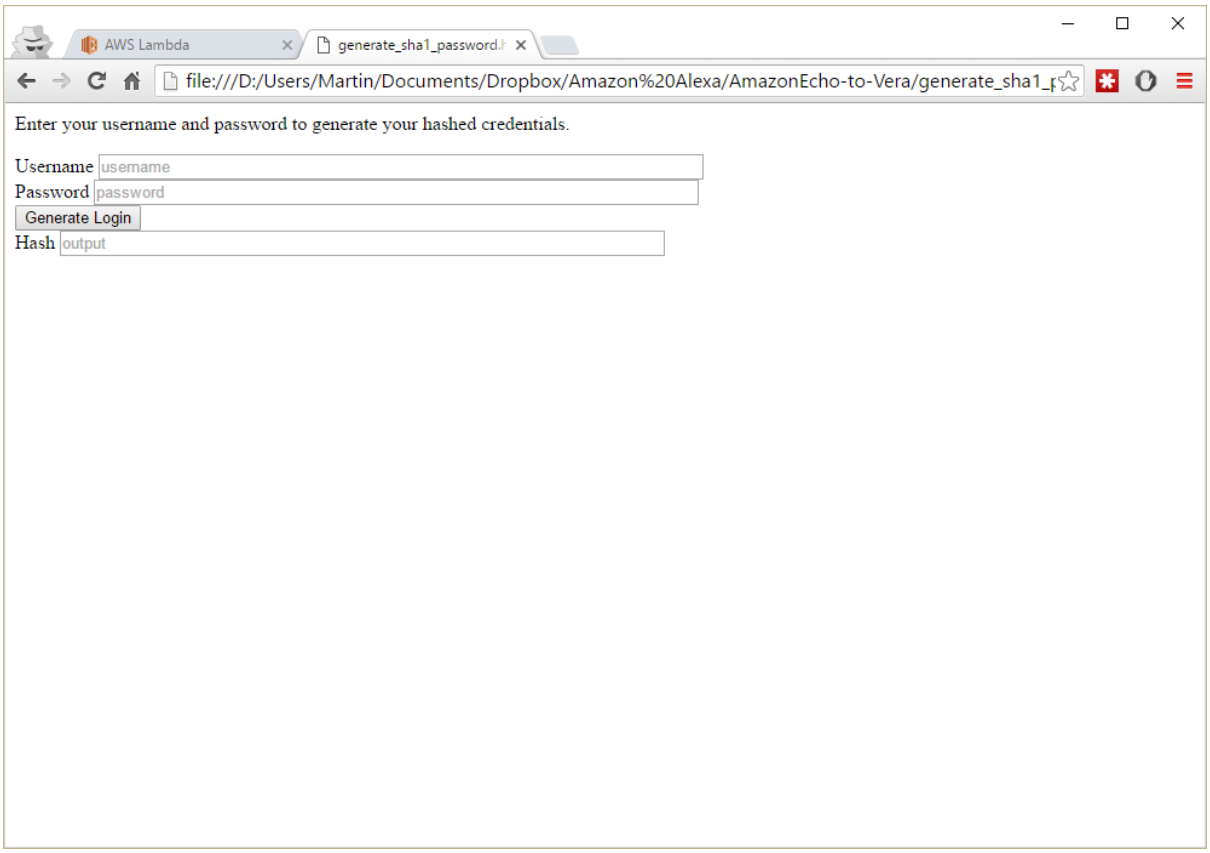
## Vera Credentials

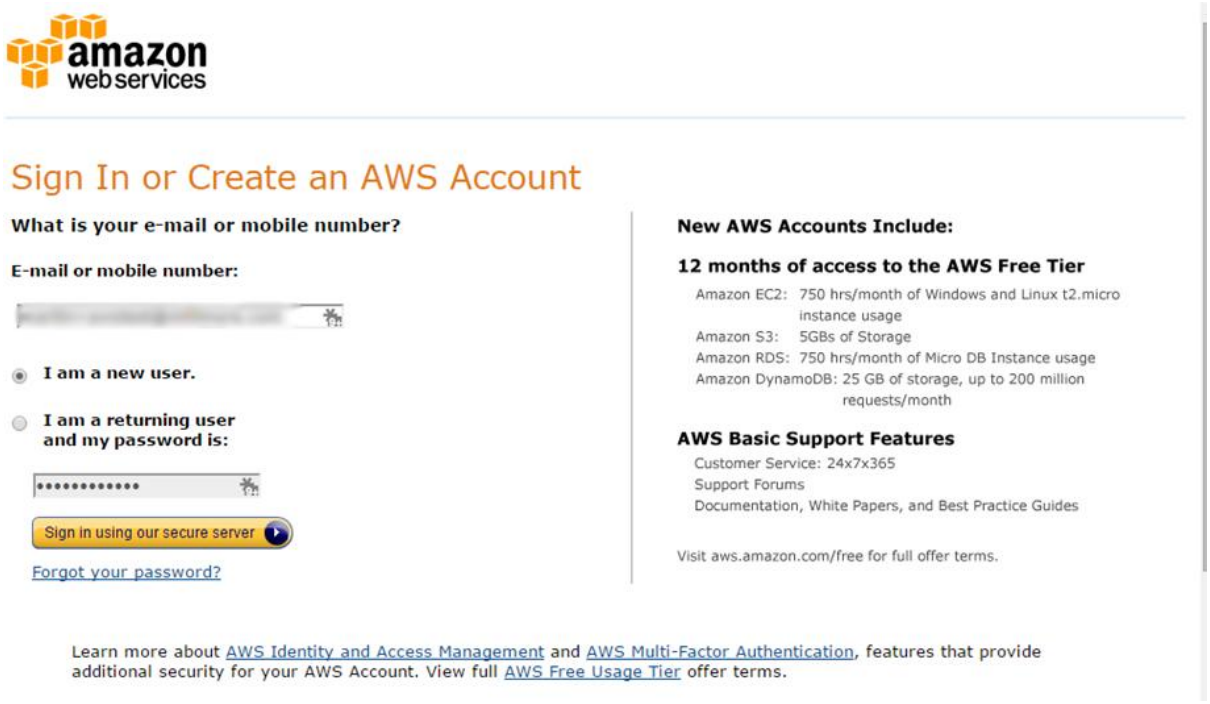
First of all you need to get your vera remote access credentials. Download https://raw.githubusercontent.com/mmillmor/AmazonEcho-to-Vera/master/generate_sha1_password.html to your computer and open it with a web browser

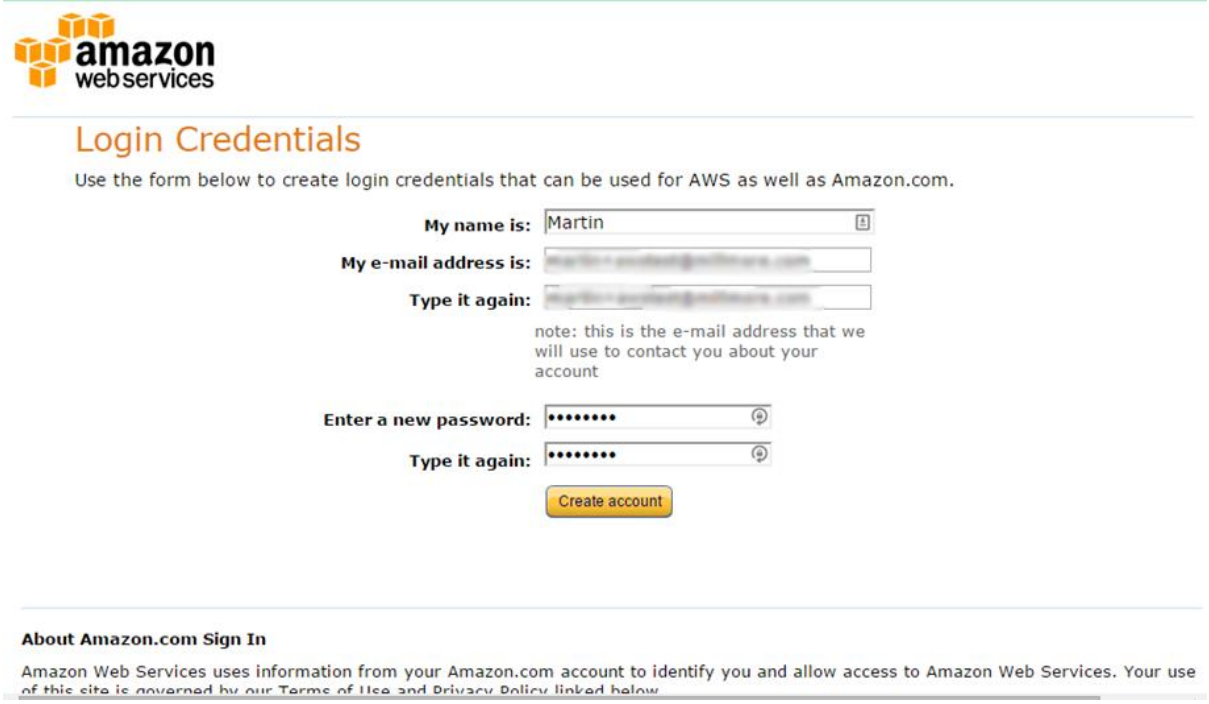



Enter your username and password, and press Generate Login. That will calculate your hashed credentials, which you will need later on. These are the same credentials that you would use at home.getvera.com

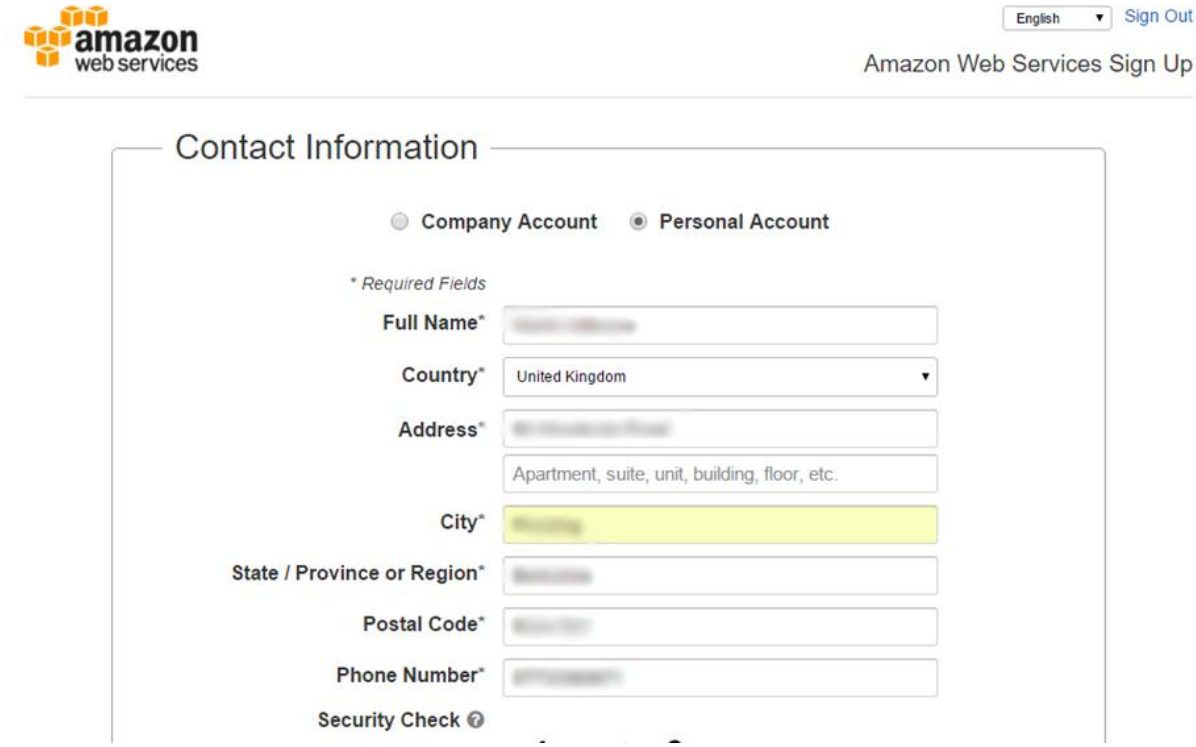
## Set up the remote code

1) Sign up for a free AWS account at https://portal.aws.amazon.com/gp/aws/developer/registration/index.html?nc2=h_ct

