

This guide is meant for everyone to understand. It won't go too much into the inner workings or code since that can be confusing - but it will be extremely detailed in the steps....it won't assume ANY ASPECT is understood prior to this. I can elaborate on anything via Reddit or Github.

Step-by-Step Guide for Alexa & Roku Integration

1. Go here and open an Amazon AWS Lambda account (free for our use):

<https://aws.amazon.com/lambda/>



2. Hit get started and login to your amazon account, if not already.
3. Fill out information (select personal account)
 - a. Hit Create account and continue
 - b. Enter payment information – this will be free – you are NOT using their paid tiers. Then hit continue
 - c. It will verify you by calling your phone number, enter the pin number you see on the next screen (once you hit call).
 - d. After being verified, hit 'Continue to select your Support Plan'

Identity Verification

You will be called immediately by an automated system and prompted to enter the PIN number provided.

1. Provide a telephone number ✓

2. Call in progress ✓

3. Identity verification complete

Your identity has been verified successfully.

Continue to select your Support Plan

- e. Choose 'basic' (default choice) and hit continue.
 - f. *You have now created an Amazon Web Services account. We will come back to this later.*
- 4. Once an account is created, hit 'sign in to the console'
 - 5. You will be greeted with this screen:

AWS services

Find a service by name (for example, EC2, S3, Elastic Beanstalk).



- > Recently visited services
- > All services

Build a solution

Get started with simple wizards and automated workflows.



