

THE OFFICIAL IONIC BLOG (/)

The next generation HTML5 hybrid app development SDK

Everything (/)

Top Posts
(/tag/top-posts)

Framework
(/tag/framework)

Cloud
(/tag/cloud)

Tutorials
(/tag/tools)

Built With Ionic
(/tag/built-with)

Handling CORS issues in Ionic (<http://blog.ionic.io/handling-cors-issues-in-ionic/>)



By Josh (<http://twitter.com/jbavari>) on February 24, 2015 IONIC

Tweet



([HTTP://BLOG.IONIC.IO/TAG/IONIC/](http://BLOG.IONIC.IO/TAG/IONIC/)) TOOLS ([HTTP://BLOG.IONIC.IO/TAG/TOOLS/](http://BLOG.IONIC.IO/TAG/TOOLS/))

If you've used `ionic serve` or `ionic run` with live reload and accessing external API endpoints, chances are you've run into some CORS (http://en.wikipedia.org/wiki/Cross-origin_resource_sharing) issues. They usually look something like this:

```
XMLHttpRequest cannot load http://api.ionic.com/endpoint.  
No 'Access-Control-Allow-Origin' header is present on the requested resource.  
Origin 'http://localhost:8100' is therefore not allowed access.
```

So what is CORS, and why is it a problem?

What is CORS?

CORS = Cross origin resource sharing.

The `origin` is the host you are currently viewing.

Since you're at `http://ionicframework.com/blog/handling-cors-issues-in-ionic`, the `origin` is `ionicframework.com`.

Say we send an AJAX request to `http://cors.api.com/api`, your host `origin` will be specified by the `Origin` header that is automatically included for CORS requests by the browser. Since `ionicframework.com` does not match the host of `api.com`, our request from `ionicframework.com` must ask the server for approval before we can access the resource, in the form of an `HTTP OPTIONS` request header.

If we get the error above, then we may not access the resource from the server.

Let's take a look what your `origin` will be when you're running your app via `ionic serve`, `ionic run`, or `ionic run -l`.

Running in the browser

What happens when you run `ionic serve`?

- A local web server is started up.
- Your browser is opened to point at the local server address.

This starts you off looking at your app loaded in a browser on your computer with the address `http://localhost:8100` (if you chose localhost).

Your origin will be `localhost:8100`.

Any AJAX request sent out to a host other than `localhost:8100` will have `localhost:8100` as its origin and thus will require a CORS preflight request to see if it can access the resource.

Running on a device

What happens when you run `ionic run`?

- Your files for the app are copied to the device (or simulator).
- The app runs, thus firing a browser on the phone/simulator to run the files that were copied over, something like:

`file://some/path/www/index.html`.

Your origin will not exist, since you are running off of a `file://` URI; therefore, any request outwards will *not* require a CORS request.

Running on a device with livereload

What happens when you run `ionic run -l`?

- A local web server is started up.
- The app runs, thus firing a browser on the phone/simulator to run the files from the server `http://192.168.1.1:8100` (or whatever your local

IP address is).

Your origin will be `192.168.1.1:8100` .

Any AJAX request sent out to a host other than `192.168.1.1:8100` will require a CORS preflight request to see if it can access the resource.

Dealing with CORS in Ionic

CORS is only an issue when we are running or testing our app when running `ionic serve` OR `ionic run -l` .

There are two ways to solve the issue: The first, and easier, solution is to just allow all origins from your API endpoint. However, we can't always control the endpoint we are accessing. What we need, then, is a request that does not specify an `origin` .

We can do this by using a proxy server. Let's look how the Ionic CLI provides an easily configurable proxy server.

The Ionic CLI proxy server

A quick definition about proxies:

In computer networks, a proxy server is a server (a computer system or an application) that acts as an intermediary for requests from clients seeking resources from other servers.

What we'd need to do to get around these CORS issues is have a proxy server that will take our requests, issue a new request to the API endpoint, receive the response, and forward it back to our app so we can get around CORS issues.

The Ionic CLI introduced the ability to have a proxy server issue requests for you to get around any CORS issues you may have.

