

# Same Origin Policy (SOP)

Ebrahim Elmegazy

- \* X site can't get request from Y site because of SOP but if X site has completed 3 conditions
  - ① Same protocol http or https etc.
  - ② Same domain name
  - ③ Same port → http:80, https:443

→ if the three conditions are not successfully completed, it will show us an error called: Cross origin request blocked.  
and the site will not be able to read the response.

- \* but Cross Origin Resource Sharing (CORS) is a response header that allows cross origin request: Access-Control-Allow-Origin: \* → allow all requests.

Access-Control-Allow-Credentials: true  
header true لكي تكون قادر على ارسال اكواد  
عندما تفتح الموقع في نفس الصفحة او في متصفح مختلف  
لذلك في ارسال اكواد او الامر الارجع الى المتصفح  
يكون login field في طارحان تكون فيه اكواد اسفله.

- \* How to hunt:

→ in request header add Origin: any.com

→ if you get a response header contains Access-Control-Allow-Origin: any.com & Access-Control-Allow-Credentials: true

Then the site is vulnerable.

The screenshot shows a browser window with the URL `http://185.45.192.228/cors/demo.html`. The page contains a button labeled "Exploit". Below the button is a script block:

```
<script>
  var http = new XMLHttpRequest();
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (xhttp.readyState == 4 && xhttp.status == 200) {
      document.getElementById("myPayLoad").innerHTML = xhttp.responseText;
    }
  };
  xhttp.open("GET", "https://benowinkelshuibleremote.nl/api/offers?countryCode=nl&languageCode=en", true);
  xhttp.send();
</script>
</body>
</html>
```

A red annotation highlights the word "my payload" above the script block.

always access-control-allow-origin: \*  
access-control-allow-credentials: true  
لكن ازاي ان وجوه كمان  
في نمرة :)

- \* It is header that gives permission to who can execute javascript code on my website.

- ## \* How to hunt.

same as CORS. try also `ayklam.targetdomain.com`  
it may works sometimes.