Launch a virtual machine

With EC2
~1 minutes



Build a web app

With Elastic Beanstalk
~6 minutes



Deploy a serverless microservice

With Lambda, API Gateway
~2 minutes



Host a static website

With S3, CloudFront, Route 53
~5 minutes



Create a backend for your mobile app

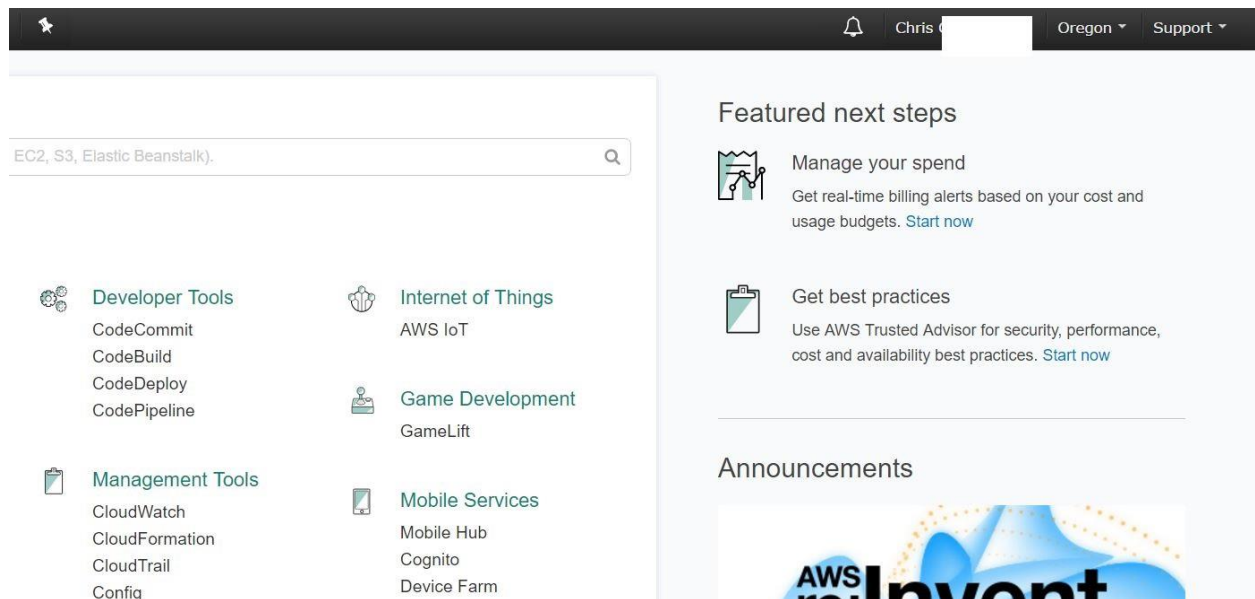
With Mobile Hub
~5 minutes



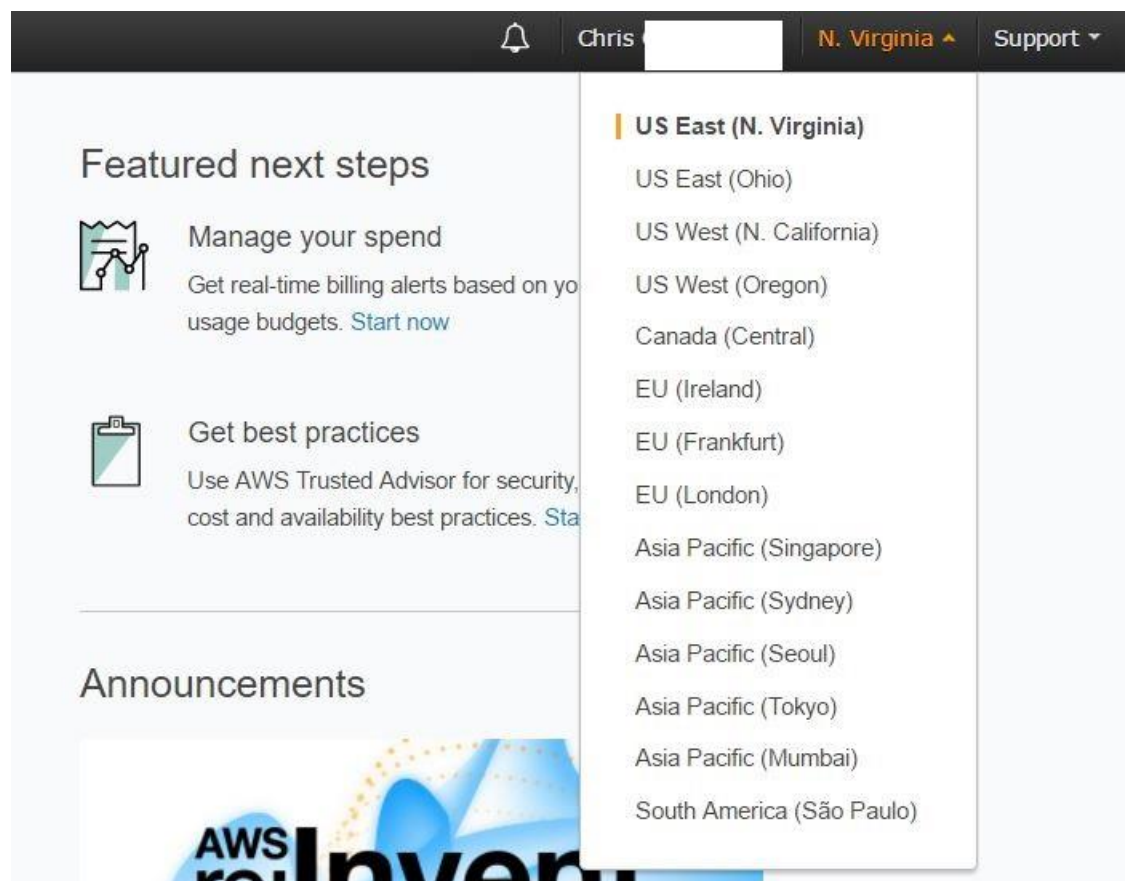
Register a domain

With Route 53
~3 minutes

6. In the type right, you will see, next to your name, a city (possibly Oregon)



a. You need to change this to US East (N. Virginia)



7. It will refresh with the same screen, but now N. Virginia will be in the top right

8. Now, in the search bar, type Lambda and select the first choice.
9. Once you are there, hit 'get started now'



10. Once at the Lambda page, choose 'Blank Function':

Select blueprint



Blueprints are sample configurations of event sources and Lambda functions. Choose a blueprint that best aligns with your desired scenario and customize as needed, or skip this step if you want to author a Lambda function and configure an event source separately. Except where otherwise noted, blueprints are licensed under [CC0](#).

Select runtime ▼

Filter

« < Viewing 1-9 of 70 > »

Blank Function Configure your function from scratch. Define the trigger and deploy your code by stepping through our wizard. custom	kinesis-firehose-syslog-to-json An Amazon Kinesis Firehose stream processor that converts input records from RFC3164 Syslog format to JSON. nodejs · kinesis-firehose	alexa-skill-kit-sdk-factskill Demonstrate a basic fact skill built with the ASK NodeJS SDK nodejs · alexa
kinesis-firehose-apachelog-to-j... An Amazon Kinesis Firehose stream processor that converts input records from Apache Common Log format to python2.7 · kinesis-firehose	cloudfront-modify-response-he... Blueprint for modifying CloudFront response header implemented in NodeJS. nodejs · cloudfront · response header	s3-get-object-python An Amazon S3 trigger that retrieves metadata for the object that has been updated. python2.7 · s3
config-rule-change-triggered An AWS Config rule that is triggered by configuration changes to EC2 instances. Checks instance types. nodejs4.3 · config	dynamodb-process-stream An Amazon DynamoDB trigger that logs the updates made to a table. nodejs · dynamodb	microservice-http-endpoint A simple backend (read/write to DynamoDB) with a RESTful API endpoint using Amazon API Gateway. nodejs · api-gateway

11. It will take you to a page called 'Configure Triggers'

a. Click inside the white open dotted box:

Configure triggers

You can choose to add a trigger that will invoke your function.