Since the server is sending a fresh request to your destination, there will be no `origin` and therefore, no CORS needed. It is important to note that the browser adds in the Origin header.

Setting up the proxy server

Please note, these are only needed for `ionic serve` and `ionic run -l`.

First we will need to set up the proxies in our `ionic.project` file. This will tell our Ionic server to listen to those paths and forward those requests on to the destination url.

In our app, we will need to replace our endpoint URLs to be set to the proxy server address for when we are running `serve` or `run -l`.

We can make this a bit easier by using some gulp tasks using the `replace` module to swap out the URLs.

The suggested method is to set up an Angular Constant to point at the API we're trying to proxy out.

This is the approach we will take below. We will also set up an Angular Service to use that API Endpoint to get data.

Setting up the proxy urls

Let's say we want to access `http://cors.api.com/api`, which is not allowing our origin from `localhost`.

The proxies settings contain two things: the `path` you use to access them on your local Ionic server, and the `proxyUrl` you'd ultimately like to reach from the API call.

Set up your `ionic.project` file to be something like:

```
{
  "name": "proxy-example",
  "app_id": "",
  "proxies": [
    {
      "path": "/api",
      "proxyUrl": "http://cors.api.com/api"
    }
  ]
}
```

Run your server with `ionic serve`.

As we specified above, when you access the ionic server at the path `http://localhost:8100/api`, it will proxy requests out to `http://cors.api.com/api` on your behalf.

Thus, no CORS is required.

Set up Angular Constant

It's quite easy to set up your API endpoints as Angular Constants.

Below, we've specified the `ApiEndpoint` to be our proxied URL right now.

Later, we can use our production url as a constant.

```
angular.module('starter', ['ionic', 'starter.controllers', 'starter.services'])
  .constant('ApiEndpoint', {
    url: 'http://localhost:8100/api'
  })
// For the real endpoint, we'd use this
// .constant('ApiEndpoint', {
//   url: 'http://cors.api.com/api'
// })
```

Once this is done, you can use the constant anywhere in your app, by including

`ApiEndpoint` as a dependency, as shown below in the service.

Setting up Angular Service

```
angular.module('starter.services', [])

//NOTE: We are including the constant `ApiEndpoint` to be used here.
.factory('Api', function($http, ApiEndpoint) {
  console.log('ApiEndpoint', ApiEndpoint)

  var getApiData = function() {
    return $http.get(ApiEndpoint.url + '/tasks')
      .then(function(data) {
        console.log('Got some data: ', data);
        return data;
      });
  };

  return {
    getApiData: getApiData
  };
})
```

Automating URL switches with Gulp

For this process, we'll need to modify our `gulpfile.js` to add in two tasks to add our proxy url or remove our proxy url.

Start by first installing the `replace` module – `npm install --save replace`.


```
// `npm install --save replace`  
var replace = require('replace');  
var replaceFiles = ['./www/js/app.js'];  
  
gulp.task('add-proxy', function() {  
  return replace({  
    regex: "http://cors.api.com/api",  
    replacement: "http://localhost:8100/api",  
    paths: replaceFiles,  
    recursive: false,  
    silent: false,  
  });  
})  
  
gulp.task('remove-proxy', function() {  
  return replace({  
    regex: "http://localhost:8100/api",  
    replacement: "http://cors.api.com/api",  
    paths: replaceFiles,  
    recursive: false,  
    silent: false,  
  });  
})
```

Final Words

This tutorial showed you one way to handle your CORS issues when running `ionic serve` OR `ionic run -l`.

We know that it can be a hassle to swap out your API url endpoints when switching between `ionic serve` and `ionic run -l`. One thing that could be suggested is to have a gulp startup process.

The easiest way to handle the CORS problem is to ultimately ask your API provider to allow all hosts. However, this isn't always an option.

Using the Angular constant and the replace module will give us a happy medium, in which we can work around CORS.

If you need a solid example, take a look at this sample project (<http://github.com/driftyco/ionic-proxy-example>).

This is everything you'd need to access an API server that restricts CORS.

