Misconfiguration CORS

1-مقدمة: الأول خلينا نفهم كام مصطلح من خلالهم الدنيا هتبقى سهلة ومفهمومة.

- Same Origin Policy: دي سياسيات محطوطة عشان لما موقع X يطلب يشوف الـ response بتاعة الموقع Y يعني بإختصار هي بتتحكم في إن الـ Cross Origin Request تحصل او لا وده من خلال بعض الشروط: 1- إن يكونوا نفس البروتوكول (Https/Http)

2- يكونوا نفس الـ Origin أو الـ Host

3- يكونوا نفس البورت 80 / 443

- لما بتبعت ريكوست HTTP من موقع لموقع تاني, بيتبعت من ضمن الريكوست أي كوكيز وسيشن كوكي مرتبطة بالموقع الي ريحاله الريكوست وبالتالي الـ Response هتكون مرتبطة بالسيشن الي اتبعتت وهتجيب أي داتا مهمة تبع اليوزر ده.
- الـ Same origin policy بيشتغل إزاي؟
- الـ SOP بيسمح إن الموقع بتاعك يجيب داتا من موقع تاني زي صورة، فيديو، أو حتى جافاسكريبت كود، ولكن بالرغم إن الحاجات دي تقدر تعملها Load في موقعك ولكن متقدرش تروح تقرأ المحتوى بتاع الـ Resource ده، يعني من الآخر إنت آخرك هتعمل Load ليها ف موقعك وبس.
- من الممكن إنك تخلي الـ SOP أسهل شوية بإستخدام الـ document. Domain دي بتخليك تسمح لموقع تاني من إنه يقدر يـ access موقعك ولكن لازم يكون ضمن النطاق بتاعك FQDN، مثال زي example.com و shop.example.com ف محتاج تخلي قيمة الـ document.Domain للأتنين هي example.com وبالتلي أي واحد منهم يقدر يـ access التاني، في الأول كان ممكن تخلي قيمة الـ document.Domain هي الـ TLD زي مثلا com. وبالتالي كل المواقع الي آخرها com. تقدر تتواصل مع بعضها وتقرأ الـ response بتاعة المواقع التانية ولكن ده مبقاش موجود في المتصفحات الحديثة.

- Cross Origin Request: لما موقع X يطلب يشوف response موقعY
- Cross Origin Resource Sharing: دي بقى وظيفتها إنها بتستثني بعض المواقع أو كلها من إنها تقرأ الـ responseمن موقع بالرغم من عدم تحقق شروط الـ Same Origin Policy.
- Content Security Policy: ده بيحدد مين من المواقع يقدر ينفذ أكواد جافاسكريبت عندك ف الموقع وبالتالي بتتفادى أي موقع خارجي او Attacker إنه ينفذ اكواد جافا سكريبت حتى لو في ثغرة XSS.

مثال: لو انا مصنع كبير وعندي داتا كبيرة ف عملت API تقدر باقي المواقع تاخده وتحطه عندها, طيب لو افترضنا إن موقع جيه يقرأ الـ response بتاعة موقع المصنع ده, أكيد هيكونوا مش نفس الـ Origin وبالتالي الـ Same Origin Policyهتمنع ده ف لو احنا عايزين نقول يا متصفح اسمح للموقع ده إنه يقرأ عادي ف هتدخل الـ CORS بقى

2-تعريفها: هي ثغرة بتحصل لما الـ Developer يضيف الهيدر الي اسمه Access-control-allow-credentials:true وعشان تتأكد من وجود الثغرة لازم تلاقي الـ 2 Header دول:

Access-Control-Allow-Origin 😏

Access-Control-Allow-Credentials:true

بعدها بقى بتبعت هيدر اسمه Origin وتكتب فيه أي موقع عشان تتأكد.

3-اسبابها:

- بسبب إن الهيدر بتاع Access-Control-Allow-Credentials:true ده بيسبب إن عشان تقدر ال responseبتاع الموقع او الصفحة الفلانية لازم تكون Authenticated وده معناه إنك ممكن تقرأ داتا بتاعة شخص تاني او معلومات مهمة من صفحة.

==========

== عشان تكون في مجموعة الاوائل لازم تكون من الاوائل (same origin policy), بس لو انت مش من الاوائل وعايز تدخل يبقى لازم معاك واسطة للمدرس ده وهتقدر تدخل المجموعة (cros origin resource sharing)