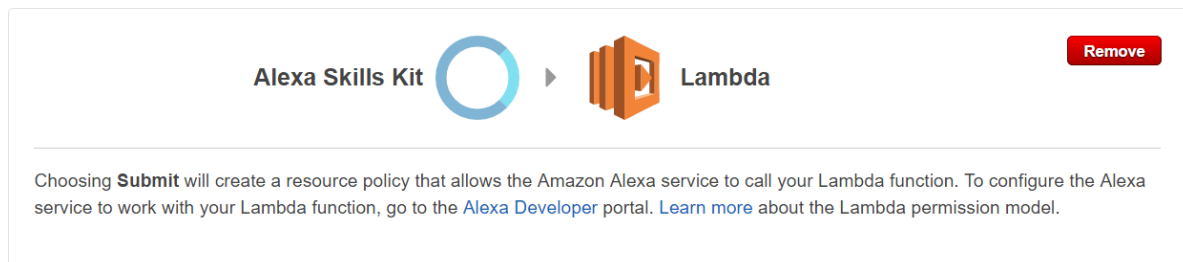


b. This will drop down a filter box, type in 'alexa' and choose 'Alexa Skills Kit'.

12. Once you have selected the Alexa Skills Kit, hit next

Configure triggers

You can choose to add a trigger that will invoke your function.



Cancel

Previous

Next

13. This will take you to the 'Configure function page'

- For now, we will only do a few things
- Under the first option, "Name" – easiest to do is just type AlexaRoku (you can name it whatever you want)
- In the description, I also put AlexaRoku
- Ignore next few options and the code section (we will edit later)

Configure function

A Lambda function consists of the custom code you want to execute. [Learn more](#) about Lambda functions.

Name*

Description

Runtime*

Lambda function code

Provide the code for your function. Use the editor if your code does not require custom libraries (other than the aws-sdk). If you need custom libraries, you can upload your code and libraries as a .ZIP file. [Learn more](#) about deploying Lambda functions.

Code entry type

```
1 exports.handler = (event, context, callback) => {  
2   // TODO implement  
3   callback(null, 'Hello from Lambda');  
4 };
```

e. Under Role Name – also type AlexaRoku (doesn't matter again, just something to remind you what this is for).

i. *I realized I put RokuAlexa (vs AlexaRoku) in my screenshot and now past that point to re-take a screenshot. But shows why it really doesn't matter for our purposes.*

You can define Environment Variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. [Learn more](#). For storing sensitive information, we recommend encrypting values using KMS and the console's encryption helpers.

Enable encryption helpers ☐

Environment variables

Lambda function handler and role

Handler* ⓘ

Role* ⓘ

Lambda will automatically create a role with permissions from the selected policy templates. Note that basic Lambda permissions (logging to CloudWatch) will automatically be added. If your function accesses a VPC, the required permissions will also be added.

Role name* ⓘ

Policy templates ⓘ

f. Leave everything else *as is*....hit next

Advanced settings

These settings allow you to control the code execution performance and costs for your Lambda function. Changing your resource settings (by selecting memory) or changing the timeout may impact your function cost. [Learn more](#) about how Lambda pricing works.

Memory (MB)* 128 ⓘ

Timeout* 0 min 3 sec

AWS Lambda will automatically retry failed executions for asynchronous invocations. You can additionally optionally configure Lambda to forward payloads that were not processed to a dead-letter queue (DLQ), such as an SQS queue or an SNS topic. [Learn more](#) about Lambda's [retry policy](#) and [DLQs](#). **Please ensure your role has appropriate permissions to access the DLQ resource.**

DLQ Resource Select resource ⓘ

All AWS Lambda functions run securely inside a default system-managed VPC. However, you can optionally configure Lambda to access resources, such as databases, within your custom VPC. [Learn more](#) about accessing VPCs within Lambda. **Please ensure your role has appropriate permissions to configure VPC.**

VPC No VPC ⓘ

Environment variables are encrypted at rest using a default Lambda service key. You can change the key below to one of your account's keys or paste in a full KMS key ARN.

KMS key (default) aws/lambda ⓘ

* These fields are required.

[Cancel](#)

[Previous](#)

[Next](#)

14. The next page will show you an overview of your settings. They should look something like this:



Alexa Skills Kit

Lambda function

Edit

Name AlexaRoku

Description AlexaRoku

Runtime Node.js 4.3

Environment variables

Handler index.handler

Role name* RokuAlexa

Policy templates

DLQ Resource

Memory (MB) 128

Timeout 3

VPC No VPC

KMS key (default) aws/lambda

15. Hit Create Function

16. It should now say it's been successfully created – we will come back to this after we do a few other things!

Qualifiers ▾

Test

Actions ▾

Congratulations! Your Lambda function "AlexaRoku" has been successfully created and configured with Alexa Skills Kit as a trigger. You can now click on the "Test" button to input a test event and test your function.