If you have any questions, issues, or ideas, please leave a comment below or contact us on twitter (<http://twitter.com/ionicframework>) or github (<http://github.com/driftyco/ionic-cli>).

Tweet

G+1

48

Like 46

8

Join our list to get Ionic news and updates just like this one!

SUBSCRIBE!

Star

26,399

Follow @ionicframework

59.1K followers

47 Comments

Ionic Framework

1 Login ▾

♥ Recommend 8

🔗 Share

Sort by Best ▾



Join the discussion...

**Mirko Nasato** • a year ago

Since CORS is only an issue when running your app in development mode with ionic serve, and not when running as a mobile app packaged with Cordova, a simpler option is to just disable CORS altogether for local development. For Chrome for example there's a plugin called "Allow-Control-Allow-Origin: *" that lets you disable CORS.

8 ^ | ▾ • Reply • Share ›


**Dayan Lemos** ➔ Mirko Nasato • 2 months ago

It didn't work to me. I'm trying to access a redmine application.

^ | ▾ • Reply • Share ›

**Dian Ariyanto** ➔ Mirko Nasato • 3 months ago

It didn't work to me. I'm trying to access a redmine application.

 I nanksss..... work perfectly!

^ | v • Reply • Share ›



anjanigupta12 → Mirko Nasato • 4 months ago

Not working

^ | v • Reply • Share ›



david lyons → Mirko Nasato • 4 months ago

Amazing!! thank you very much! I was getting crazy with proxies, hosts files, dns. And that's it! easy peasy!!!

^ | v • Reply • Share ›



This comment was deleted.



Mirko Nasato → Guest • a year ago

Here's a step-by-step guide:

1. Install this Chrome plugin: <https://chrome.google.com/webs...>
2. There is not step 2.

:)

3 ^ | v • Reply • Share ›



Abhinav Gupta → Mirko Nasato • 5 months ago

Sweet and simple, I was trying to find a way to disable this thing via chrome://flags but this plugin is way easier to pass along my dev team. Thanks **@Mirko Nasato** !

^ | v • Reply • Share ›



Julien Monty → Mirko Nasato • 9 months ago

You saved my day :)

^ | v • Reply • Share ›



Douglass Hock • a year ago

I am attempting to use Amazon AWS Javascript for Browser SDK. When running my ionic app in the browser my calls to AWS work. When I load the app in either the emulator/simulator or on the device I get a 403 response when attempting to call the Amazon AWS functions. The on difference I see on the request header is the Origin. On the desktop browser it is set to http://localhost:8100, but on the request sent from the emulators or devices that Origin is set to file://. Everything I have read so far would suggest that I would be having 403 issues with the desktop browser and not the app running on the device.

2 ^ | v • Reply • Share ›



Slash → Douglass Hock • 10 months ago



I would appreciate if you had the solution for this to share it here. I have the same exact problem!

^ | v • Reply • Share ›



Douglass Hock → Douglass Hock • a year ago

Too make it easier for someone to assist, I have setup a github repository (<https://github.com/doughock/aw...>). This is a simple ionic blank app with the Amazon EC2 call in the index.html file.

Synopsis

This project is to help demonstrate a 403 Forbidden response when attempting to call describeInstances() from within an Ionic app on either an Android or iOS device (or in emulators). This same application, when run from a desktop browser, works and returns the expected results.

^ | v • Reply • Share ›



Douglass Hock → Douglass Hock • a year ago

Ok. Here is another update. I wrote a small test html page that just includes the Amazon SDK and was able to run this file both in the desktop browser and on the browser on my Android and iPhone devices. So, now I am wondering if the issue I am having is related to either ionic framework or more specifically AngularJS.

^ | v • Reply • Share ›



Slash → Douglass Hock • 10 months ago

I am having the same issue. Unfortunately, I am not finding supports or answers to solve this. did you manage to solve it?

^ | v • Reply • Share ›



Douglass Hock → Slash • 10 months ago

I am in the same situation... not finding support or answers. Thanks for sharing your feedback.

