(https://ko-fi.com/C0C674BW)

◀ home (/)

Creating Serverless Plotly Chart Exports

This is a step by step tutorial on creating a chart URL API on AWS. At the end of this tutorial, you will have an API endpoint that renders plot.ly charts and outputs them to the browser.

Plot.ly is a javascript charting library for the frontend and won't work on the server without some help. To render charts on the backend, we are going to use PlotlyChartExport (https://github.com/mbejda/plotlychartexport). PlotlyChartExport (https://github.com/mbejda/plotlychartexport) is a chart rendering module that renders static plot.ly charts on the server. It's a lightweight module designed for serverless environments. There are other modules out there that can generate charts on the server, but they have dependencies that rely on graphics libraries such as Cairo.

PlotlyChartExport (https://github.com/mbejda/plotlychartexport) doesn't rely on those dependencies nor does it rely on those libraries. It uses phantom.js (scriptable browser) to render a Plot.ly chart which gets cropped, base64 encoded and returned in a resolving promise.

Getting Started

To create serverless chart exports we need to :

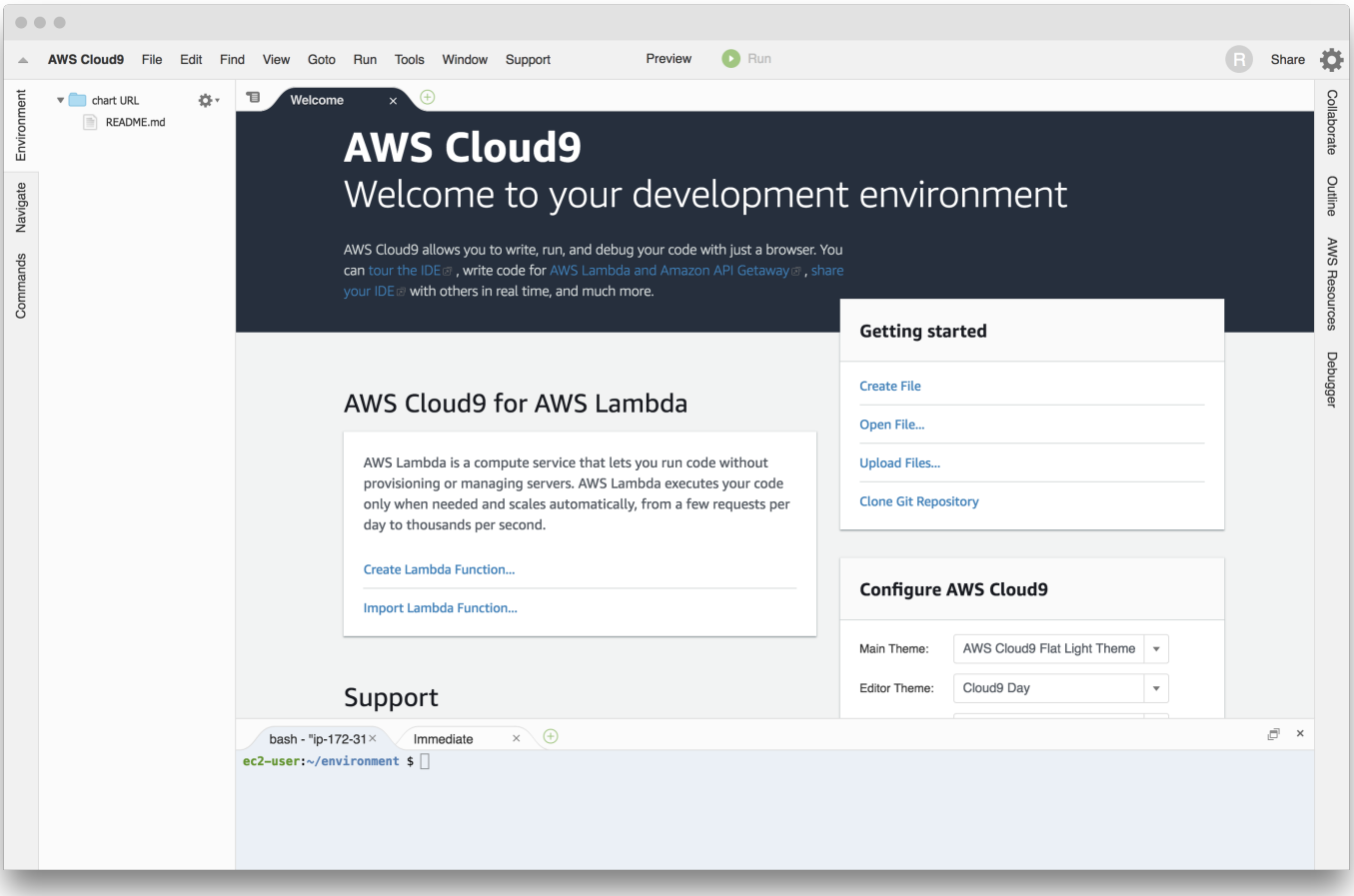
- Create a lambda function that renders a chart