Code

Configuration

Triggers

Monitoring

Alexa Skills Kit

To configure the Alexa service to work with your Lambda function, go to the [Alexa Developer portal](#).

Delete

+ Add trigger

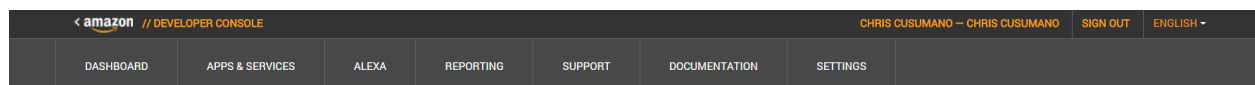
17. Now it's time to set up an Amazon Developer Alexa skill account

18. Head to <http://developer.amazon.com>

- In the top right, hit 'sign in'
- Sign in....this will automatically take you to the registration page.

- i. Usually all of your relevant personal info from your amazon account is filled in except for 'developer name or company name' – just enter your name.
 1. If anything else is missing and is required, feel free to fill in.
- ii. Hit save and continue
- iii. You will come to their agreement page, hit Accept and Continue at bottom right
- iv. The next page will take you to two questions for money-related skills/apps.
 1. It's asking you if you plan on making money / charging for skills or apps.
 2. Hit no for both and then hit save and continue
 - a. You can change this later if you decide to make skills in the future and want to charge people (this specific Roku skill cannot be made public due to how it works).

19. You will now come to their home page for the developer account. Choose 'Alexa' at the top:



Developer Communications

Announcements Notifications	
Introducing the ASK Built-in Library Developer Preview	Dec 7, 2016
Announcing the 12 Sponsored Teams for the Alexa Prize	Nov 14, 2016
Announcing the List Skill API	Oct 13, 2016
New Hands-Free Capabilities from the Alexa Voice Service (AVS)	Oct 6, 2016
Introducing the Flash Briefing Skill API	Oct 3, 2016
Alexa and Amazon Echo Now Available in the UK and Germany	Sep 14, 2016



Dashboard

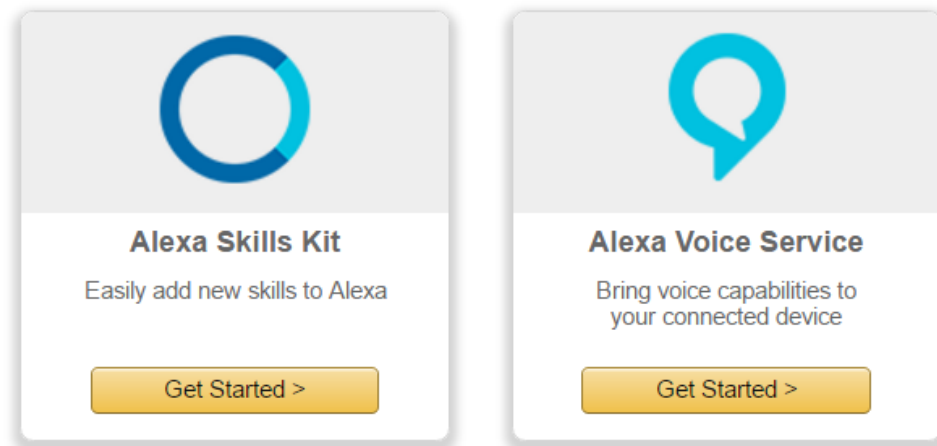
App Sales (Units) | [In-App Item Sales \(Units\)](#) | [Mobile Ad Earnings](#)
Nothing Found

Add a New App

20. Now choose 'get started' under Alexa Skills Kit (first option)

Get started with Alexa

Add new voice-enabled capabilities using the Alexa Skills Kit, or add voice-powered experiences to your c



21. Choose 'add new skill' at top right

22. You will come to a page of settings

- a. Leave Skill Type the same (Custom Interaction Model)
- b. If you are in the US, leave Language the same.
 - i. If not, or your Echo is set to a different language, choose the language that is set with your Echo. **They must match.**
- c. Under name, I just called it AlexaRoku (same as the lambda for consistency – but you can call this whatever your heart desires).
- d. Invocation name: Roku
 - i. This can be changed to your liking – **THIS IS THE WORD USED TO START THE SKILL** (Alexa, Tell Roku to blah blah blah) – you can call it whatever you desire.
- e. Leave everything else the same, choose next

Skill Type

Define a custom interaction model or use one of the predefined skill APIs. [Learn more](#)

☒ Custom Interaction Model

☐ Smart Home Skill API

☐ Flash Briefing Skill API

Language

Language of your skill

English (U.S.) ▼

Name

Name of the skill that is displayed to customers in the Alexa app. Must be between 2-50 characters.