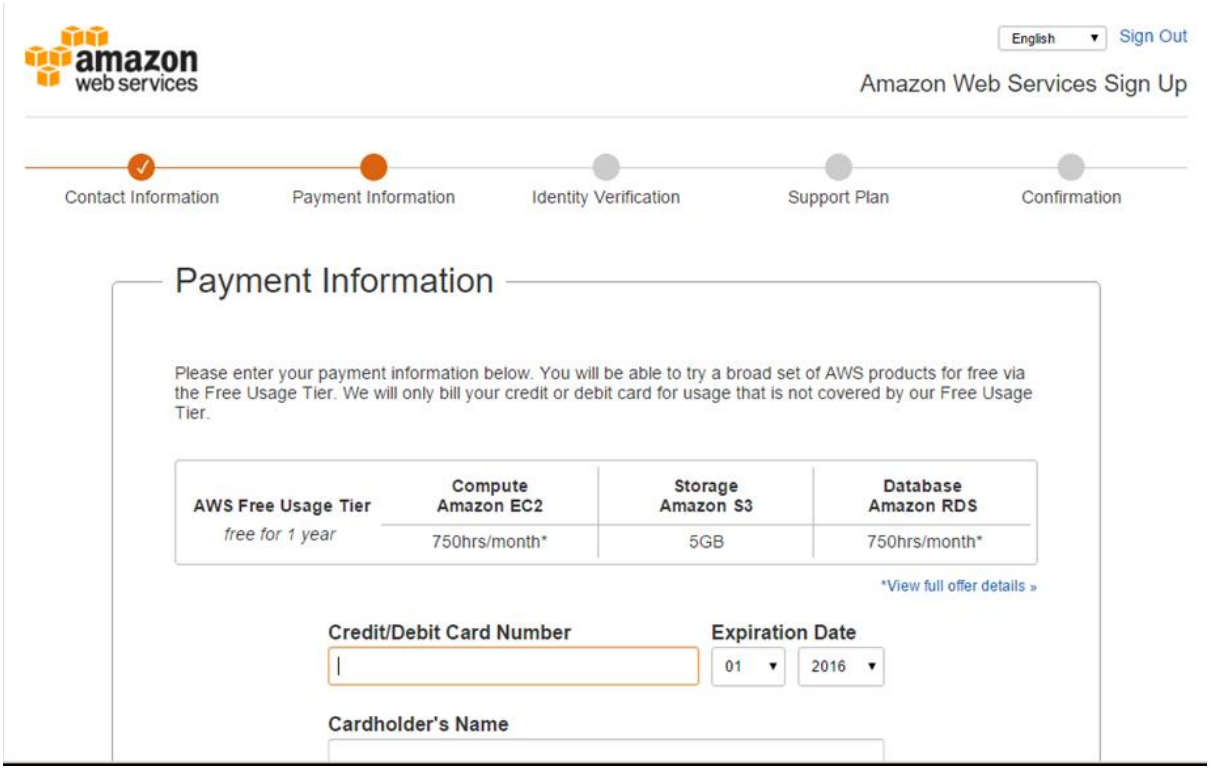


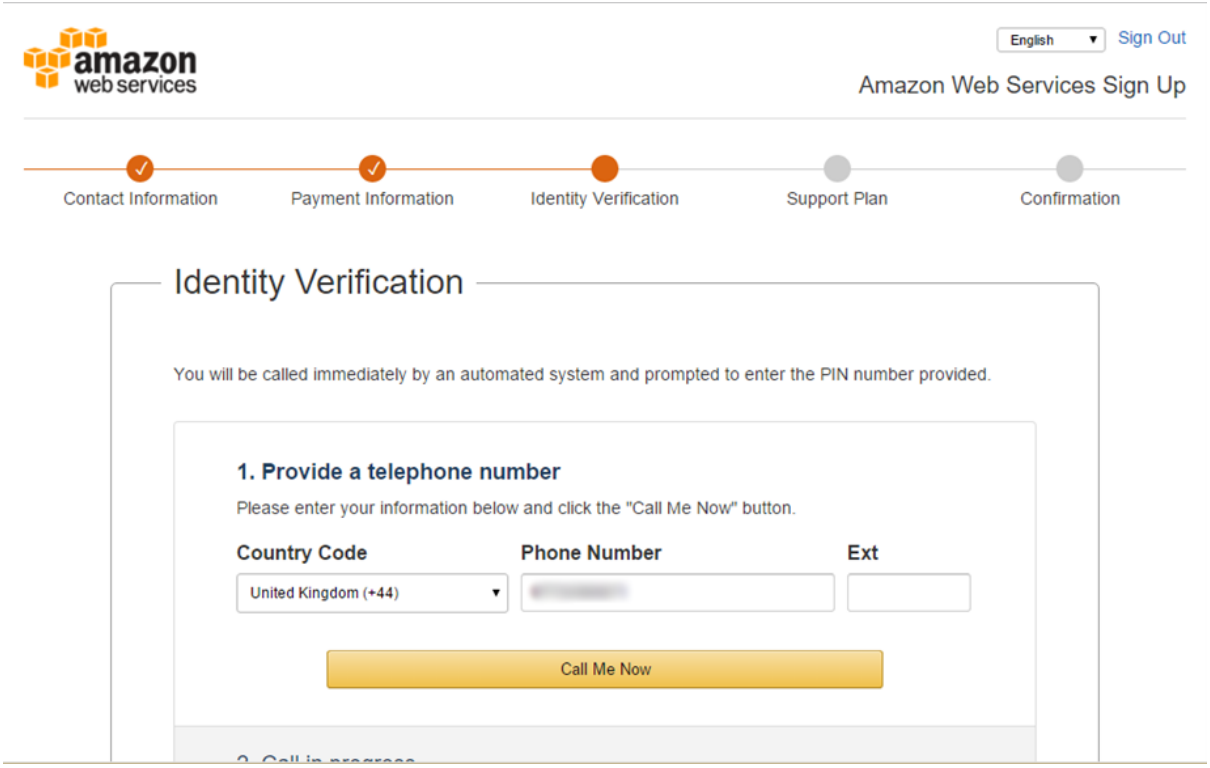

Enter your name, email and password



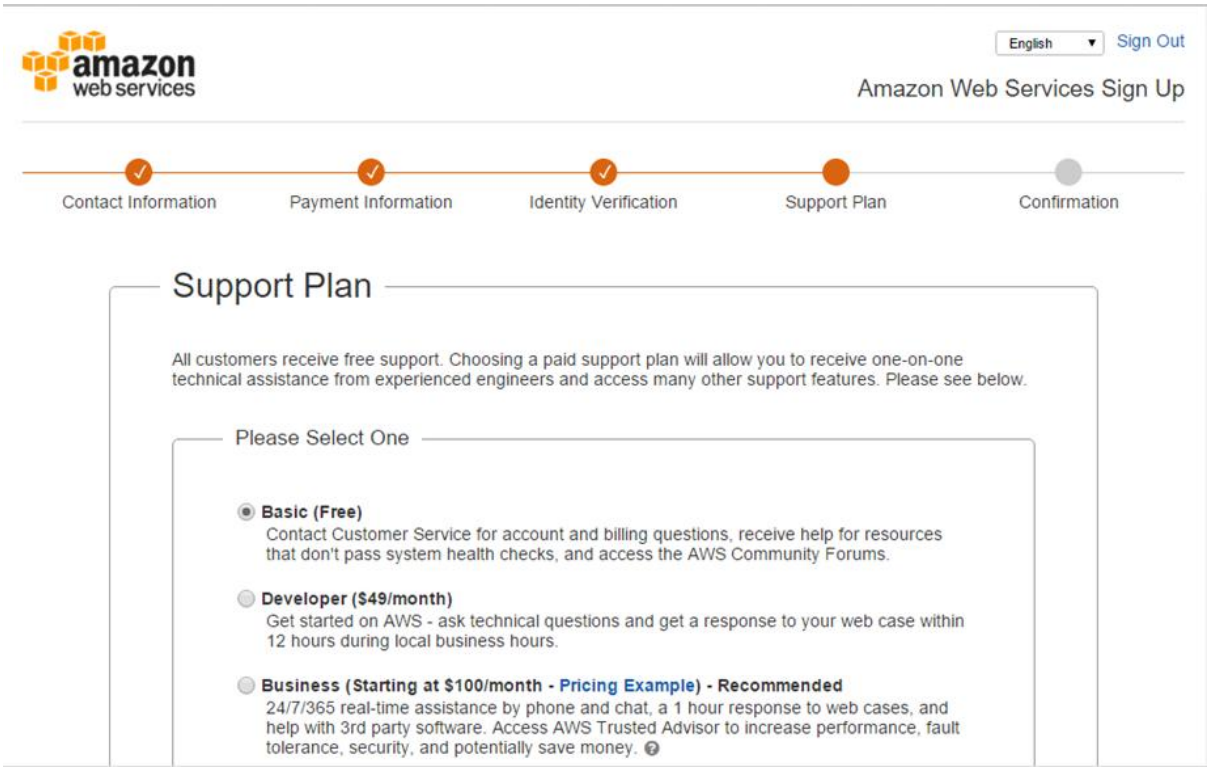

Enter contact details



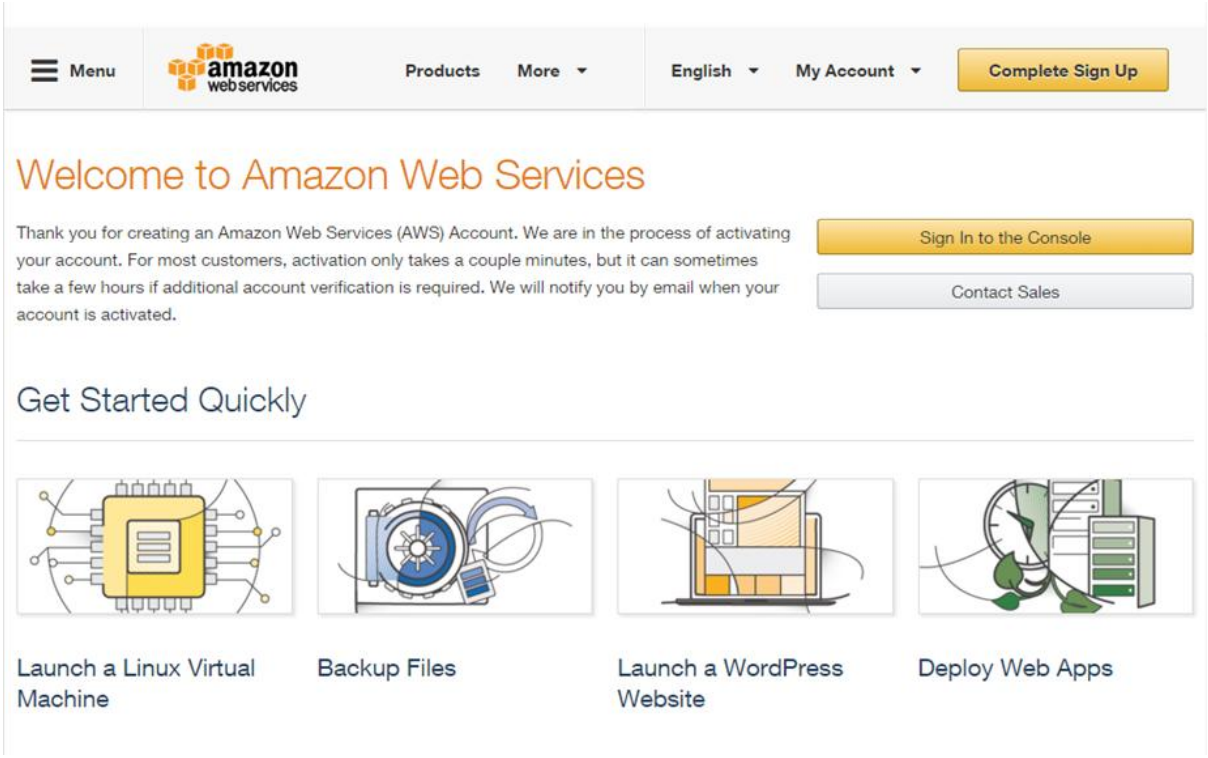

Enter a credit card number. Note, you will never be billed for the usage describe here - the Lambda service is free under 1 million hits a month (https://aws.amazon.com/lambda/pricing/)