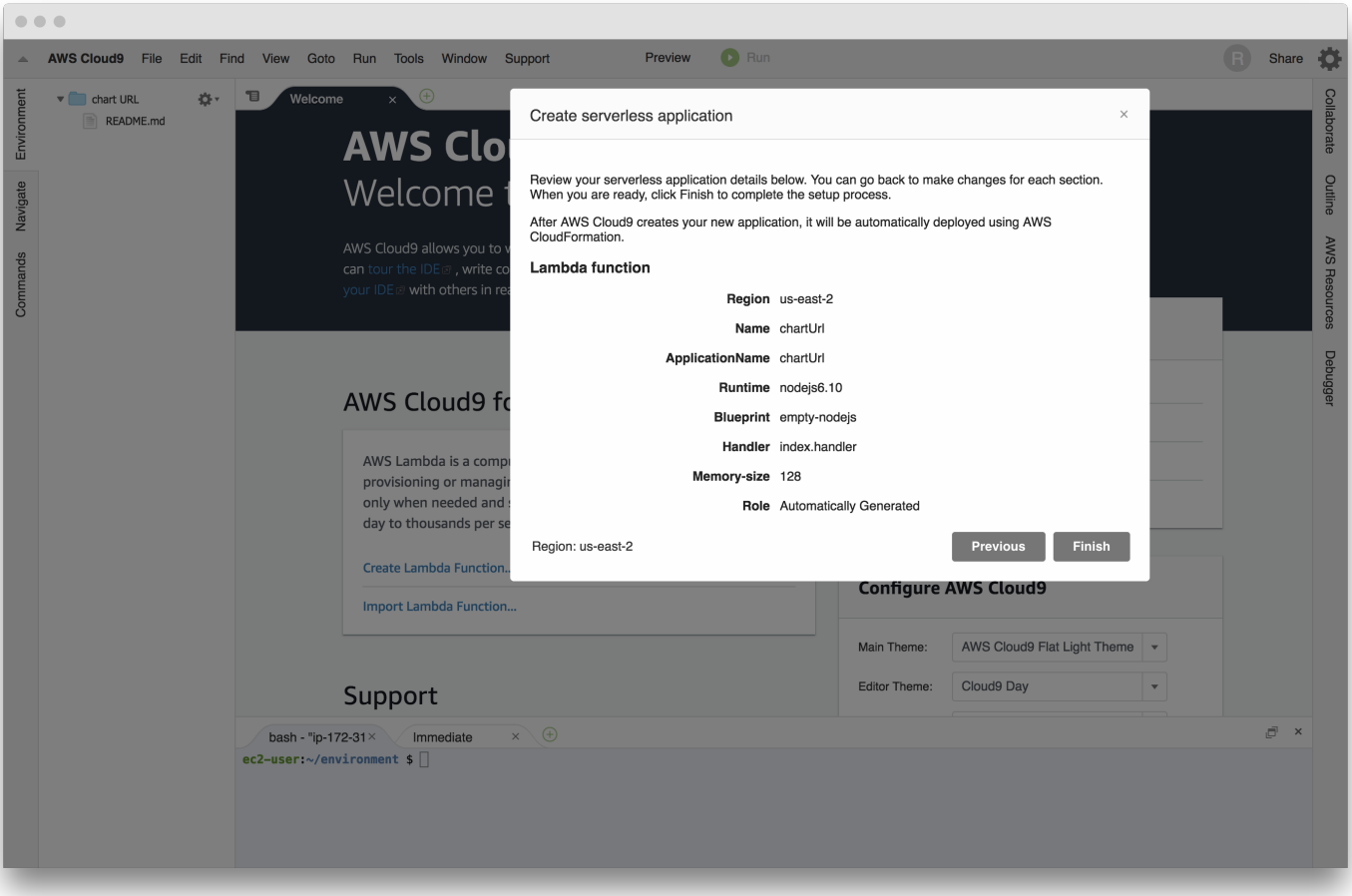
- Create an API Endpoint that invokes that lambda function and outputs that chart

Create a Node.js Lambda Function

The first thing we need to do is create lambda function that will render our chart. In the AWS console, navigate to the Cloud9 service and create a new environment. Cloud9 is an online IDE that lets you write code on a server as well

