
- Browser implementieren dazu eine *Same-origin policy*.
- HTTP-Methoden welche potentiell Daten modifizieren (z.B. PUT Request) senden vor dem eigentlichen Request einen *preflight request* um den API-Zugriff anzukündigen. Der Server akzeptiert den Request oder auch nicht.
- *https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS*

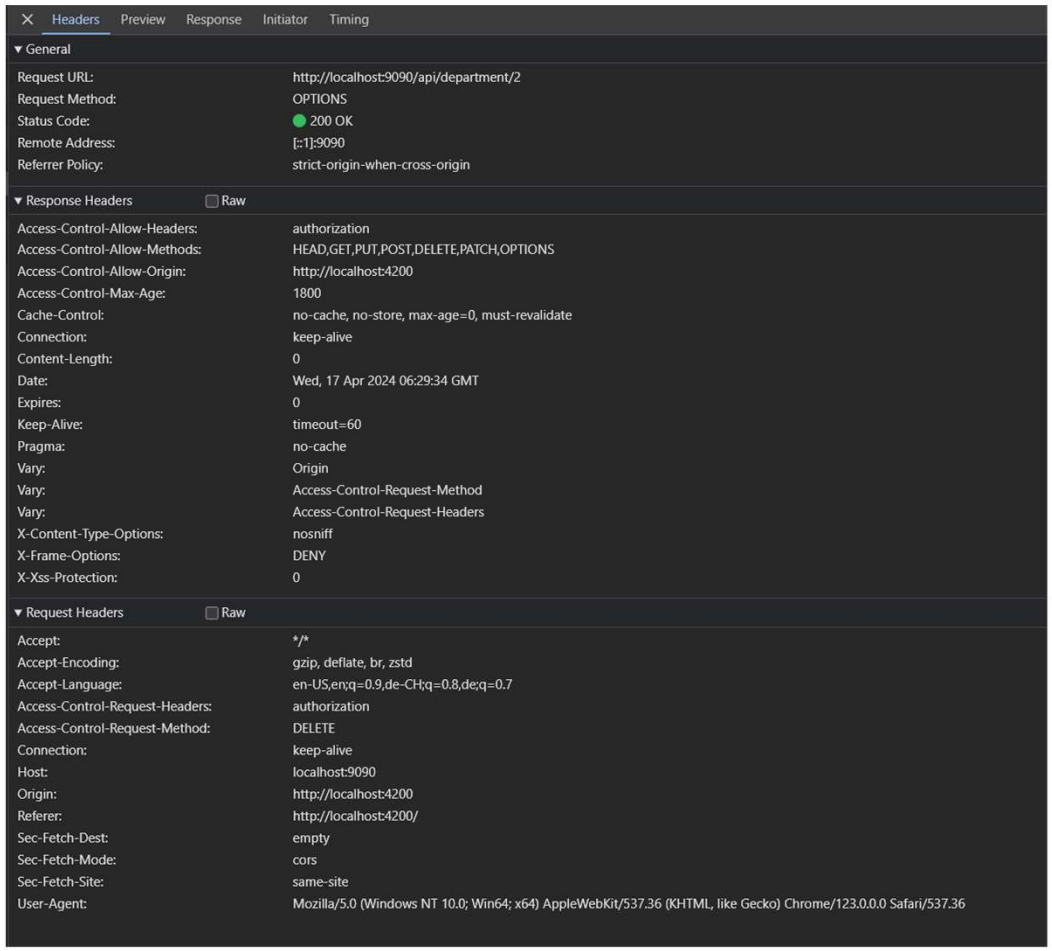
CORS / 2

- Bei nicht modifizierenden Requests (z.B. GET) wird der Access Control Header üblicherweise direkt mit dem Request mitgeschickt. Ansonsten wird durch den Browser zuerst ein Preflight Request ausgelöst.



Quelle: https://aws.amazon.com

2	OPTIONS	200	preflight	Preflight	0 B	2 ms
2	DELETE + Preflight	200	xhr	department-list.component.ts:67	587 B	21 ms


Headers
Preview
Response
Initiator
Timing
General
Request URL: http://localhost:9090/api/department/2
Request Method: OPTIONS
Status Code: 200 OK
Remote Address: [::1]:9090
Referrer Policy: strict-origin-when-cross-origin
Response Headers
Raw
Access-Control-Allow-Headers: authorization
Access-Control-Allow-Methods: HEAD,GET,PUT,POST,DELETE,PATCH,OPTIONS
Access-Control-Allow-Origin: http://localhost:4200
Access-Control-Max-Age: 1800
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Connection: keep-alive
Content-Length: 0
Date: Wed, 17 Apr 2024 06:29:34 GMT
Expires: 0
Keep-Alive: timeout=60
Pragma: no-cache
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
X-Xss-Protection: 0
Request Headers
Raw
Accept: */*
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9,de-CH;q=0.8,de;q=0.7
Access-Control-Request-Headers: authorization
Access-Control-Request-Method: DELETE
Connection: keep-alive
Host: localhost:9090
Origin: http://localhost:4200
Referer: http://localhost:4200/
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-site
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.0.0 Safari/537.36

Was ist CSRF

- **C**ross-**S**ite **R**equest **F**orgery
- Requests werden für den (angemeldeten) Benutzer unbewusst ausgeführt.
- GET Requests können einfach z.B. durch das Einbetten von Bildern in die Webseite abgesetzt werden.
- Durch eingeschleuste Formulare können POST Requests ausgeführt werden.
- Solche Requests können unter anderem durch ein automatisch generiertes Token verhindert werden welches zwischen Server und Client ausgetauscht wird.
- GET Requests sollten keine Daten verändern!
- Ziemlich gute Zusammenfassung zum Thema:

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