Cross-Origin Resource Sharing (CORS)

Jogesh K. Muppala

THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系

香港科技大學
THE HONG KONG UNIVERSITY OF
SCIENCE AND TECHNOLOGY

Same-Origin Policy

- Web app security model that restricts how a document or script loaded from one origin can interact with a resource from another origin
  - Isolating potentially malicious documents
- Origin defined by three tuple: (Protocol, host name, port number)

Same-Origin Policy

- Example: from page http://www.abc.com/xyz/page.html

URL	Outcome	Reason
http://www.abc.com/abc/page2.html	success	
http://www.abc.com/abc/def/page3.html	success	
https://www.abc.com/ghi/page4.html	failure	Different Protocol
http://www.abc.com:123/jkl/page5.html	failure	Different port
http://store.abc.com/mno/page6.html	failure	Different host

Cross-Origin Requests

- Cross-origin HTTP request: Accessing a resource from a different domain, protocol or port
- Browsers restrict cross-origin HTTP requests initiated from within scripts, e.g., XMLHttpRequest or Fetch

Cross-Origin Resource Sharing (CORS)

- Mechanism to give web servers cross-domain access controls
  - Browser and server can interact to determine whether or not it is safe to allow the cross-origin request
  - New set of HTTP headers that allow servers to describe the set of origins that are permitted to read the information using a web browser
    - Access-Control-Allow-Origin
    - Access-Control-Allow-Credentials
    - Access-Control-Allow-Headers etc.

Cross-Origin Resource Sharing (CORS)

- Simple cross-site requests:
  - GET or POST with request body containing application/x-www-form-urlencoded, multiplart/form-data or text/plain
  - No custom headers
  - For widely accessed resources like GET, can send back reply with Access-Control-Allow-Origin: * header
  - If need to restrict the access, then send reply with Access-Control-Allow-Origin: http://abc.com

Cross-Origin Resource Sharing (CORS)

- Preflighted Requests
  - Methods that can cause side-effects on server’s data: non GET or POST, or even POST with content-type other than mentioned earlier
  - Mandated to “preflight” the request by soliciting the server’s supported methods by sending a HTTP OPTIONS request method
  - Then upon “approval” from the server sending the actual request
  - Server response may include Access-Control-Allow-Methods, Access-Control-Allow-Headers, Access-Control-Allow-Credentials

Cross-Origin Resource Sharing (CORS)

- Credentialed Requests
  - Requests that are accompanied by Cookies or HTTP Authentication information
  - Server needs to respond with Access-Control-Allow-Credentials: true
  - Accces-Control-Allow-Origin header cannot have a wildcard “*” value, must mention a valid origin domain

CORS NodeModule

- Middleware to configure CORS with various options
- Installing

```
npm install cors --save
```

- Simple CORS enabling all CORS requests
- Enabling CORS for specific routes
- Configuration options for various headers