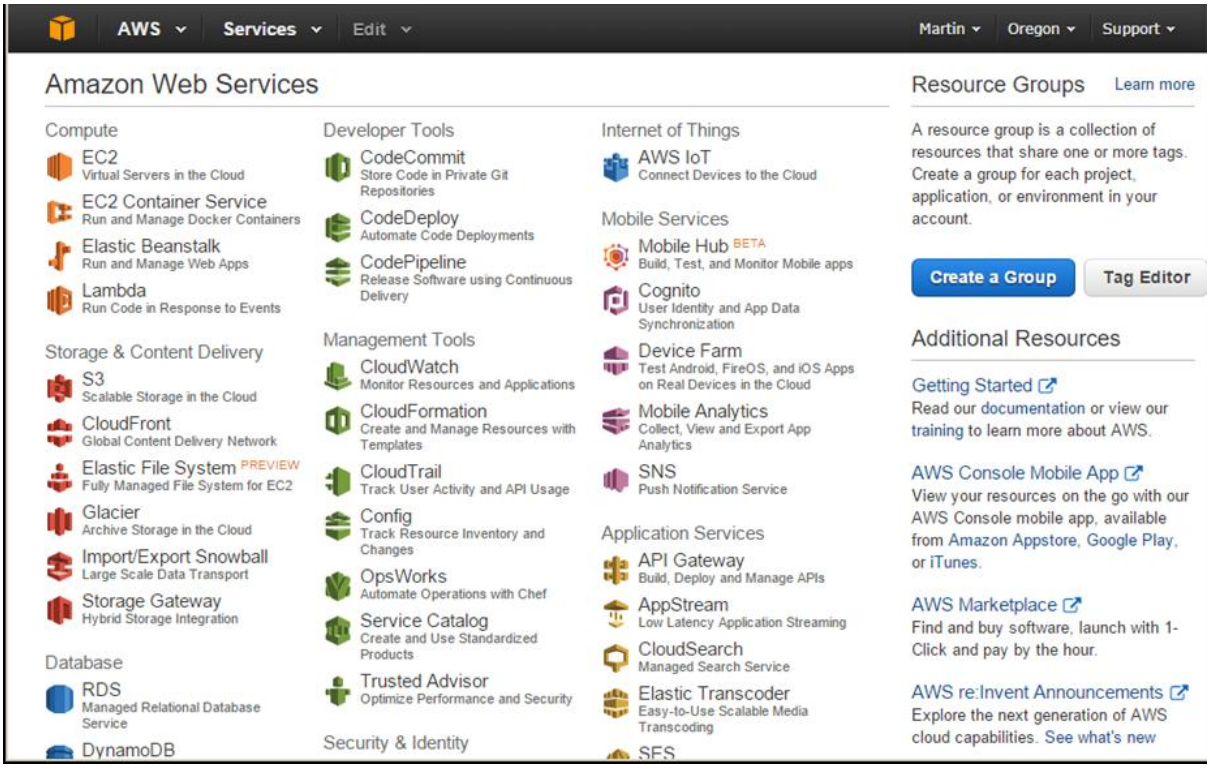

Amazon will verify your identity




Pick the free support plan

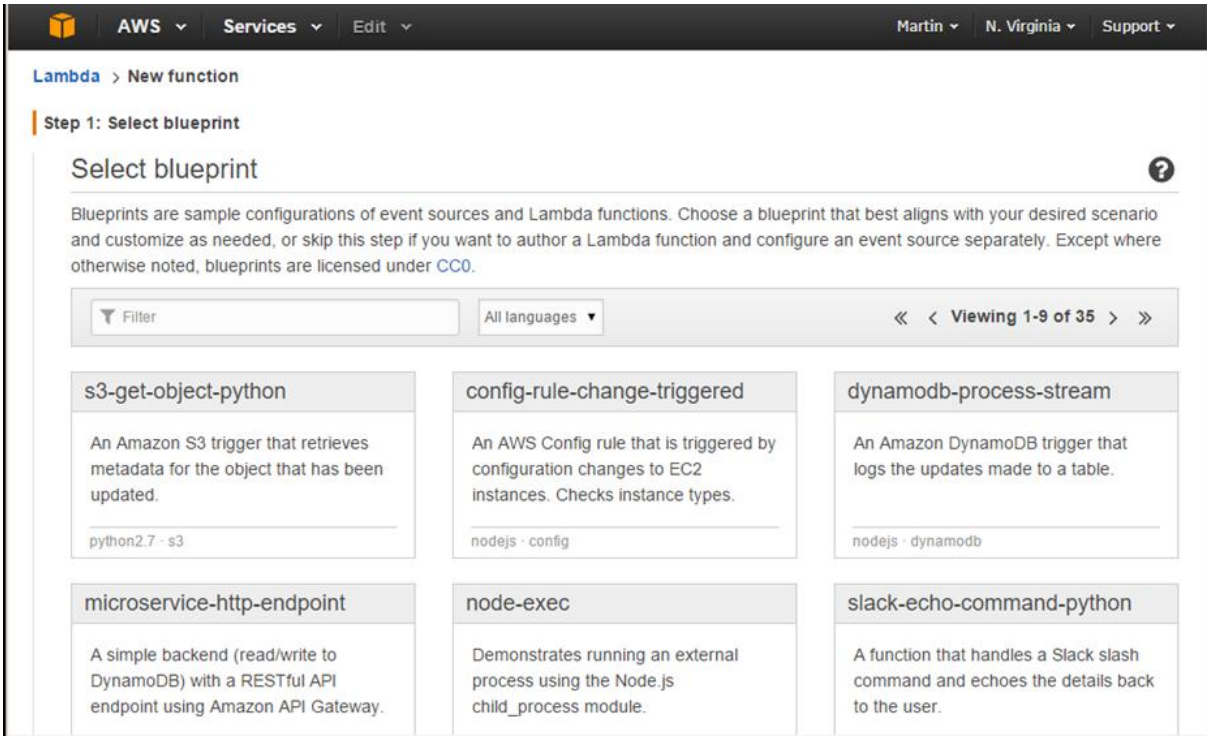



And that's your account created. Sign in with the account you just created

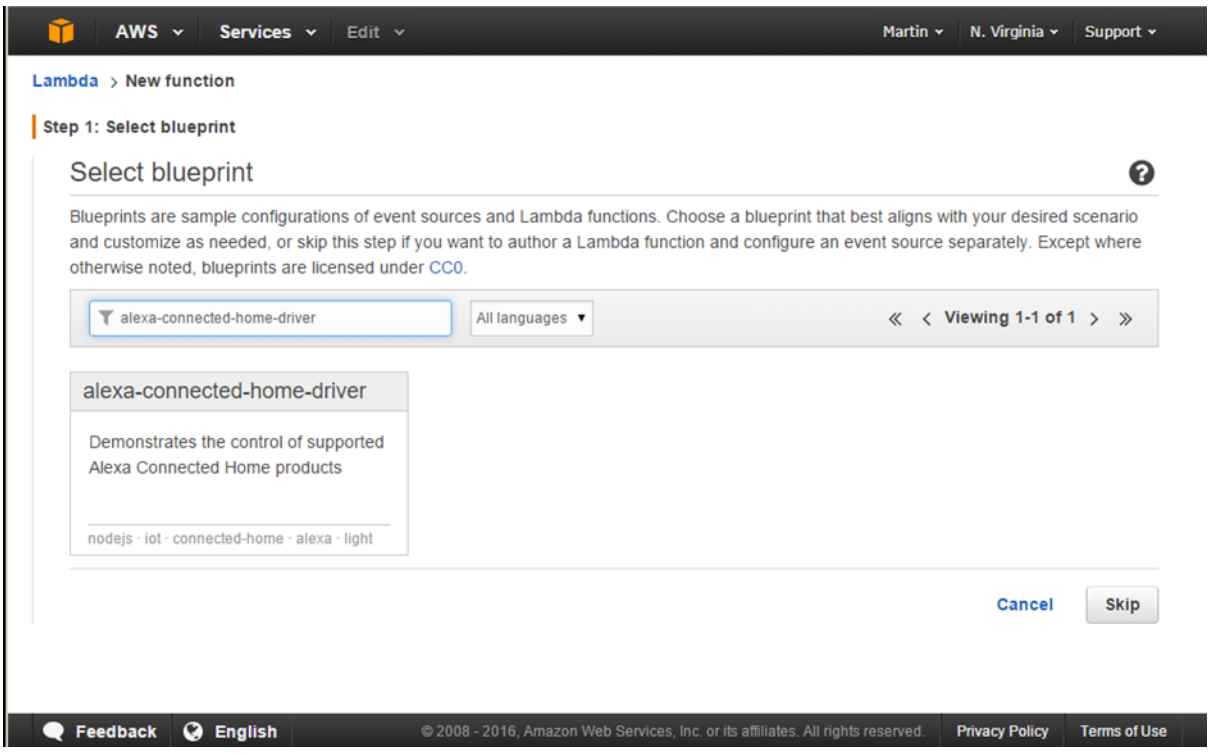


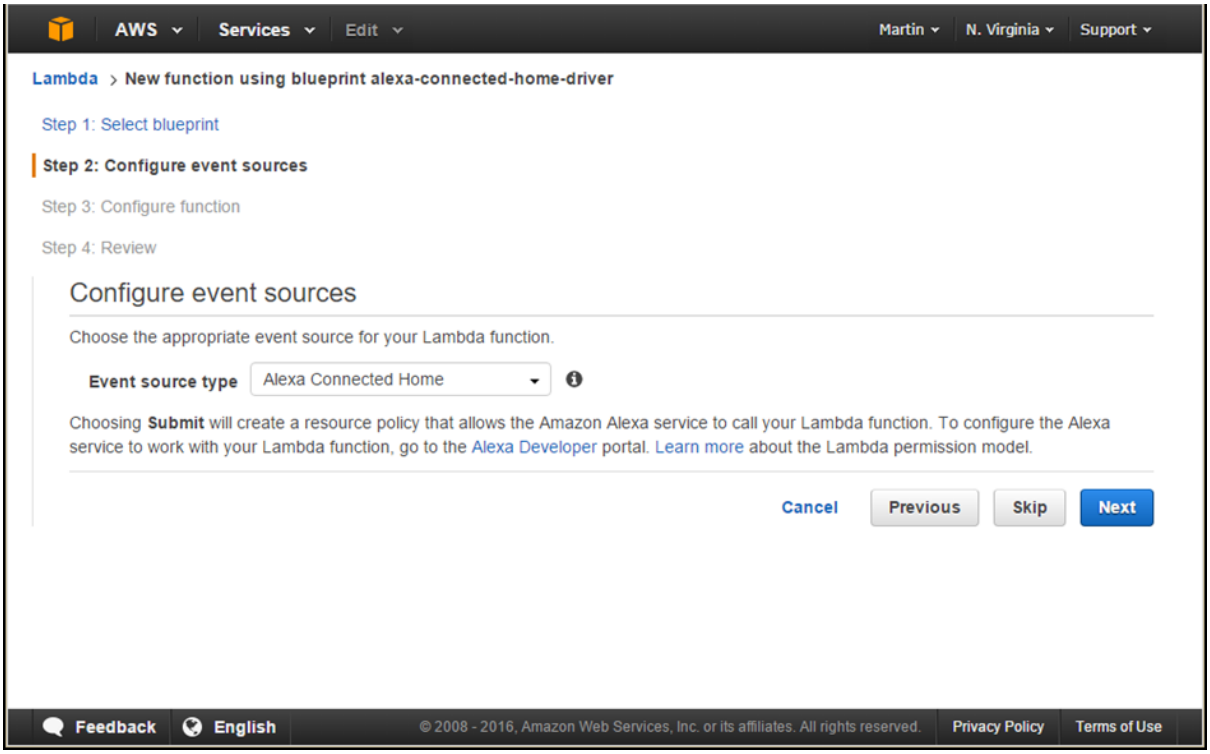

You will be taken to the AWS dashboard. Click on Lambda


Region must be us-east-1. Click on the region link between your name and "Support" in the top right to change region to us-east-1 (N. Virginia) then press Get Started Now