AlexaRoku

Invocation Name

The name customers use to activate the skill. For example, "Alexa ask Tide Pooler...".

[Invocation Name Guidelines](#)

Roku

Global Fields

These fields apply to all languages supported by the skill.

Audio Player

Does this skill use the audio player directives? [Learn more](#)

☐ Yes ☒ No

23. You are brought to a page asking for 'intent schema' and 'sample utterances'

- i. In the simplest explanation, these are first asking for what the function is called (intent schema)....and then what voice command do you want to use to do that function (sample utterances).
- b. **Go to the files you downloaded**, within the 'Alexa Skill Voice Commands & Intents folder': open 'IntentSchema.txt'
 - i. This can be opened in Notebook, Wordpad, Word, etc – they all h
- c. Right click and select all, then copy
- d. Paste in the box of 'Intent Schema'

English (U.S.) ✓

Add New Language

Intent Schema

The schema of user intents in JSON format. For more information, see [Intent Schema](#).
Also see [built-in slots](#) and [built-in intents](#).

```
1 {  
2   "intents": [  
3     {  
4       "intent": "Home",  
5       "slots": []  
6     },  
7     {  
8       "intent": "Pandora",  
9       "slots": []  
10    },  
11  ]  
}
```

- e. Similarly, open the file called 'SampleUtterances.txt'
- f. Select all, copy
- g. Paste in the 'Sample Utterances' box

Custom Slot Types (Optional)

Custom slot types to be referenced by the Intent Schema and Sample Utterances

For general information about custom slots, see [Custom Slot Types](#).

Example: TOPPINGS - cheese | onions | ham (note: newlines displayed as | for brevity)

Add Slot Type

Sample Utterances

These are what people say to interact with your skill. Type or paste in all the ways that people can invoke the intents. [Learn more](#)

Up to 3 of these will be used as Example Phrases, which are hints to users.

```
1 Hulu hulu  
2 Hulu set to hulu  
3 Hulu open hulu  
4 Hulu start hulu  
5 Hulu go to hulu  
6  
7 Amazon amazon  
8 Amazon set to amazon  
9 Amazon open amazon  
10 Amazon start amazon  
11 Amazon go to amazon
```

Save

Submit for Certification

Next

I is being built

- h. Click Next

24. You will then be brought to this page:

Global Fields

These fields apply to all languages supported by the skill.

Endpoint

Service Endpoint Type:

☒ **AWS Lambda ARN (Amazon Resource Name)** ⓘ ☐ HTTPS

Recommended

AWS Lambda is a server-less compute service that runs your code in response to events and automatically manages the underlying compute resources for you.

[More info about AWS Lambda](#)

[How to integrate AWS Lambda with Alexa](#)

Account Linking

Do you allow users to create an account or link to an existing account with you?

☐ Yes ☒ No

[Learn more](#)

25. At this point, you want to open another browser tab or a separate browser **(LEAVE THIS PAGE OPEN)**.

26. In the other tab, navigate to: <http://aws.amazon.com> (we are going back to the lambda function for something we need)

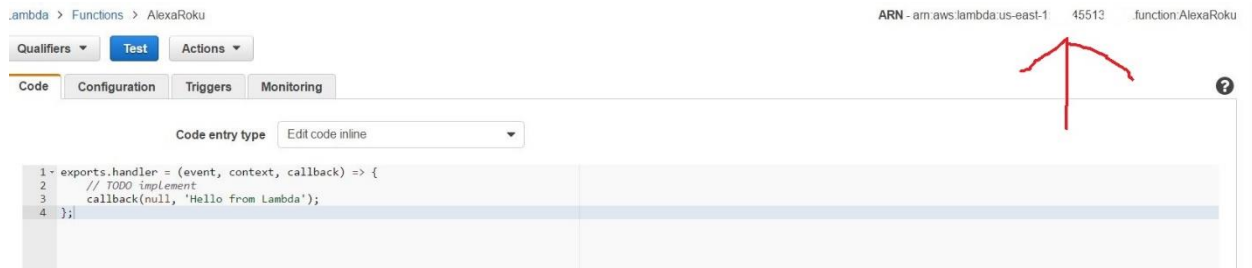
- In the top right, hit Sign in to the console
- Once arrived to the home page, search for Lambda (may already be showing in 'recently visited services') and click it.
 - Same as what we did before
- You will now be brought to a page that should be showing your previously created function 'AlexaRoku' (or whatever you called it)

Lambda > Functions ?