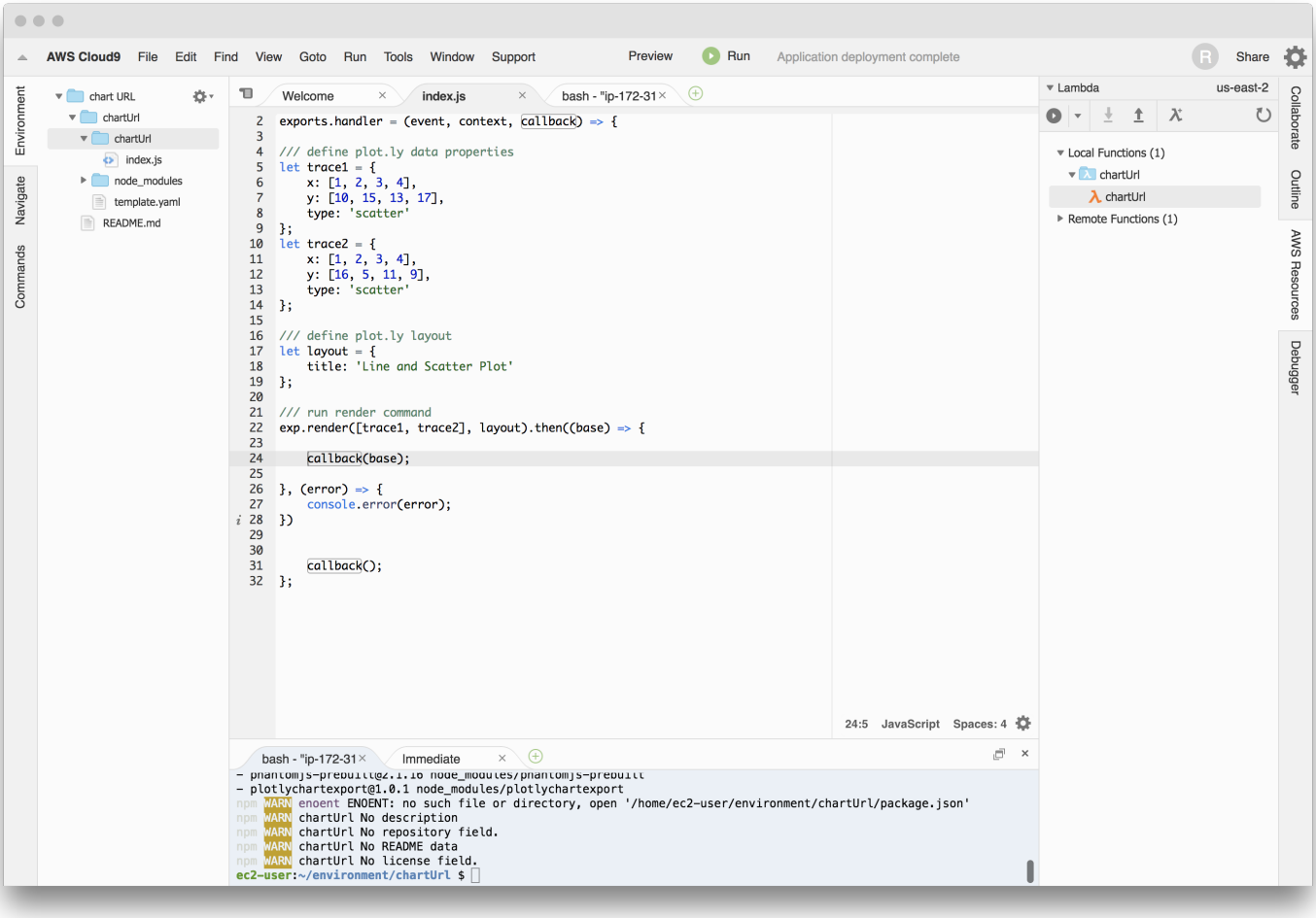
as create lambda functions, test them, and deploy them.





Inside the environment, open a shell terminal and install `plotlychartexport`. https://github.com/mbejda/plotlychartexport (https://github.com/mbejda/plotlychartexport)

```
npm install plotlychartexport --save
```



Add the following code to the lambda function. The code defines 2 scatter plots and sends them along with layout parameters to PlotlyChartExport (https://github.com/mbejda/plotlychartexport) for rendering.

```
const exp = require('plotlychartexport');
exports.handler = (event, context, callback) => {

/// define plot.ly data properties
let trace1 = {
    x: [1, 2, 3, 4],
    y: [10, 15, 13, 17],
    type: 'scatter'
};
let trace2 = {
    x: [1, 2, 3, 4],
    y: [16, 5, 11, 9],
    type: 'scatter'
};

/// define plot.ly layout
let layout = {
    title: 'Line and Scatter Plot'
};

/// run render command
exp.render([trace1, trace2], layout).then((base) => {

    callback(null,base);

}, (error) => {
    console.error(error);
    callback(error);
})

};
```

Awesome, we have our Lambda function prepared to render charts. Deploy it and let's get our API endpoint working.