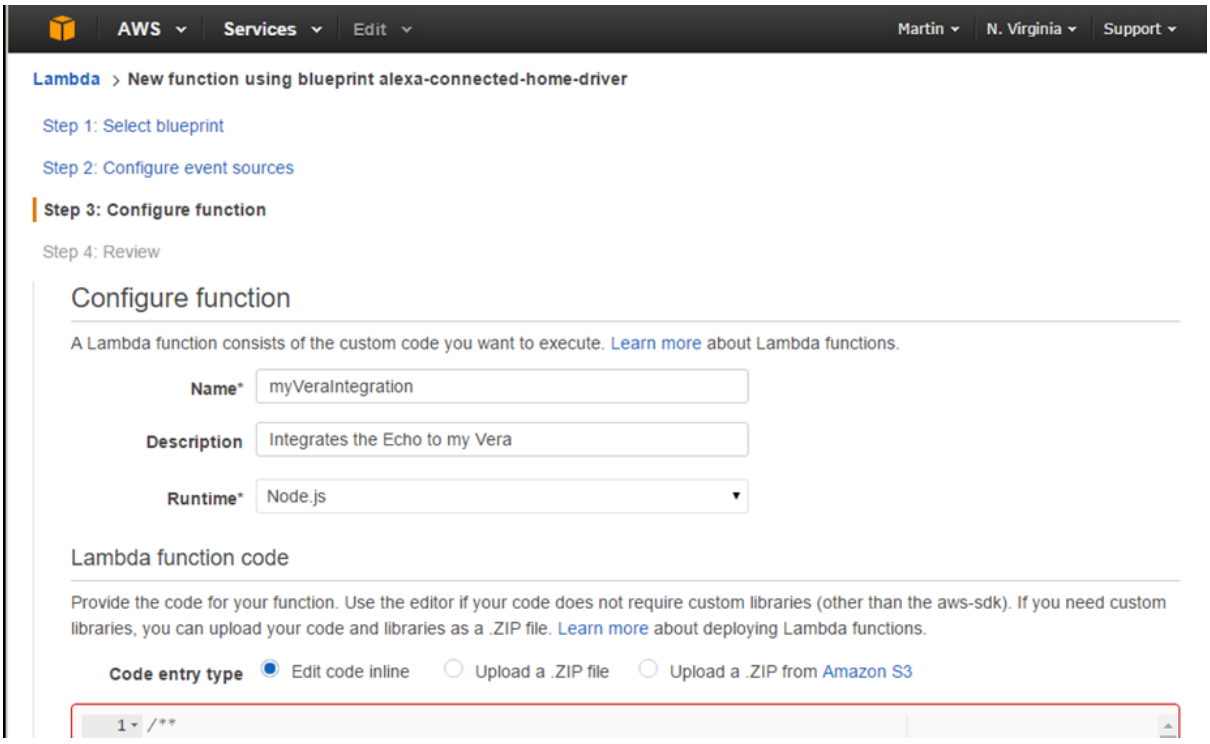



Enter "alexa-connected-home-driver" in the filter box, and click on that box





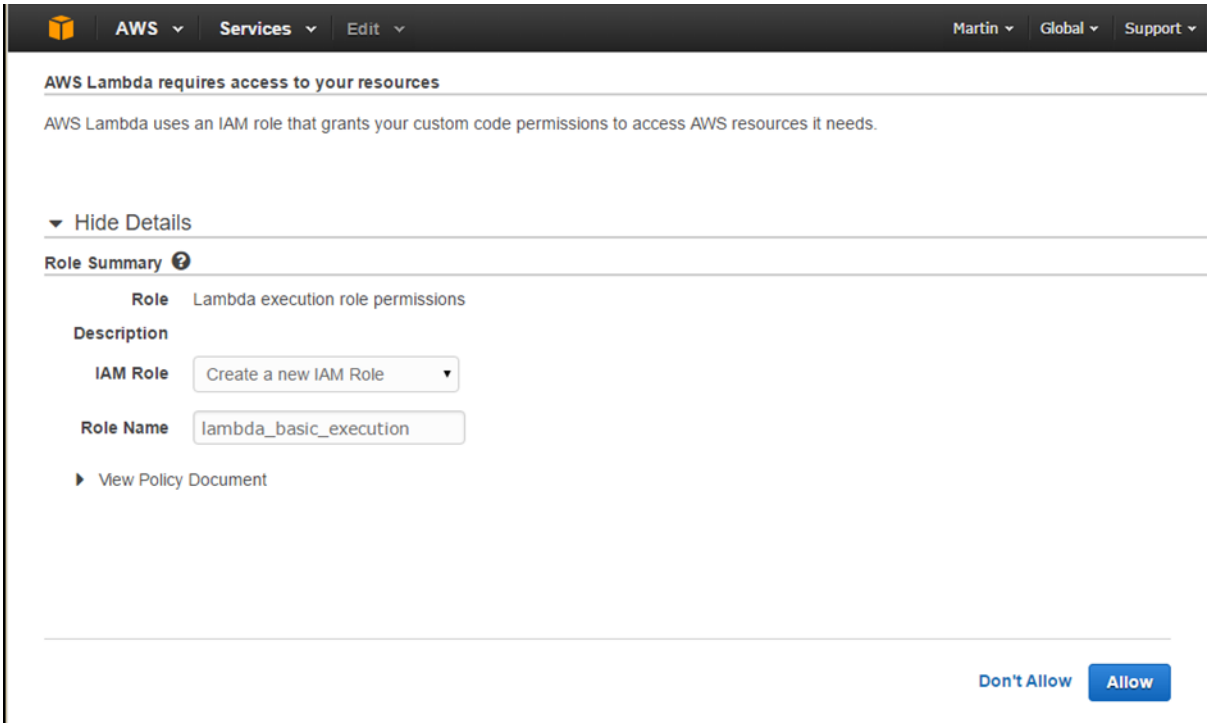

Leave the event source as Alexa Connected Home, and press Next



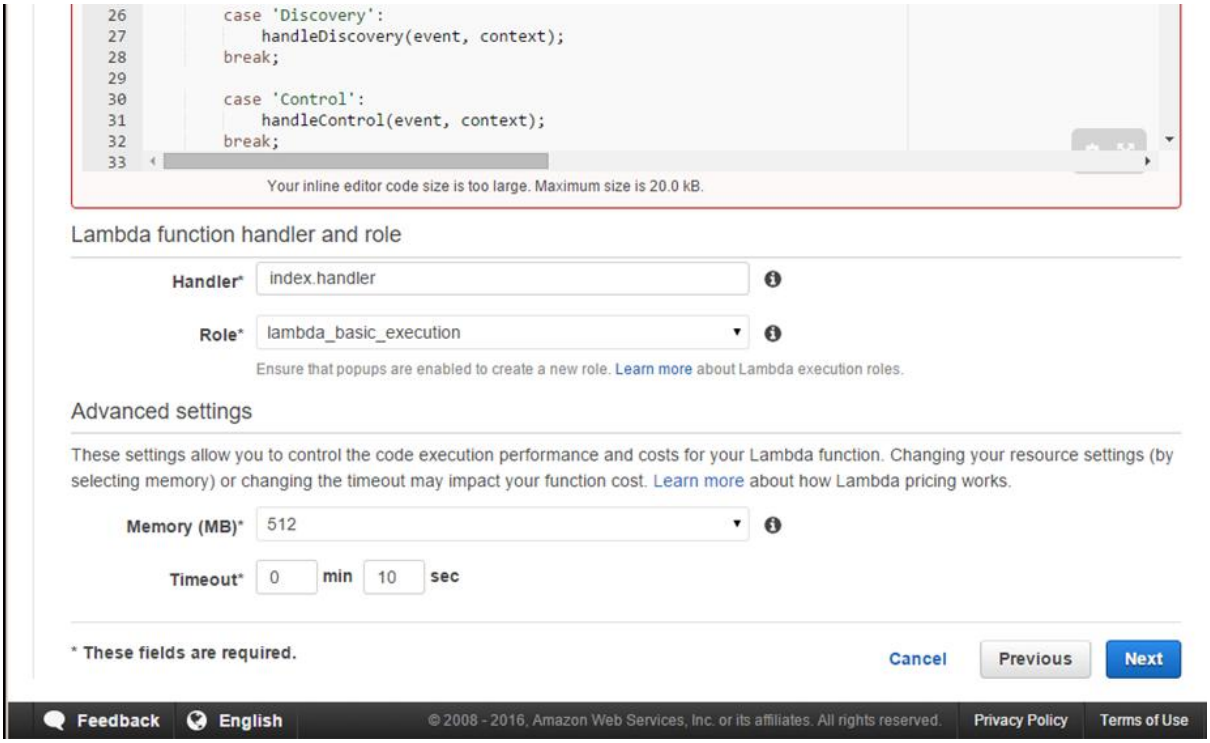

Enter any name and description for your code. Leave the runtime as Node.js. Scroll down to the code entry. Copy and paste the entire contents of https://raw.githubusercontent.com/mmillmor/AmazonEcho-to-Vera/master/lambda/alexa_lambda_amazon_oauth.js in there.

Scroll down and replace the text {enter your username} with your vera username, and the text {enter your encoded password} with the hashed vera password from the first step.

Scroll down to the bottom of the page. For Role, pick Basic Execution Role* and it will open a new window to create a role;

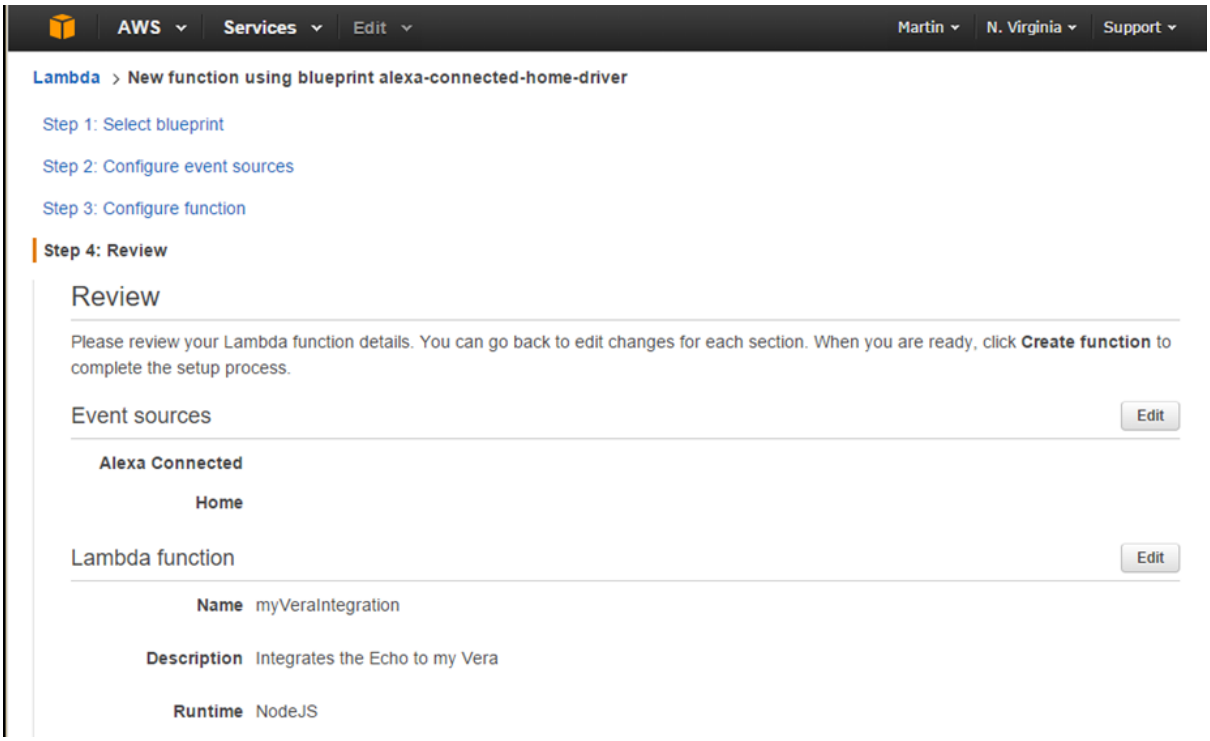



