

Title: XMLHttpRequest cannot load

Post Body:

I've written a spring boot MAVEN application which runs on embedded tomcat server(port: 8080). I've another nodeJS express server running on port 8000. Now, from port 8000, I've proxied the port 8080. So, any address '<http://localhost:8080/>' will take me to the code in spring boot application. I'm testing this by keeping the breakpoint in the required location in spring boot application.

But, after the code in the spring boot is executed, I'm getting

XMLHttpRequest cannot load [http://localhost:8080/users/?\\_=1468052674994](http://localhost:8080/users/?_=1468052674994). No 'Access-Control-Allow-Origin' header is present on

[/users/](#) is my request mapping in the spring boot app.

I just got to know that I'm supposed to 'EXPOSE' the port 8080 to the host.

<https://www.wouterdanes.net/2014/04/11/continuous-integration-using-docker-maven-and-jenkins.html> Above article says that docker can do this. I didn't understand much to start with this. Can anybody please help me.

Accepted Answer: None

Highest Rated Answer:

The XMLHttpRequest is used by browsers to create Ajax requests. To prevent abuse, browsers use a 'Same Origin Policy' to prevent loading resources from different locations as the page, that causes the request. Same IP, but with different ports, as in your case, is also treated as 'different locations'.

You can solve this problem using special headers on the page loading the request. This may sound very complicated, but its is understandable explained at

[Same-origin\\_policy](#) and [Cross-origin\\_resource\\_sharing](#)

### **Solution**

The first response, with the page that will make the XMLHttpRequest, must include a header :

Access-Control-Allow-Origin: <http://localhost:8080>