[Create a Lambda function](#) Actions

	Function name	Description	Runtime	Code size	Last Modified
<input type="radio"/>	AlexaRoku	AlexaRoku	Node.js 4.3	216 bytes	23 minutes ago

- Double click the function (anywhere on the line).
- This will bring you to the code page.
 - In the top right of the page, you will see a bolded **ARN:** with a bunch of numbers and letters following (part of mine is blocked out due to security reasons):



- ii. Highlight and copy the full ARN address, starting with the *lowercase* 'arn' – for example: arn.aws.lambda.us-east-1:45513:function:AlexaRoku and ending with the name of your function (in my example: 'AlexaRoku')

27. Head back to the original tab for the Amazon Developer Skill page (what we left doing a bit ago):

Global Fields

These fields apply to all languages supported by the skill.

Endpoint

Service Endpoint Type:

☒ **AWS Lambda ARN (Amazon Resource Name)** ⓘ ☐ HTTPS

Recommended

AWS Lambda is a server-less compute service that runs your code in response to events and automatically manages the underlying compute resources for you.

[More info about AWS Lambda](#)

[How to integrate AWS Lambda with Alexa](#)

Pick a geographical region that is closest to your target customers: ⓘ

28. Now Choose AWS Lambda ARN (if not already selected)

29. Choose North America

30. In the box for North America that now appears, enter that arn address you just copied

English (U.S.)


Add New Language

Global Fields

These fields apply to all languages supported by the skill.

Endpoint

Service Endpoint Type:


☒ **AWS Lambda ARN (Amazon Resource Name)**  ☐ **HTTPS**

Recommended

AWS Lambda is a server-less compute service that runs your code in response to events and automatically manages the underlying compute resources for you.

[More info about AWS Lambda](#)

[How to integrate AWS Lambda with Alexa](#)

Pick a geographical region that is closest to your target customers: 

☒ **North America** ☐ **Europe**

North America

arn:aws:lambda:us-east-1:5337:function:AlexaRoku

31.Hit Next


- If correctly copied, it will bring you to the next page
- If you get an error at the bottom of the page, in red – you copied the arn address incorrectly – make sure to start with the lowercase arn and *no spaces* before the first character or after the final



32.At this point, you should come to the testing page and the first four options on the left should be check-marked in green:

English (U.S.)

Add New Language

Skill Information	✓
Interaction Model	✓
Configuration	✓
Test	✓
Publishing Information	⌵
Privacy & Compliance	⌵

 Please complete the Interaction Model tab to start testing this skill.

 **Enabled** This skill is enabled for testing on your account. 

Once you have completed testing on your device, please complete the Description and Publishing Information tab, then submit the skill for certification.

If it passes Amazon's testing and certification process, it will become available to Alexa end users.

You will be able to see your skill in the Skills tab in Alexa App and you can enable the skill and start testing.

After completing your testing please submit the skill for certification. If it passes Amazon's testing and certification process, it will become available to Alexa end users.


The skill is available in "Skills > Your Skills" page of the Alexa App when you select 'Yes' above. You can then enable the skill and test its functionality by asking Alexa, **ask roku**

Voice Simulator

Hear how Alexa will speak a response entered in plain text or SSML. [Learn more about supported SSML tags.](#)

For example: Here is a word spelled out: `<say-as interpret-as="spell-out">hello</say-as>`.

Here is a word spelled out: `<say-as interpret-as="spell-out">hello</say-as>`.

Listen 

Service Simulator

Use Service Simulator to test your lambda function: `arn:aws:lambda:us-east-1:529455139337:function:AlexaRoku`

33. At the top left of the page, you will see 'back to All Skills'
- Click this.
 - If it asks you to save, hit save – although it shouldn't, as it should have autosaved.

34. At this point, the skill on the Developer side of things is complete. You can never technically 'publish' the skill to the public, as it won't be approved – so you will remain in 'test' mode for your account, which is exactly what you want.