name the role lambda_basic_execution, and click Allow. This will go back to the previous page with the role populated. Leave everything else at the default values.



If it complains about the file being too big, remove the copyright text at the top of the page (everything between /** and */)

Click Next

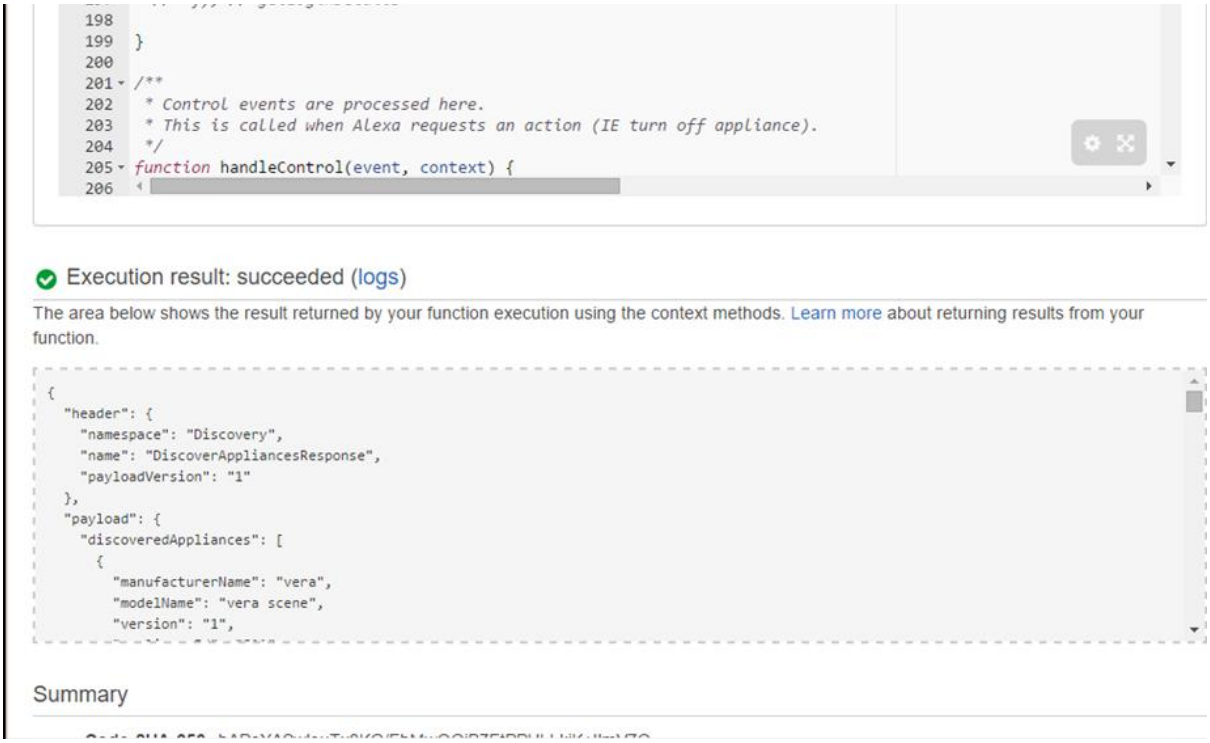



Scroll down and click Create Function.

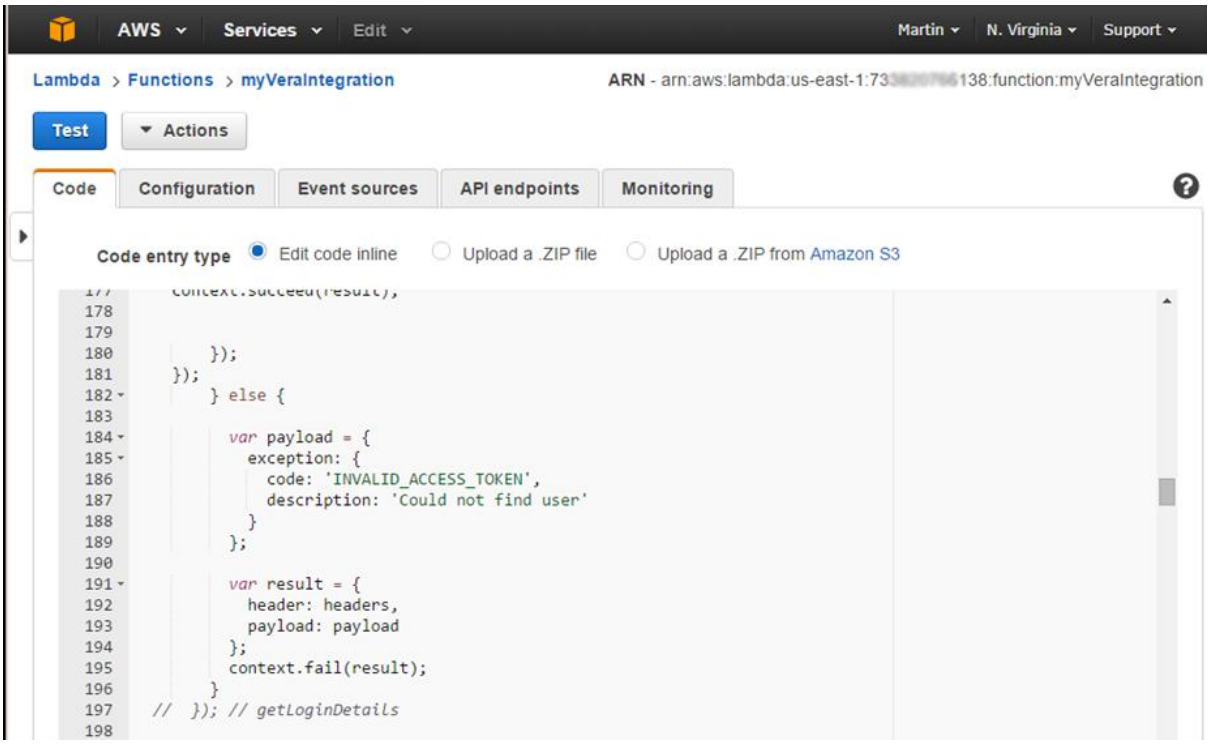
Click the Code tab, then the Test button, and paste the following in to the text box;

```
{
  "header": {
    "namespace": "Discovery",
    "name": "DiscoverAppliancesRequest",
    "payloadVersion": "1"
  },
  "payload": {
    "accessToken": "dummy"
  }
}
```

Scroll down, and you should see a list of your devices;




At the top of the page you will see the text "ARN - " followed by a string like arn:aws:lambda:us-east-1:XXX. You will need that string later.