AWS API Gateway

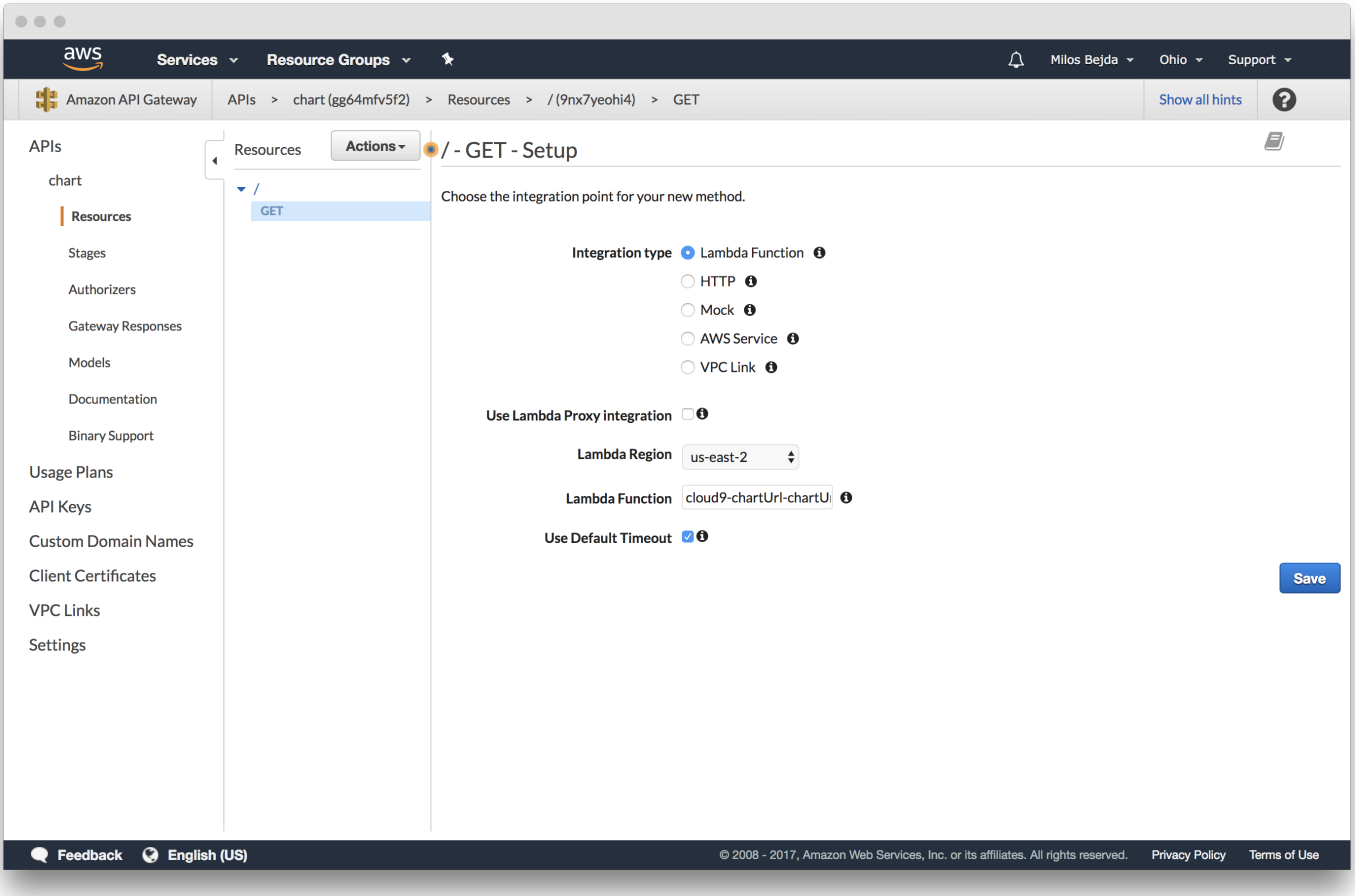
To get the API endpoint up and communicating with our lambda function we need to :

- Create a new API endpoint

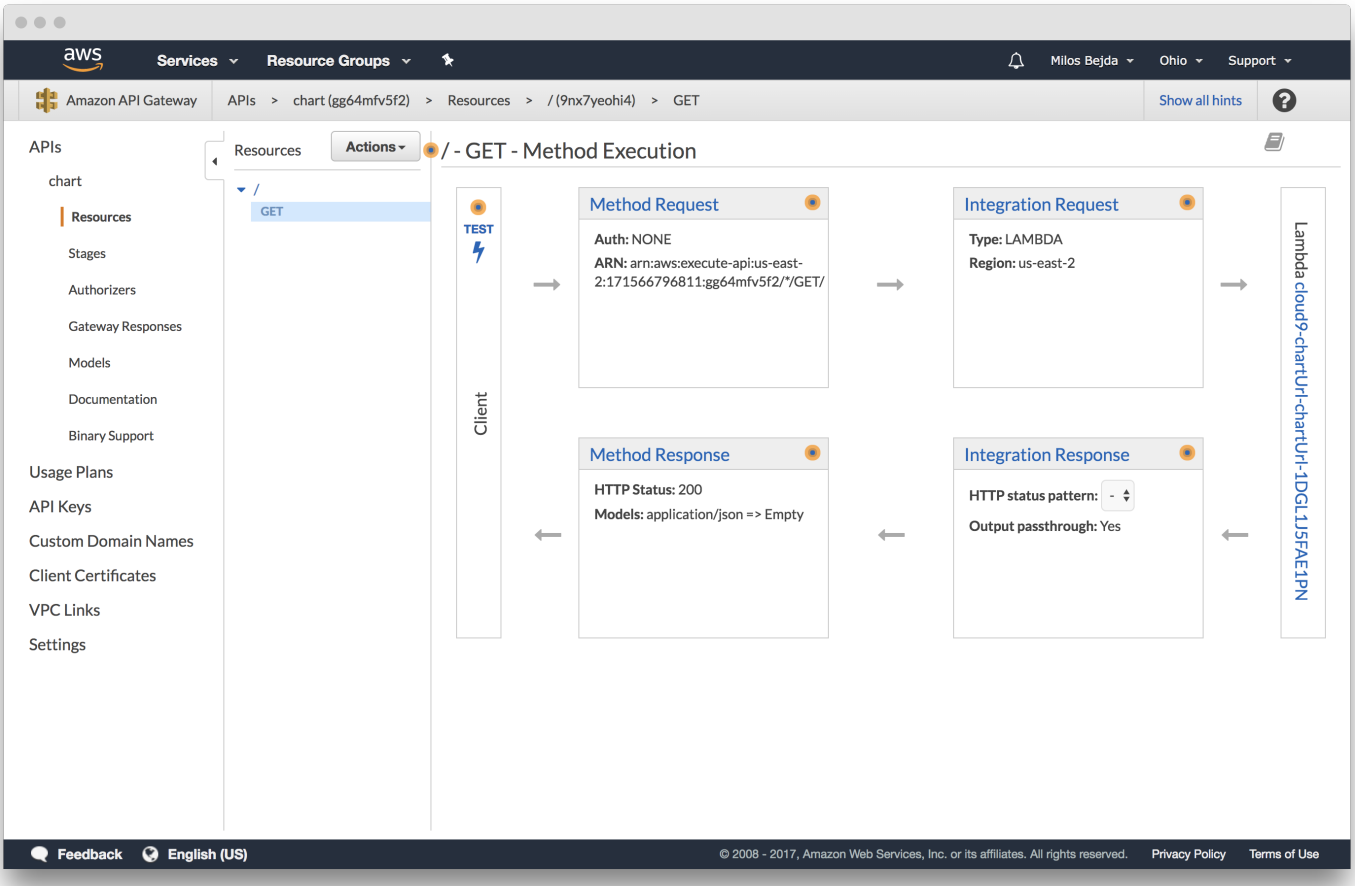
- Connect the Lambda
- Define API Content Type
- Add Binary Support for `image/png`.