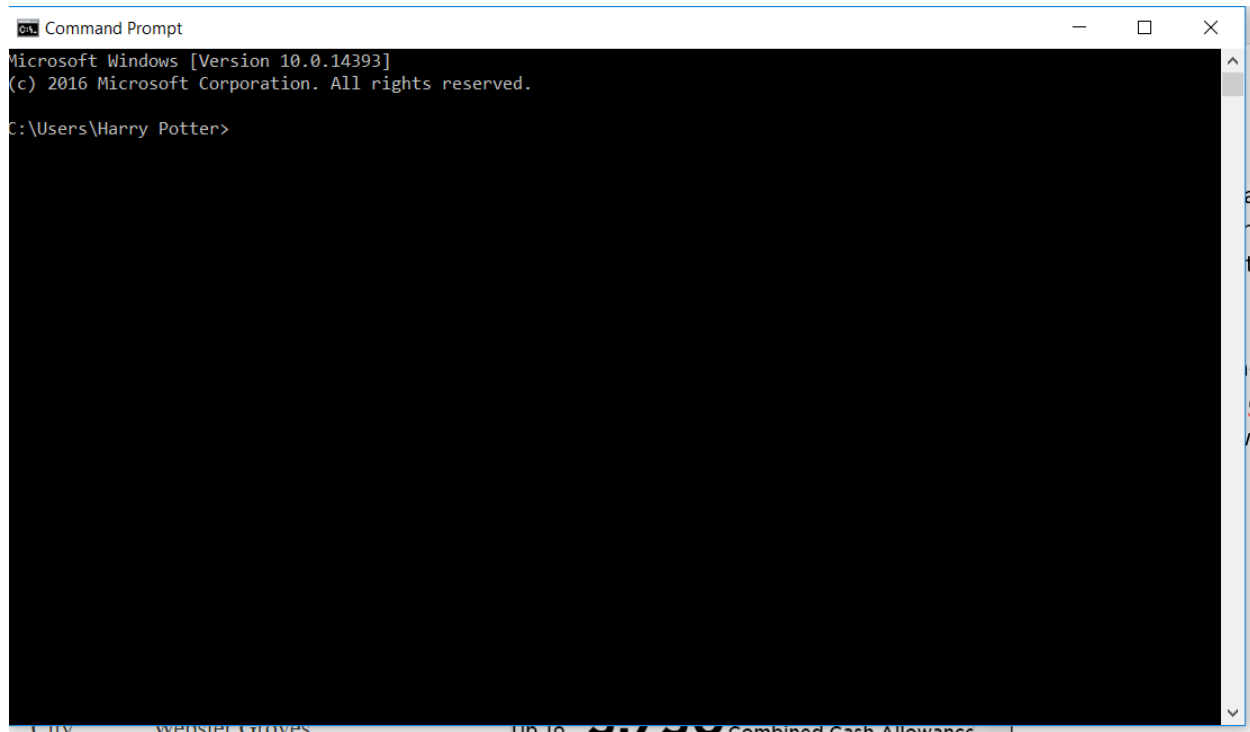
35. We will come back to this page soon, though – to get your App ID number.

We will now install the Node server program and use the file to get a quick piece of information that will allow us to do the next step.

At this point, we will want to set up port forwarding for your router to allow for Alexa to talk to your computer through the node server (which we will install later).

This is not the absolute *safest* way of doing things, as it allows for access externally (if someone knew both your public IP and port number). There are thousands of port numbers, however – you are choosing whichever you want and that adds *some* level of security – but still not full-proof. Just want that pointed out.

36. This part is a tad harder to explain for everyone universally, since their router may have different steps – but I will try.
- Head to <http://www.whatismyip.com>
 - Make sure you don't have a VPN running during this time.
 - Copy or write down this address
 - In Windows, head to the start menu on bottom left
 - In the search part (at the bottom) - Type: cmd and choose the first option (will be 'Command Prompt')
 - This will open up:



As you can see, I am fan of Harry Potter since I have named my computer that.

If you're not a fan of Harry Potter, you're not my friend and you should not be able to do this skill 😊.