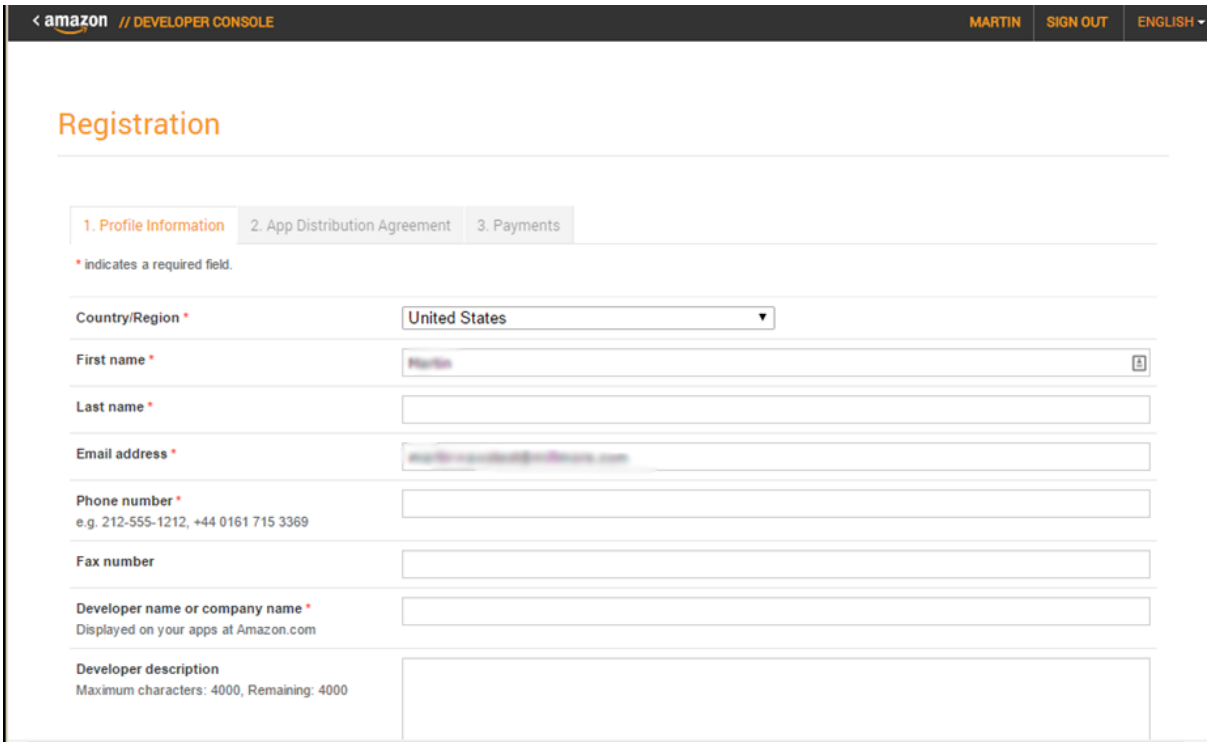



Well done - you are almost done!

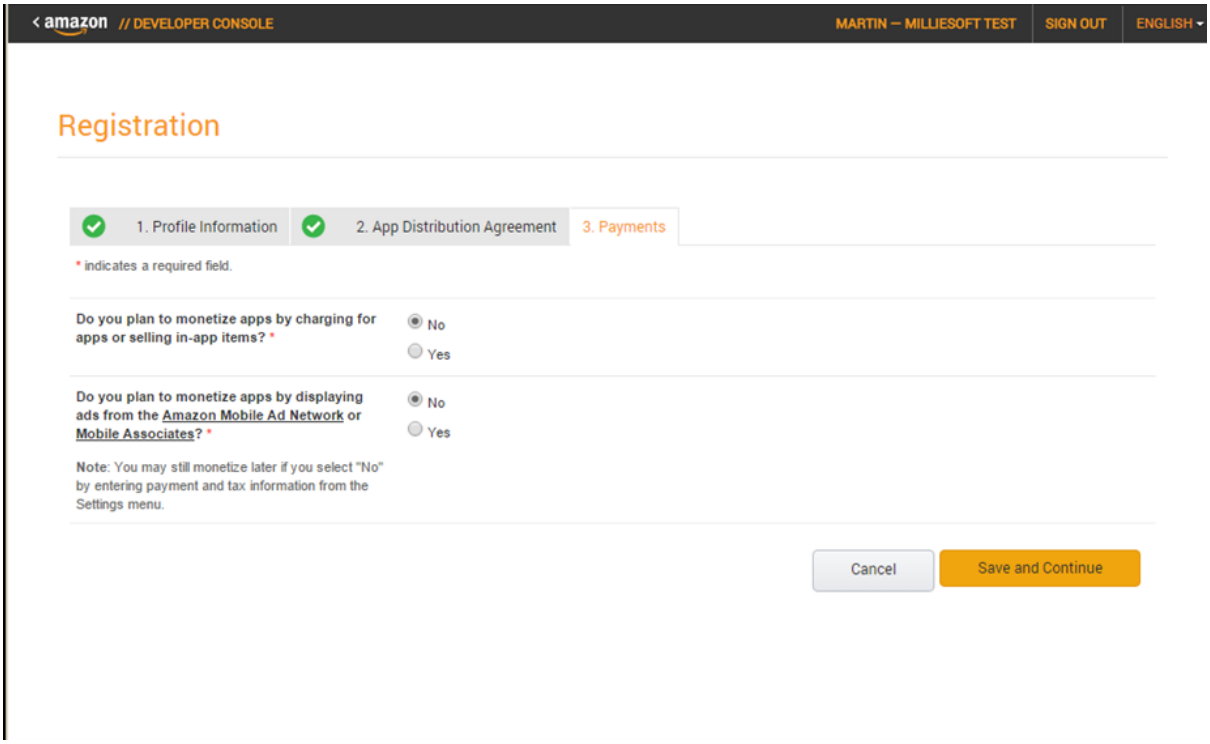
## Setting up OAuth2

Amazon expect everyone to use OAuth2 for login, but the Vera doesn't (which is why we hard coded the username and password). We still need an OAuth2 server though or the Echo won't do anything. Fortunately Amazon provide a simple one we can use.

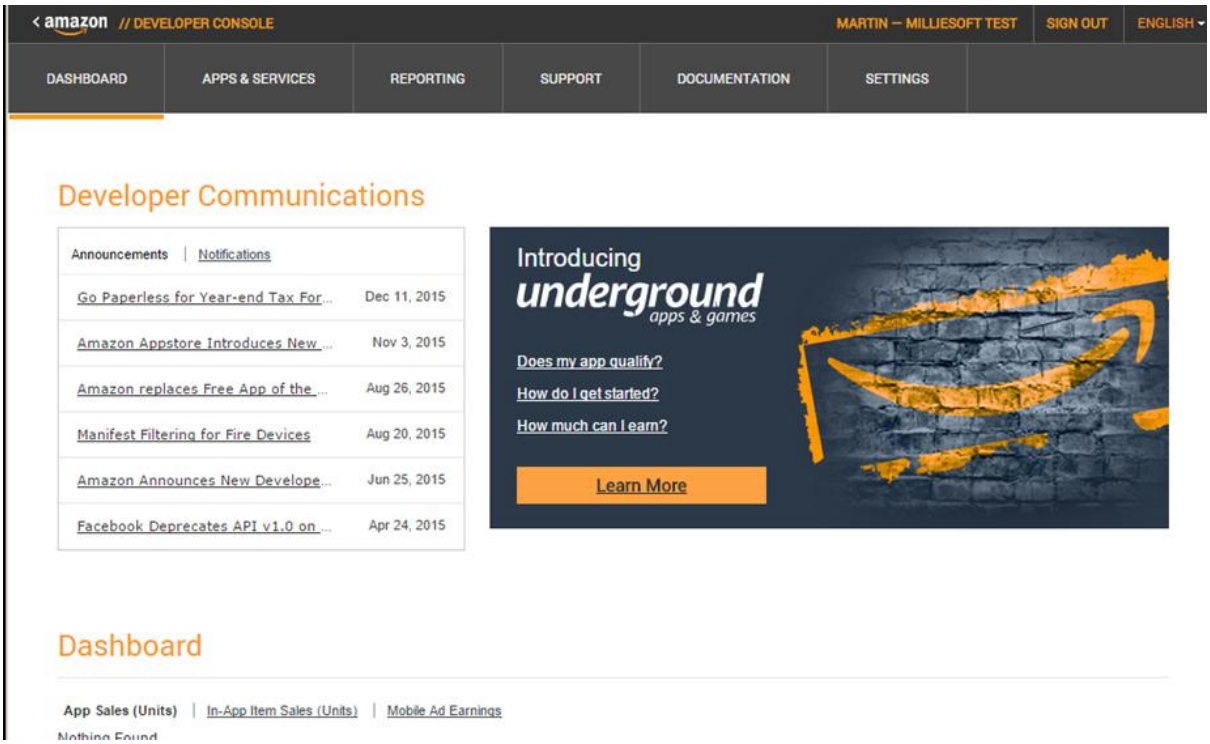
Go to https://developer.amazon.com/lwa/sp/overview.html. You will get a page to register as a developer (free)



