**How to use CORS in Node.js with Express**

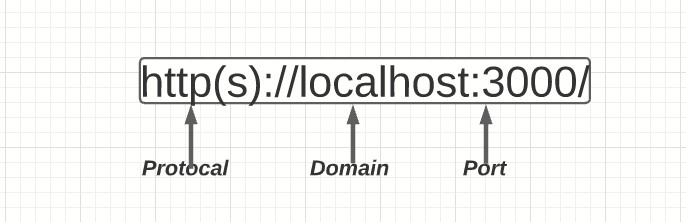
May 6, 2021

* Topics:
* [Node.js](https://www.section.io/engineering-education/topic/node.js/)
* [API](https://www.section.io/engineering-education/topic/api/)

Node.js is an open-source and cross-platform runtime used when executing JavaScript code on the server-side. One of the popular Node.js server frameworks is Express. Implementing CORS in Node.js helps you access numerous functionalities on the browser.

Express allows you to configure and manage an HTTP server to access resources from the same domain.

The three parts that form an [origin](https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Origin) are protocal, domain, and port.



Today, there are many applications that depend on APIs to access different resources. Some of the popular APIs include weather, time, and fonts. There are also servers that host these APIs and ensure that information is delivered to websites and other end points. Therefore, making cross-origin calls, is a popular use case for the modern web application.

Let’s say accessing images, videos, iframes, or scripts from another server. This means that the website is accessing resources from a different origin or domain. When building an application to serve up these resources with Express, a request to such external origins may fail. This is where CORS comes in to handle cross-origin requests.

Goal

This guide will help you learn how to configure CORS with Express.

Prerequisites

To follow this article along, prior knowledge of [Node.js](https://www.youtube.com/watch?v=fBNz5xF-Kx4) and [Express](https://www.section.io/engineering-education/express/) is essential.

What is CORS?

CORS stands for Cross-Origin Resource Sharing. It allows us to relax the security applied to an API. This is done by bypassing the Access-Control-Allow-Origin headers, which specify which origins can access the API.

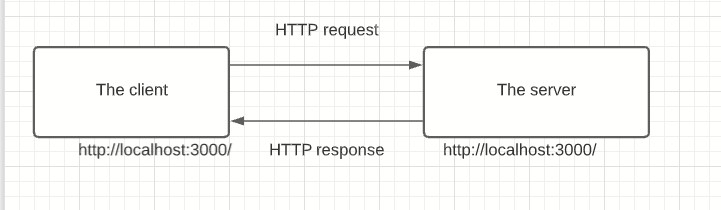
In other words, CORS is a browser security feature that restricts cross-origin HTTP requests with other servers and specifies which domains access your resources.

Check this guide to learn more about the [CORS policy](https://www.section.io/engineering-education/what-is-cors-policy/).

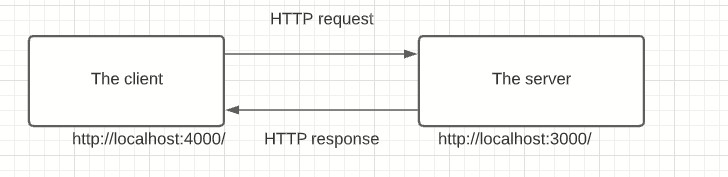
How CORS works

An API is a set procedure for two programs to communicate. This means that API resources are consumed by other clients and servers.

Here are two scenarios:



The client and the server have the same origin. In this example, accessing resources will be successful. You’re trying to access resources on your server, and the same server handles the request.



The client and server have a different origin from each other, i.e., accessing resources from a different server. In this case, trying to make a request to a resource on the other server will fail.

This is a security concern for the browser. CORS comes into play to disable this mechanism and allow access to these resources. CORS will add a response header access-control-allow-origins and specify which origins are permitted. CORS ensures that we are sending the right headers.

Therefore, a public server handling a public API will add a CORS related header to the response. The browser on the client machine will look at this header and decide whether it is safe to deliver that response to the client or not.