Lets Get Started

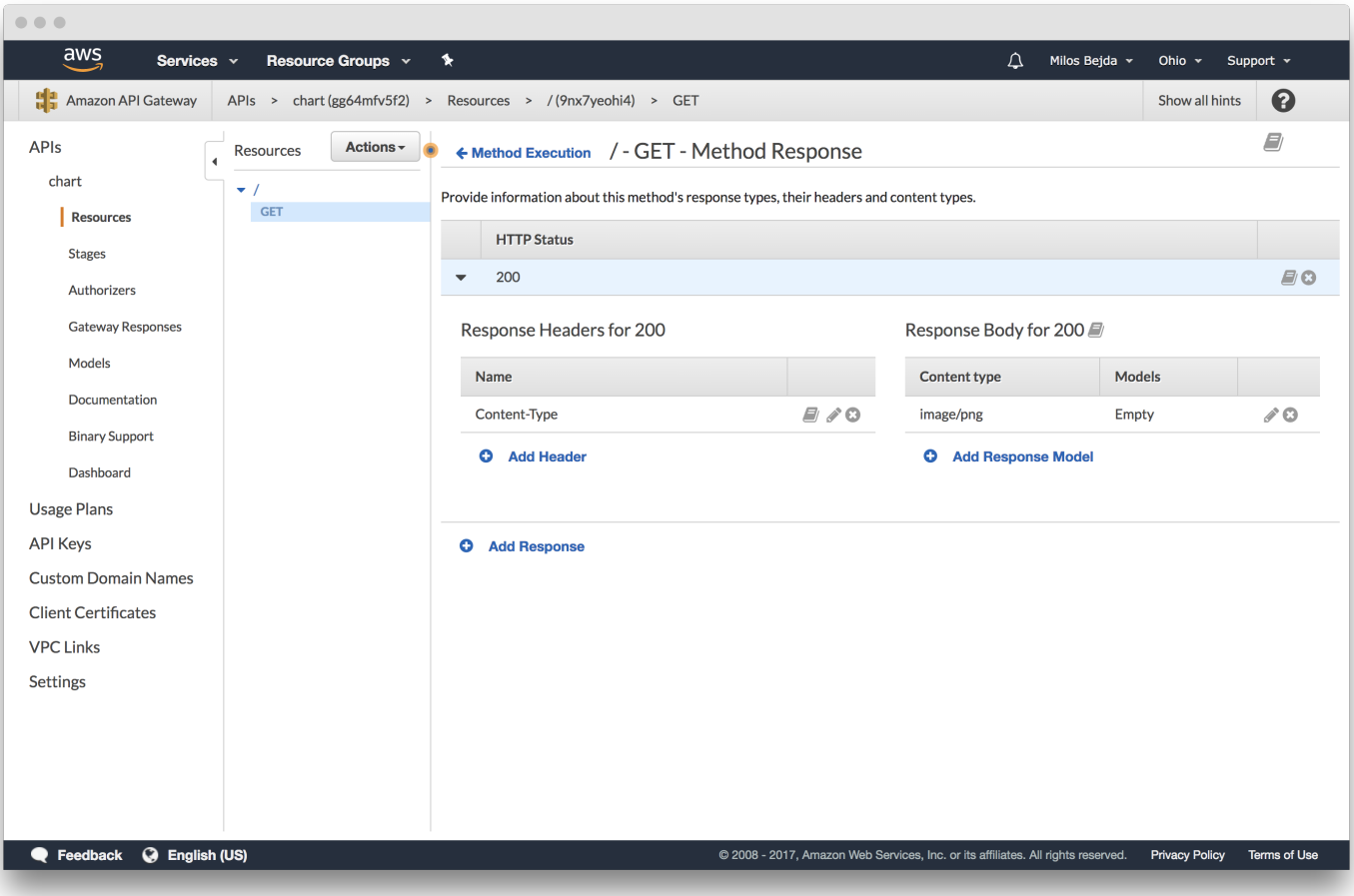
Navigate to the AWS API Gateway service. Create an API Gateway endpoint by clicking on the *actions* dropdown and selecting *GET* method. Then in the setup section, for the integration type select *Lambda Function* and set our lambda function.