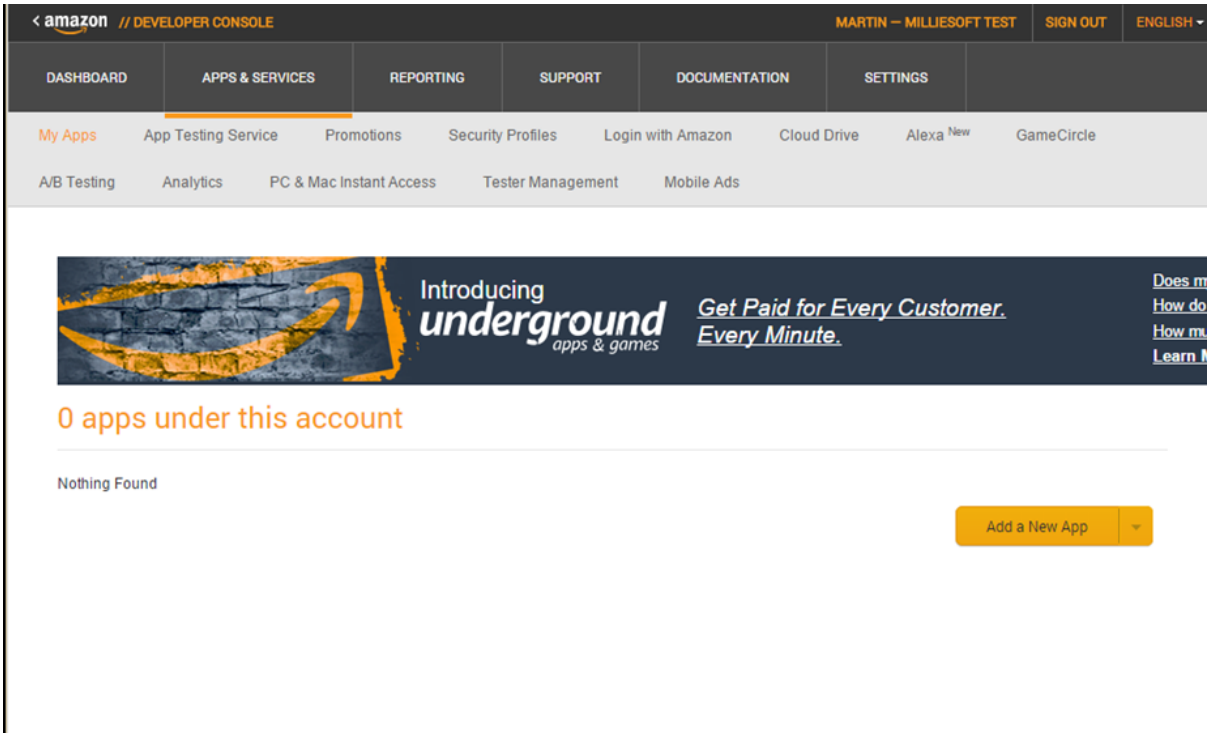

Fill out your details, then press Save and Continue. Accept the license agreement on the next page.



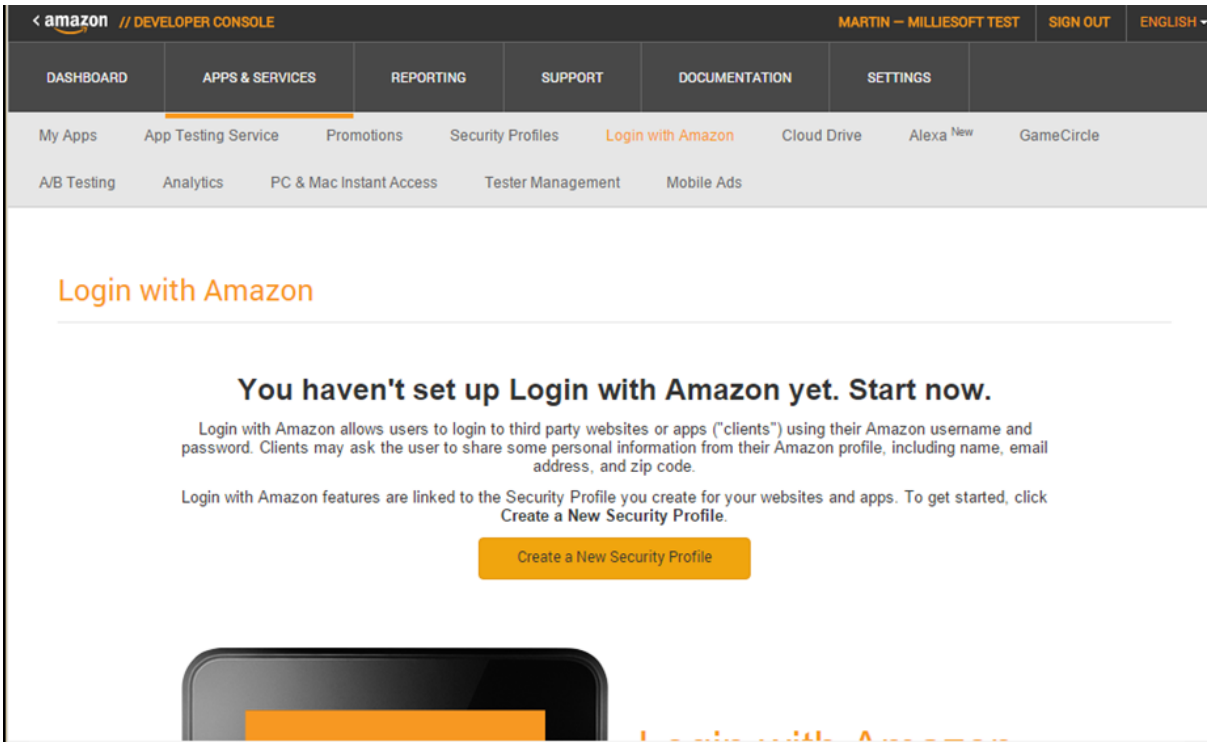

You aren't planning on releasing any apps, so select No to both questions, then press Save and Continue.