^ | v • Reply • Share ›



Rob_McB • a year ago

Thanks for this solution I was able to get it working with either my web browser on my machine while going through the proxy OR via a mobile when removing the proxy.

I have added the Gulp tasks to replace the proxy. I assume when I now serve my Ionic app I have to specify the add-proxy task somehow and then when I build to iOS or Android then I have to run the remove-proxy task?

Are you able to clarify if this is required and if so how to do it?

Thanks

2 ^ | v • Reply • Share ›



Rajendra Khope • 4 days ago

What if I've multiple endpoint server. As I work with WiFi sensors, I need to control all from one app. How to add multiple endpoint proxyUrl?

^ | v • Reply • Share ›



Ntando Masondo • 8 days ago

you saved my day Mirko Nasato. this has been my problem for a few months

^ | v • Reply • Share ›



Andreas Pfr • a month ago

Thank you for the article. I followed the steps you provided but I still have a CORS issue: XMLHttpRequest cannot load file:///.../www/modules/home/view/home.html. Cross origin requests are only supported for HTTP.

^ | v • Reply • Share ›



Kieran Desmond • a month ago

Hi, great resources and tips. Just wondering how I could apply the above when accessing google api's. They accept localhost though not any up Base. For example, if I use ionic run android -l it uses the phone's IP - 192.168.x.0.1 as an example. This is rejected by Google except in ionic serve on my laptop as it's localhost. Can I somehow redirect to localhost on the phone without an IP? Really appreciate any help. Relatively new to ionic.

I've tried creating a proxy on one of my servers though I'd prefer to use ionic proxy if I can! Thanks all!

^ | v • Reply • Share ›



Davo • 3 months ago

With WKWebView, CORS is also an issue while running the application on devices. Any chance you will extend the fantastic use of your proxy to let us include it in our built applications?

^ | v • Reply • Share ›



hugo blanc • 3 months ago

Is it working with ionic V2 ?

^ | v • Reply • Share ›



Prashant Mittal • 4 months ago

'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'http://localhost:8100' is therefore not allowed access. If an opaque response serves your needs, set the request's mode to 'no-cors' to fetch the resource with CORS disabled.

I have configured my .htaccess file using :

Header add Access-Control-Allow-Origin "*"

Header add Access-Control-Allow-Methods: "GET POST OPTIONS DELETE PUT"

Header add Access-Control-Allow-Methods: GET,POST,OPTIONS,DELETE,PUT

Once I was testing the application it runs fine on the browser but after sometime it start showing the same CORS error.

can any suggest me the solution for this ?

^ | v • Reply • Share ›



Chirag thaker • 5 months ago

proxyUrl tag in ionic.project working in "ionic serve" browser but when trying "ionic run -lc android" not working.

^ | v • Reply • Share ›



Chirag thaker • 5 months ago

work like charm !!

^ | v • Reply • Share ›



dellagustin • 6 months ago

Hello, I solved this problem putting an apache server between my ionic app and my server where I was consuming services with cross domain requests.

The important part of the apache config file looked like:

Redirec the rest requests

ProxyPass /rest http://localhost:8080/MyApp/rest

Adjusts the cookies path

ProxyPassReverseCookiePath /MyApp /

>>> CORS Headers for testing mobile app in the browser

>>>> Copy information from request to response

Must accept "allow credentials" so that the cookies are passed

When performing a cross domain request.

see <http://stackoverflow.com/quest...>

Header set Access-Control-Allow-Credentials "true"

Connect to "http://localhost:8080/MyApp/rest" with Credentials in HTTP Request

[see more](#)

^ | v • Reply • Share ›



Alan Barboza • 6 months ago

That's saved my day, thanks!

^ | v • Reply • Share ›



Shash • 8 months ago

i'am using rails server..and installed cors gems in the rails 4 server...Still getting the same....

^ | v • Reply • Share ›

^ | v · Reply · Share ›



Shash · 8 months ago

XMLHttpRequest cannot load

<https://accounts.google.com/o/...>

No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'null' is therefore not allowed access.