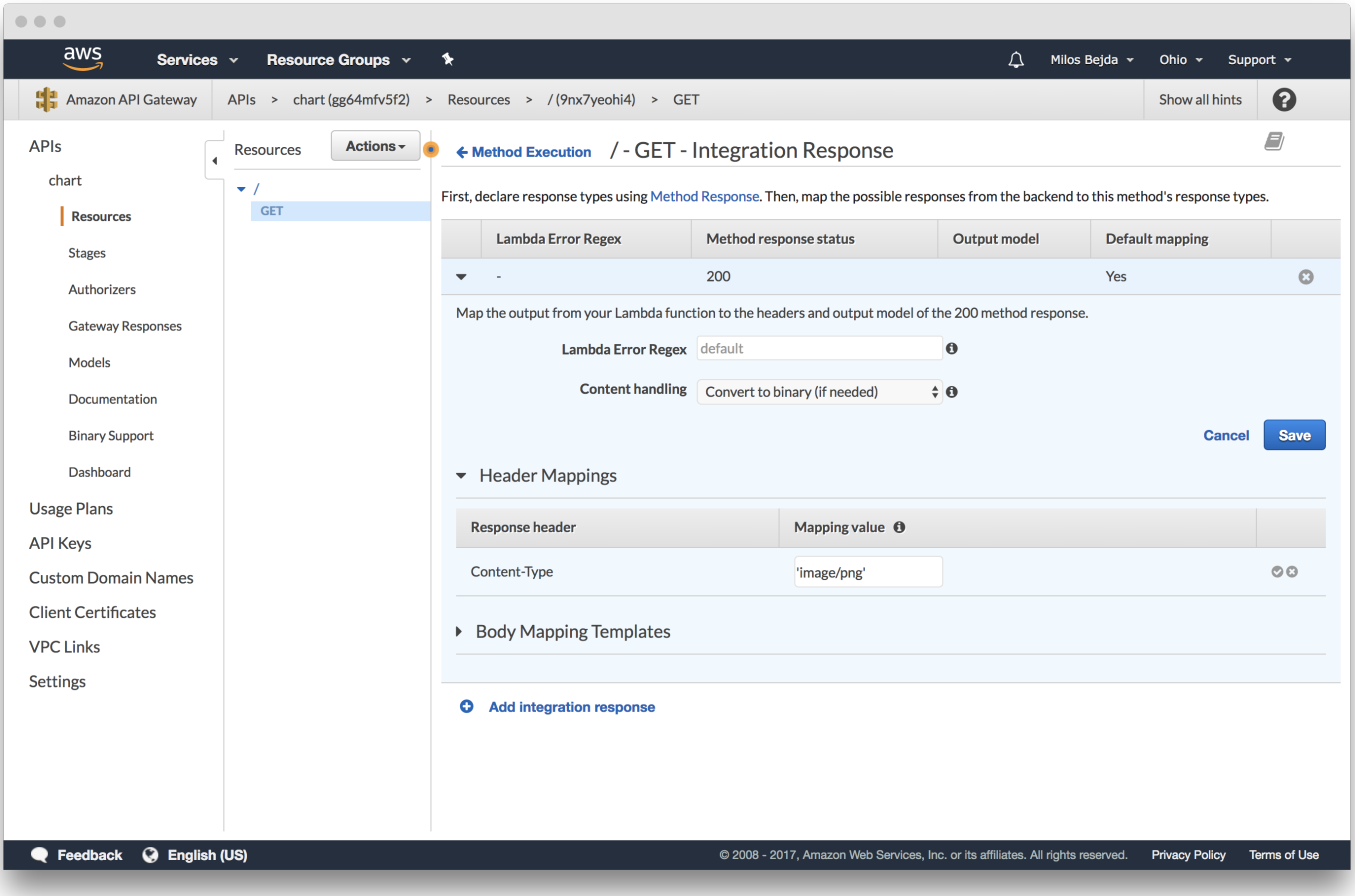
Navigate to method response page



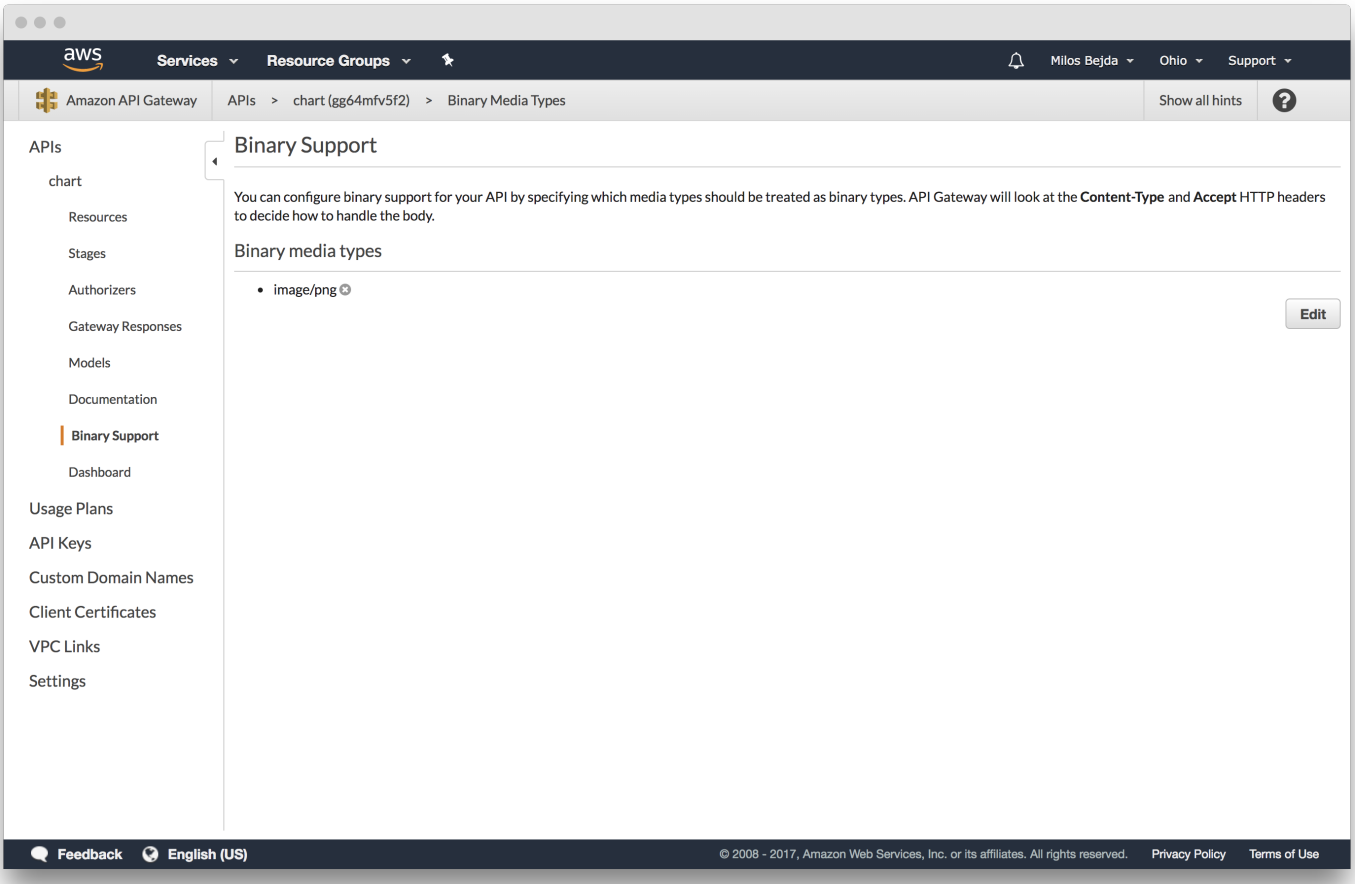
Add header `Content-Type` and response model `image/png`.



