**CORS**

**(Cross Origin Resource Sharing)**

A request made to a resource with a different protocol, sub-domain or port qualifies as cross origin request.

**Simple request methods**

* GET, HEAD and POST (x-www-form-urlencoded, multipart/form-data, text/plain). These are not pre-flighted.
* The browser blindly makes a request with the "origin" header field set to the ip address of the client. The server responds with the resource and also the **Access-Control-Allow-Origin** header.
* The browser compares the "origin" with this header and if they match, the resource is shown to the user. Otherwise, blocks it.

**Pre-flighted Requests**

The other methods - PUT, DELETE and also POST with other content-types are preflighted with the "OPTIONS" method. With custom-headers set in the request.

Browser sends the following headers (first line is the "kind of" request anyway)

***OPTIONS http://somecrossdserver.com HTTP/1.1***

***Origin: ip address***

***Access-Control-Request-Method: POST (the method type to be used during the ACTUAL request)***

***Access-Control-Request-Headers: Content-type, X-SOME-OTHER***

Server responds with whatever headers the code has set (plus default of its own)

Now the browser checks the **Access-Control-Allow-Origin**, **Access-Control-Allow-Methods** to make sure it **CAN** make the ACTUAL request.