I'm getting the response from the server..i.e google redirection...but at the same time its giving the error..Any idea..?

^ | v · Reply · Share ›



Paul Lang · 8 months ago

In Ionic2 you would set the proxies in ionic.config.js instead of ionic.project

^ | v · Reply · Share ›



Jonathan · 8 months ago

Thanks a lot! you saved my day

^ | v · Reply · Share ›



Slash · 10 months ago

Unfortunately this methods don't work with Angular routing feature. Still ionic has a lot to do in regards to local views. Notting that not all apps are server hosted or located. I have a nested app with hundreds of views routed through angular ui-inlcude so imagine...

^ | v · Reply · Share ›



Luc BONNIN · a year ago

Thanks a lot, you saved my day.

I've not used the gulp way. i've use window.cordova in an if and it is working for both dev and prod. See the forum topic here :

<http://forum.ionicframework.co...>

^ | v · Reply · Share ›



Ken Whipday · a year ago

Also, you can disable CORS in Safari by enabling Develop (Preferences -> Advanced) then on the menu selecting Develop; Disable Cross-Origin Restrictions - courtesy of Nic Raboy, there are examples for other browsers on his blog <https://blog.nraby.com/2014/o...>

^ | v · Reply · Share ›



KB · a year ago

Hello there, I am new to Ionic and the whole development. I have this error, can you explain what is it please?

WebSocket connection to 'ws://localhost:35720/livereload' failed: Error in connection

Establishment: net::ERR_CONNECTION_REFUSED

Also thank you, this is really useful. Now I start to understand a little more of what is going.

^ | v • Reply • Share ›



Chris Tiearney • a year ago

I'm having a problem with this setup. I've setup my .project file's proxy with path "/api" and proxyUrl "http://localhost:8000/api" which is a Laravel app we are using for JWT authentication. When I try to make a POST request to "/api/authenticate" I get the following error in my ionic serve console output:

? Error: connect ECONNREFUSED

at exports._errnoException (util.js:746:11)

at TCPConnectWrap.afterConnect [as oncomplete] (net.js:1010:19)

Any ideas?

^ | v • Reply • Share ›



Lucas Cardoso Zago → Chris Tiearney • 9 months ago

Same problem here, did you solve it?

^ | v • Reply • Share ›



Chris Tiearney → Lucas Cardoso Zago • 9 months ago

ended up using a google chrome plugin. its been mentioned a lot in the comments here. see [@Mirko Nasato](#)'s comment:

"For Chrome for example there's a plugin called "Allow-Control-Allow-Origin: *" that lets you disable CORS"

Just remember to disable the plugin after you're done with it as by default it affects all sites, not just your API :)

^ | v • Reply • Share ›



Jinett Andrea Morales Rivera • a year ago

Hello and tried this configuration did not work me. Someone tried it and it worked? Please

^ | v • Reply • Share ›



Nicolás Schürmann Lindemann • a year ago

Works great the proxies. But at least for me is not true that is not needed when emulated. My app is throwing an error when i try to makes API calls from the emulator.

^ | v • Reply • Share ›



Nicolás Schürmann Lindemann • a year ago

great!, what happen if the url doesn't have a prefix?, IE i want to make requests to www.bla.com and the url that i want to query is www.bla.com/here ???

^ | v • Reply • Share ›



Andrea • a year ago

its doesn't works for my app project: I'm trying to call a webservice to create a ion list.

I followed the tutorial and this are my setting:

ionic.project:

```
{
  "name": "lego",
  "app_id": "",
  "proxies": [
    {
      "path": "/api",
      "proxyUrl": "http://brickset.com/web-service..."
    }
  ]
}
```

gulpfile.js:

```
var replace = require('replace');
var replaceFiles = ['./www/js/app.js'];
```

```
gulp.task('add-proxy', function() {
```

[see more](#)

^ | v • Reply • Share ›



Nabi • 2 years ago

Josh - this works like a charm!!! Thank you so much for your effort!