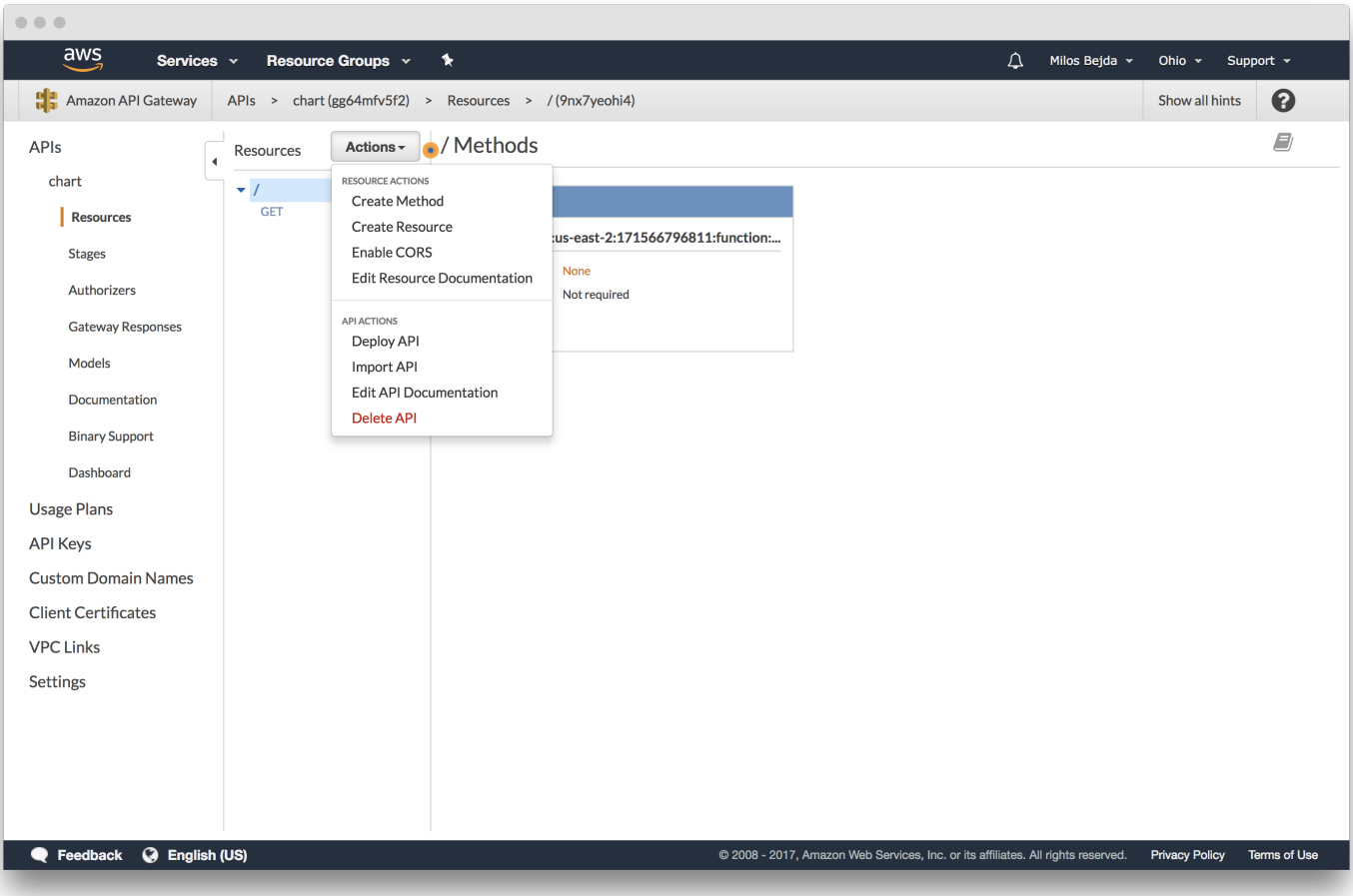
Navigate to integration response page and edit the header mappings to contain `image/png`. Then in the content handling drop down, select `Convert to binary (if needed)`.