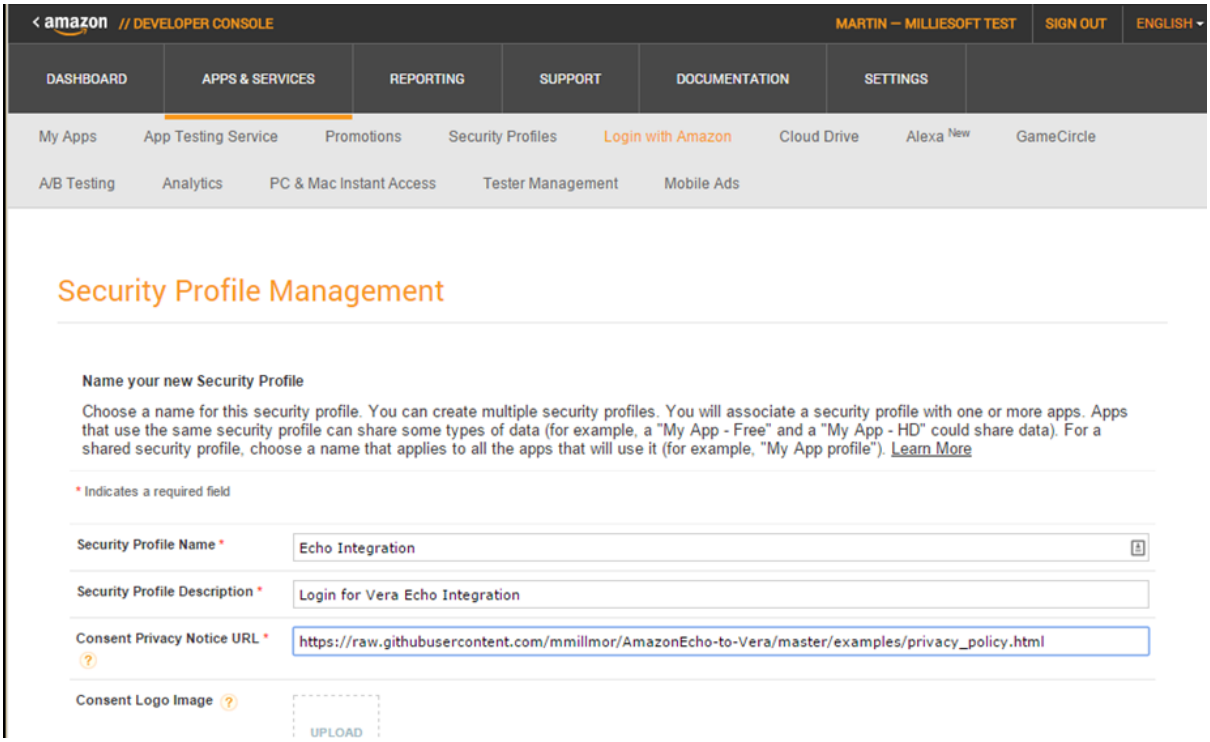

Click on Apps & Services



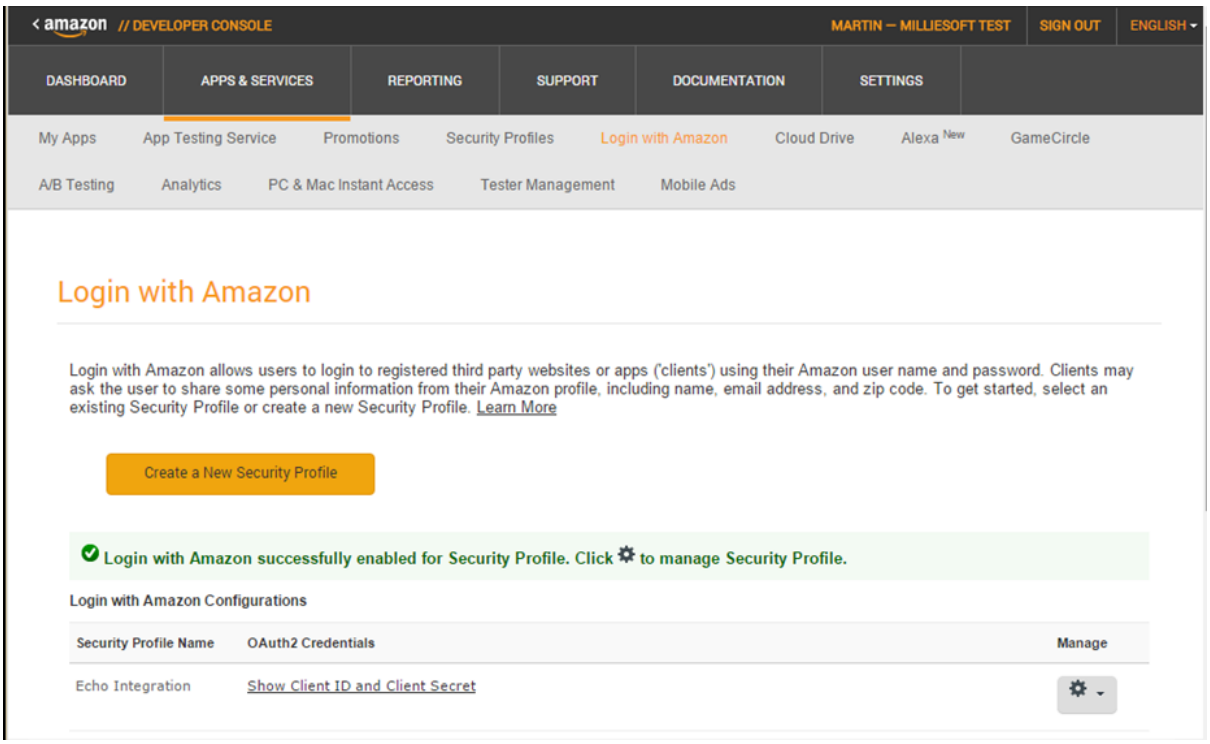

Click on Login with Amazon



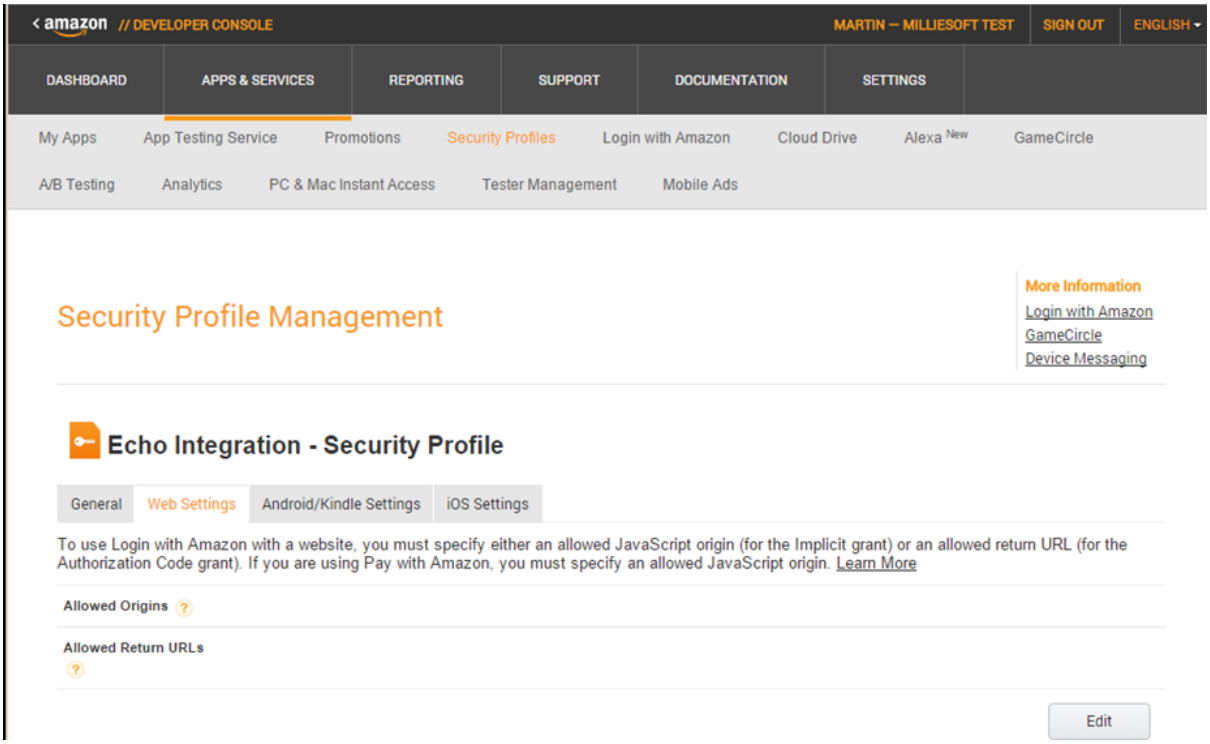

Click Create a New Security Profile



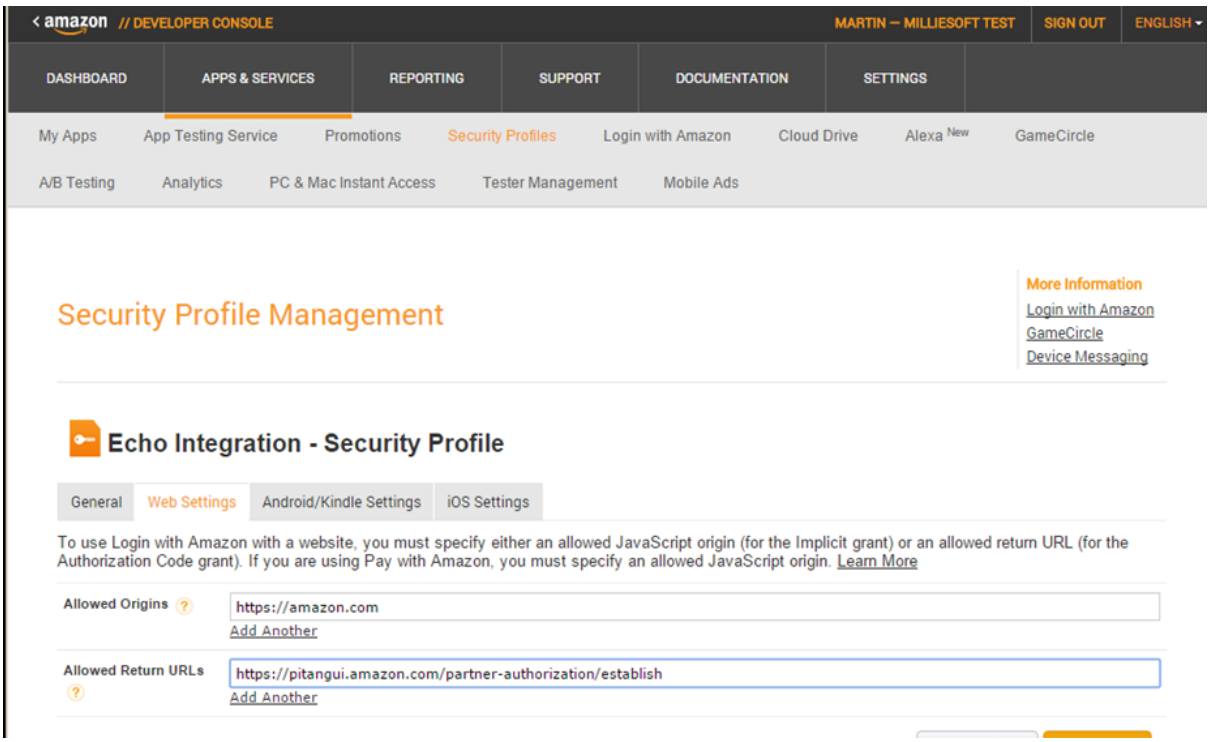

Enter any name and description and a URL to a privacy policy. You can use the one at https://raw.githubusercontent.com/mmillmor/AmazonEcho-to-Vera/master/examples/privacy_policy.html if you like. Scroll down and press Save.




Click on the Manage menu and pick Web Settings



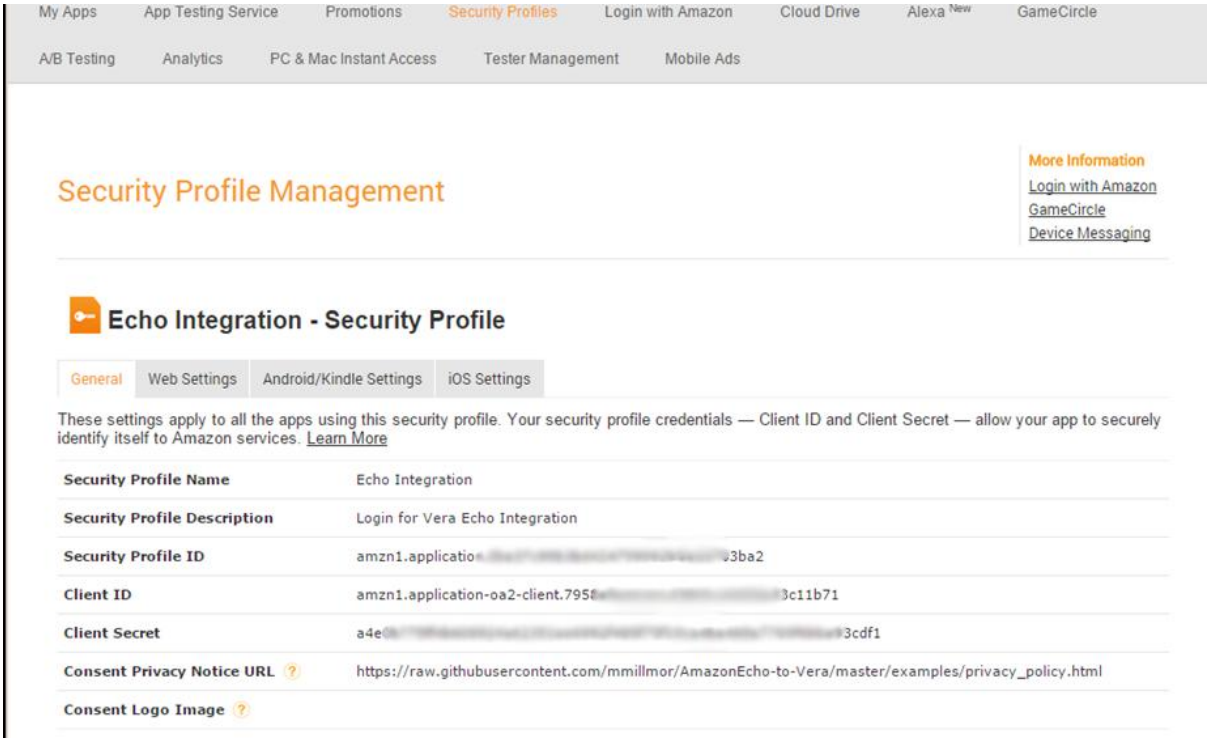

Click Edit




Set Allowed Origins to https://amazon.com and Allowed Return URLs to https://pitangui.amazon.com/partner-authorization/establish

Click Save.

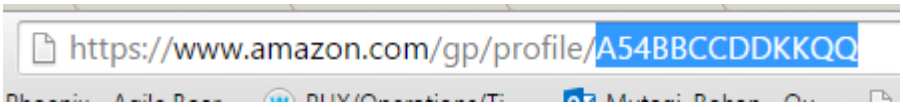
Click on General




In there you will see your Client ID and Client Secret. You will need those later

## Amazon Account Details

Go to https://amazon.com/profile while logged in with the account that your Echo is connected to. The URL will change to have your customer ID in the URL. You will need that later




## Contact Amazon

The final step is to contact Amazon with your information so that they can wire it all together. Send an e-mail to alexa-coho-submissions@amazon.com with the following details

| | |
|---|---|
| *Skill adapter display name* | The name for your integration, e.g. "Martin's Vera System" |
| *Skill adapter description* | A description, e.g. "Integration with my Vera" |
| *AWS Lambda function name* | This is the ARN string from the end of the coding step |
| *OAuth Client ID* | This is the Client ID from the Login step. |
| *OAuth Client Secret* | This is the Client Secret from the Login step |
| *OAuth Scope* | This is just the string "profile:user_id" |
| *OAuth authorization URL* | This is https://www.amazon.com/ap/oa |

| | |
|---|---|
| *OAuth token URL* | This is https://api.amazon.com/auth/o2/token |
| *Amazon Customer ID* | This is the customer ID from the last step above |

Amazon will respond with confirmation that it has all been set up, and then you can run discover and control devices on your Echo.