However, there is one issue which we are facing for our development on localhost: redirecting to https with self signed or untrusted certificates fail with the following error message on the cli output:

```
✗ Error: self signed certificate
at Error (native)
at TLSSocket.<anonymous> (_tls_wrap.js:927:36)
at TLSSocket.emit (events.js:104:17)
at TLSSocket._finishInit (_tls_wrap.js:458:8)
```

That means we currently have no chance to connect to any https server via the proxy because it has an invalid certificate. I believe there is a good reason to add some configuration that allows accepting even invalid certs, here is our use case:

1. we use ionic serve
2. we use this config:

[see more](#)

^ | v • Reply • Share ›

**pliablepixels** → Nabi • a year ago

Nabi, did you find a workaround for this?

^ | v • Reply • Share ›

**Nabi** → pliablepixels • a year ago

pliablepixels, I created a pull request on Github (see [https://github.com/driftyco/ionic...](https://github.com/driftyco/ionic/pull/10000) and the champions behind ionic accepted my suggestion and the code I offered. Here is what you can do:

[http://forum.ionicframework.co...](http://forum.ionicframework.com/t/ionic-cors-issues/10000)

My last code snippet in that example is what you can do...

^ | v • Reply • Share ›

**pliablepixels** → Nabi • a year ago

Hi Nabi - thank you! I found your commit right after I posted this question and then I forgot where I posted the question so I could update it - Your commit saved me a lot of headaches - works perfectly :-)

^ | v • Reply • Share ›

**Ace Han** • 2 years ago

Hi, this article addresses the situation about developing a hybrid app on CORS issue. What if it comes to releasing a hybrid app to apk/dmg?

Let's say I do have control on the server side. I don't want ACCESS-CONTROL-ALLOW-ORIGIN set to *

and from stackoverflow like [http://stackoverflow.com/quest...](http://stackoverflow.com/questions/10000000/cors-allow-origin-to-null-to-address-scheme-with-file-scheme), it suggests set ACCESS-CONTROL-ALLOW-ORIGIN to null to address scheme with file:// situation

Won't that be equal to * on server side if any other hybrid app developer knows about your endpoint uri and sends CORS from his hybrid app?

I've been googling around and get no where, all the existing answers are suggesting to set ACCESS-CONTROL-ALLOW-ORIGIN to * which I think is vulnerable

As this article addresses CORS issue, any suggestion on this particular situation?

^ | v • Reply • Share ›

**Bhavik** • 2 years ago

I need to call a Js file that returns a JS function like DoSomething(JsonData). I'm planning to get this file using \$http.get on my /#/tab/home page, I've configured my project file as below. however, i'm still getting Cors error when I try ionic serve.

would someone please help me to resolve this issue. Also I'm wondering about the easiest way to handle the JS function (only content in the file is a JS function(that has JSON data as argument) that returns from the \$http.get to get the JSON data from JS function.

```
"proxies": [
{
"path": "/tab/home",
"proxyUrl": "http://otherdomain/api/"
}
]
like JS file returns doSomething(JSONDATAload);
^ | v • Reply • Share ›
```

ALSO ON IONIC FRAMEWORK

Layout the Cool Way: Using the Ionic 2 Grid Component

3 comments • 2 months ago •

Julian — There is one typoionic plugin add cordova-image-pickershould beionic plugin add cordova-plugin-image-pickerBeen ...

One MEAN Ionic 2 Todo App on Heroku, Part 2

8 comments • 2 months ago •

Marcin Gościcki — Same here :/EDIT:For some reason the error is caused by empty events values in view - REF ...

¿Hablas Español? Localize Your App with NG2-Translate!



8 comments • 2 months ago •


Aoki Sora — Thank you for the post. Here is boilerplates for ionic with ng2-translate. Might be helpful for ...

Ionic Build Process Updates

5 comments • 20 hours ago •

Bo83 — The quotes below should be the mission statement for all of web development. Good job guys! Do whatever ...

 [Subscribe](#)  [Add Disqus to your site](#) [Add Disqus](#) [Add](#)  [Privacy](#)

An Ionic (<http://ionic.io/>) creation.  [RSS \(/feed/\)](/feed/)