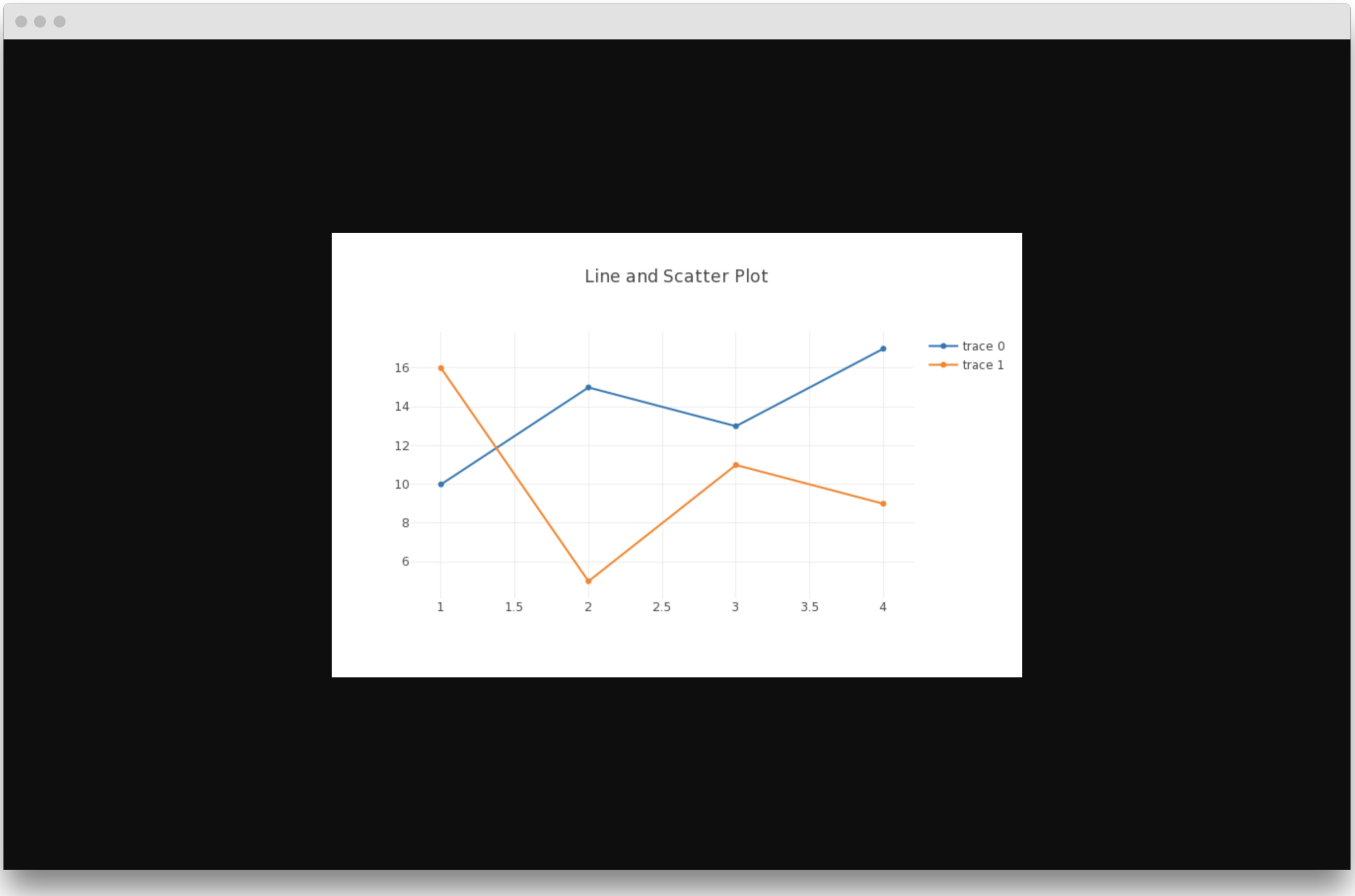
Navigate to binary support and add `image/png`



That should be it! Time to deploy the API and try it out.



Navigate to the API endpoint to invoke the lambda function. The function renders the plot.ly chart and returns it in base64 encoded format. The API gateway then converts that base64 encoding to binary and outputs it to the page.



Conclusion

NPM Dependency: https://www.npmjs.com/package/plotlychartexport (https://www.npmjs.com/package/plotlychartexport)
Github : https://github.com/mbejda/plotlychartexport (https://github.com/mbejda/plotlychartexport)

If you have any feedback on this tutorial, you are stuck or I'm missing something, send me a tweet and let me know. @notmilobejda (https://twitter.com/notMiloBejda).

Happy Hacking