Cross Origin Resource Sharing

Christopher Christensen

CORS

Cross Origin Resource Sharing

- bietet eine Möglichkeit, um Cross-Origin Zugriffe verantwortungsvoll und sicher zu lösen
- ein Mechanismus, um auf ausgewählte Ressourcen von einem Server mit einer anderen Origin zuzugreifen
- zusätzliche HTTP-Header verwendet, welche dem Browser mitteilen, dass eine Webanwendung die Berechtigung dazu hat
- Aus Sicherheitsgründen beschränken Browser HTTP-Anfragen, die aus Skripten (JavaScript) heraus initiiert werden und auf eine andere Origin zugreifen möchten = Same Origin Policy (SOP)
- Dies bedeutet, dass Webanwendungen grundsätzlich nur HTTP-Ressourcen vom gleichen Ursprung bzw. der gleichen Origin anfordern können, es sei denn, die Antwort von der anderen Origin enthält die richtigen CORS-Header

Welche Anfragen verwendet CORS?

- Cross-Origin Sharing Standard verwendet, um standortübergreifende HTTP-Requests zu ermöglichen, bsp., für:
  - Aufrufe von XMLHttpRequests- oder Fetch-API's
  - Webfonts
  - WebGL-Texturen (Web Graphics Library)
  - Bilder / Videos
  - Stylesheets
  - Scripts

Wie funktioniert CORS?

- durch Hinzufügen neuer HTTP-Header, die es Servern ermöglichen, die Herkunft der Informationen zu beschreiben, die über einen Webbrowser gelesen werden dürfen
- Im einfachsten Fall beginnt die domainübergreifende Kommunikation damit, dass ein Client eine GET-, POST- oder HEAD-Anfrage für eine Ressource auf dem Server stellt. Die Anfrage enthält einen Origin-Header, der den Ursprung des Client-Codes angibt. Der Server berücksichtigt den Ursprung der Anfrage und erlaubt oder verbietet diese. Wenn der Server die Anfrage zulässt, antwortet er mit der angeforderten Ressource und einem "Access-Control-Allow-Origin"-Header in der Antwort. Dieser Header zeigt dem Client an, von wo aus der auf die Ressource zugreifen darf
- Heisst also: Falls der Access-Control-Allow-Origin-Header mit dem Ursprung der Anfrage übereinstimmt, erlaubt der Browser die Anfrage. Falls dies nicht der Fall ist, wird die Anfrage entsprechend verboten
- Wieso kann JS bestimmten von wo der Origin ist?

Preflight Requests

- Wenn eine Anfrage Auswirkungen auf die Benutzerdaten haben kann, reicht eine einfache Anfrage (simple request) nicht aus
- Stattdessen wird vor der eigentlichen Anfrage ein Preflight-CORS-Request gesendet, um sicherzustellen, dass die eigentliche Anfrage sicher gesendet werden kann
- Preflight-Anfragen sind notwendig, wenn die eigentliche Anfrage eine andere HTTP-Methode als GET, POST oder HEAD ist, oder wenn der Inhaltstyp einer POST-Anfrage etwas anderes als application/x-www-form-urlencoded, multipart/form-data oder text/plain ist
- Wenn die Anfrage benutzerdefinierte Header enthält, ist eine Preflight-Anfrage erforderlich