CORS basics

WASA: Web and Software Architecture

Emanuele Panizzi

CORS = Cross-Origin Resource Sharing

Different Origins

- a user agent (UA) loads a web page from the origin server www.origin1.com
- a script on this page wants to fetch something from another origin: www.origin2.com

```
GET / HTTP 1.1
Host: www.origin2.com
```

- Would Origin2 accept that a script provided by Origin1 reads its data?

The UA asks (*Simple Request*)

The UA adds a header to the request…

`Origin: http://www.origin1.com`

…to verify if Origin2 accepts to share its resources with Origin1

Origin2 responds

- If Origin2 answers positively, it adds a header to the response, like:

Access-Control-Allow-Origin: http://www.origin1.com

or

*Access-Control-Allow-Origin: **

- Else, it returns an error indicating it does not accept cross origin requests from that origin

What about non SAFE methods?

- e.g., DELETE
- If Origin2 supports CORS, it can respond with an error and not perform the operation

What if Origin2 does not support CORS?

- a non-SAFE operation could be executed
- browsers implement **preflight requests** to avoid this

Preflight requests

1. make a request just to verify
2. then make the real request

```
OPTIONS / HTTP/1.1
Host: www.origin2.com
Origin: http://www.origin1.com
Access-Control-Request-Method: DELETE
```

If response from Origin2 contains these headers:

```
Access-Control-Allow-Origin: http://www.origin1.com
Access-Control-Allow-Methods: DELETE
```

then it is ok to send the *DELETE* request.