**Opdracht 4a SOP/CORS/CSP**

1. vraag over SOP

Geef antwoord op deze vraag. Een webpagina op de site web.nl bevat dit html-formulier. Kan deze data verstuurd worden?

    <form action="https://example.com/results.json">

        <label for="email">Enter your email: </label>

        <input type="email" name="email" required>

        <button type="submit">Subscribe</button>

    </form>

Antwoord: Ja, formulierdata kun je zenden naar een cross-origin url, zoals vermeld in de action-attribuut van het <form>

1. vraag over SOP

Geef antwoord op deze vraag. Een webpagina op de site web.nl bevat dit iframe. Is dit toegestaan? Waarom?

<iframe src="https://voorbeeld.nl/pagina.html" alt="VB iframe"></iframe>

Antwoord: Cross-origin iframes zijn toegestaan tenzij de origin (voorbeeld.nl) in de HTTP-header de X-Frame-Options op deny of sameorigin heeft ingesteld

**INLEVEROPDRACHT:**

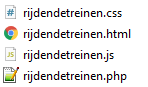
Kijk naar deze websites. Wat is de CORS instelling (access-control-allow-origin) en wat de Content-Security-Policy. Deze kun je vinden met DevTools, tabblad Network.

Vul de tabel in en lever in op itslearning.

|  |  |  |
| --- | --- | --- |
| Site | CORS | CSP |
| belastingdienst.nl | X | |  |  | | --- | --- | | Content-Security-Policy | default-src [https://\*.belastingdienst.nl](https://*.belastingdienst.nl) <https://vinden.belastingdienst.nl> [https://\*.readspeaker.com](https://*.readspeaker.com); connect-src 'self' [https://\*.belastingdienst.nl](https://*.belastingdienst.nl) [https://\*.optimizely.com](https://*.optimizely.com) [https://\*.readspeaker.com](https://*.readspeaker.com) \*.abtasty.com; child-src 'self' <https://belastingdienst.nl> [https://\*.belastingdienst.nl](https://*.belastingdienst.nl) [https://\*.cdn.optimizely.com](https://*.cdn.optimizely.com) <https://secure.opinionlab.com> [https://\*.readspeaker.com](https://*.readspeaker.com) <https://www.anbi-instellingen.nl> <https://www.youtube.com> <https://www.youtube-nocookie.com>; frame-src 'self' [https://\*.belastingdienst.nl](https://*.belastingdienst.nl) [h…](https://*.readspeaker.com)'self' <https://n01d05.cumulus-cloud.com> [https://\*.readspeaker.com](https://*.readspeaker.com) <https://img.youtube.com> data: [https://\*.belastingdienst.nl](https://*.belastingdienst.nl) blob: data: \*.abtasty.com; font-src 'self' [https://\*.belastingdienst.nl](https://*.belastingdienst.nl) blob: data: \*.abtasty.com; script-src 'self' [https://\*.belastingdienst.nl](https://*.belastingdienst.nl) <https://cdn.optimizely.com> [https://\*.readspeaker.com](https://*.readspeaker.com) <https://bdtm.containers.piwik.pro> 'unsafe-eval' 'unsafe-inline' blob: \*.abtasty.com ; style-src 'self' [https://\*.belastingdienst.nl](https://*.belastingdienst.nl) [https://\*.readspeaker.com](https://*.readspeaker.com) \*.abtasty.com 'unsafe-inline' | |
| knmi.nl | X | X |
| summacollege.nl | X | X |
| w3schools.com | X | |  |  | | --- | --- | | content-security-policy | frame-ancestors 'self' <https://mycourses.w3schools.com>; | |
| rijdendetreinen.nl/feed.json | X | X |
| api.open-notify.org/iss-now.json | \* | X |

**Opdracht 4b CORS**

Haal rijdendetreinen.zip op, bekijk de bestanden en controleer of ze werken.



Wanneer je rijdendetreinen.html start dan zal deze geen data tonen. Waarom niet? Kijk in DevTools bij Network en bij Console.

Wanneer je rijdendetreinen.php start dan werkt het wel. Waarom wel? Kijk in DevTools bij Network en bij Console.

Ga nu iets bijzonders doen. Maak het bestand proxytreinen.php met de volgende inhoud:

<?php

header('access-control-allow-origin: \*');

header('content-type: application/json');

echo file\_get\_contents('https://www.rijdendetreinen.nl/feed.json');

Verander in rijdendetreinen.js de url waarnaar de asynchrone request wordt verstuurd, ongeveer als volgt:

//const api = 'https://www.rijdendetreinen.nl/feed.json'

const api = 'http://localhost/proxytreinen.php'

De asynchrone request loopt nu via de proxy en rijdendetreinen.html werkt nu wel. Waarom?

**INLEVEROPDRACHT CORS:**

Maak rijdendetreinen.html werkend met behulp van de proxy.

Maak een screen shot, waarin je laat zien dat het werkt. Dat doe je door DevTools te openen in tabblad netwerk. Daarin moet te zien zijn dat o.a. deze bestanden gebruikt worden:

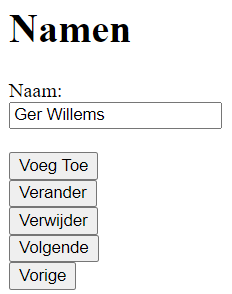
* rijdendetreinen.html
* rijdendetreinen.css
* rijdendetreinen.js
* proxytreinen.php

Screen shot inleveren op itslearning.

**Opdracht 4c Preflight**

1. Neem apipreflight.zip en preflight.html van itslearning en pak de api uit.
2. Start de API
3. Controleer de response op <http://127.0.0.1:8000/api/namen/> (adres van de laravel deployment server)  
   Deze response zou een JSON-lijst van namen moeten zijn.
4. Open preflight.html in de browser.  
   (misschien is het nodig om met een editor in preflight.html de url van api aan te passen.   
   Deze staat nu op const api = <http://127.0.0.1:8000/api/>namen/)

Als het goed is krijg je deze webpagina:



Met deze applicatie kun je berichten uitwisselen met de api.

In DevTools kun je zien dat voor toevoegen (post request) geen preflight nodig is, maar voor veranderen (patch request) en verwijderen (delete request) wel.

**INLEVEROPDRACHT preflight:**

Maak behulp van DevTools van de preflights (patch en delete) een screen shot waarin je de aanwezigheid van de preflight laat zien. Het moet een OPTIONS-request zijn, waarin vermeld staat welke type request is/zijn toegestaan.

In de screen shot moet zichtbaar zijn:

* Request URL
* Request Method
* Access-Control-Request-Method (request header)
* Access-Control-Allow-Methods (response header)
* Access-Control-Allow-Origin

Lever in op itslearning