- e. Type: ipconfig and hit enter
- f. A list of items will appear....find the one with 'default gateway' and numbers next to it. It will likely be something along the lines of 192.168.0.1 or 192.168.1.1 (possibly others, but those are most often)

```
Node.js command prompt

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . : fe80::c78:9056:
IPv4 Address. . . . . : 192.168.1.103
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1

Ethernet adapter Bluetooth Network Connection:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

Tunnel adapter Reusable ISATAP Interface {4A5DE2AA-EDC6-47C

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

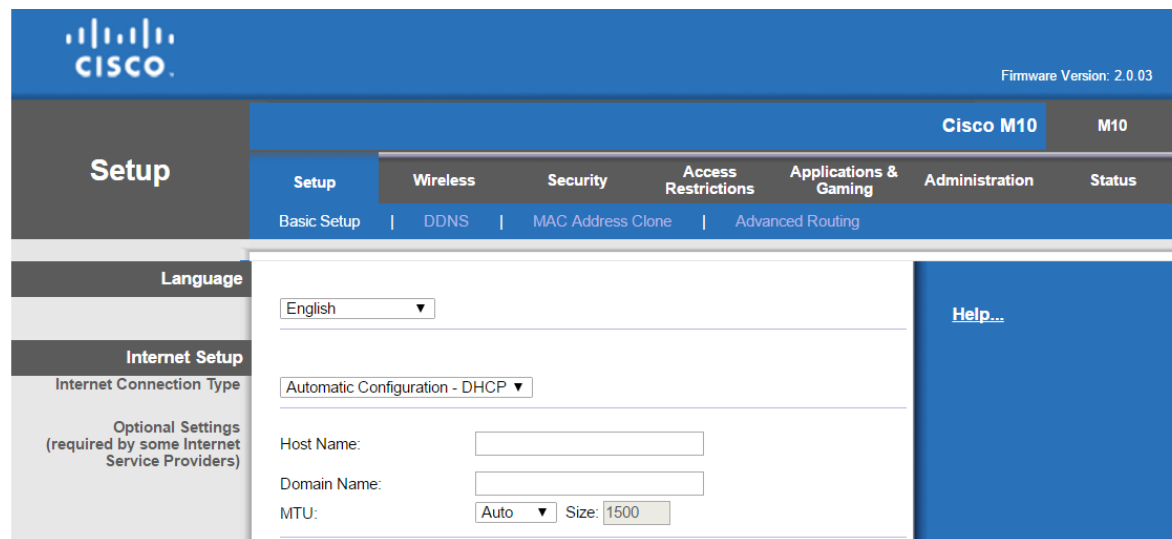
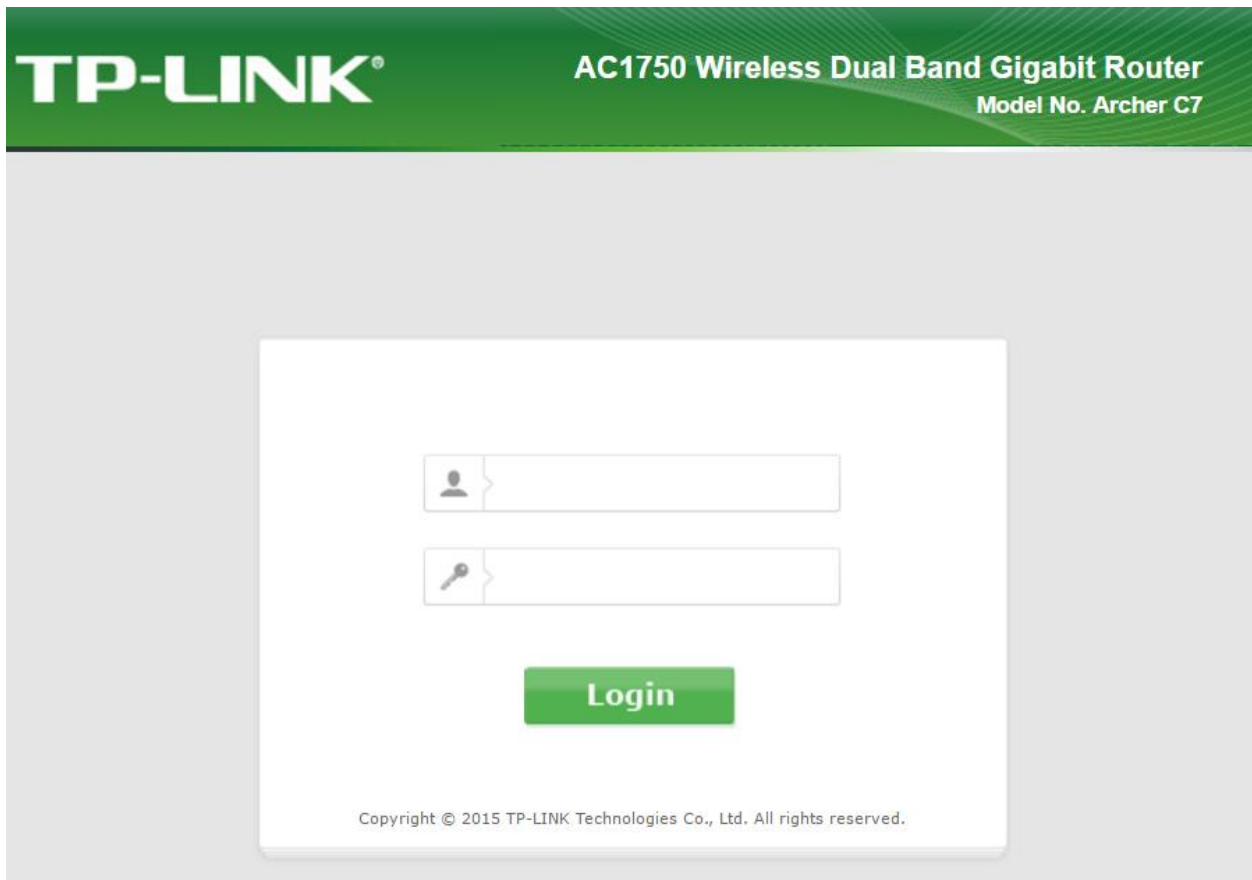
Tunnel adapter Local Area Connection* 14:

Connection-specific DNS Suffix . :
IPv6 Address. . . . . : 2001:0:4137:5
Link-local IPv6 Address . . . . : fe80::28d1:3e
Default Gateway . . . . . : ::

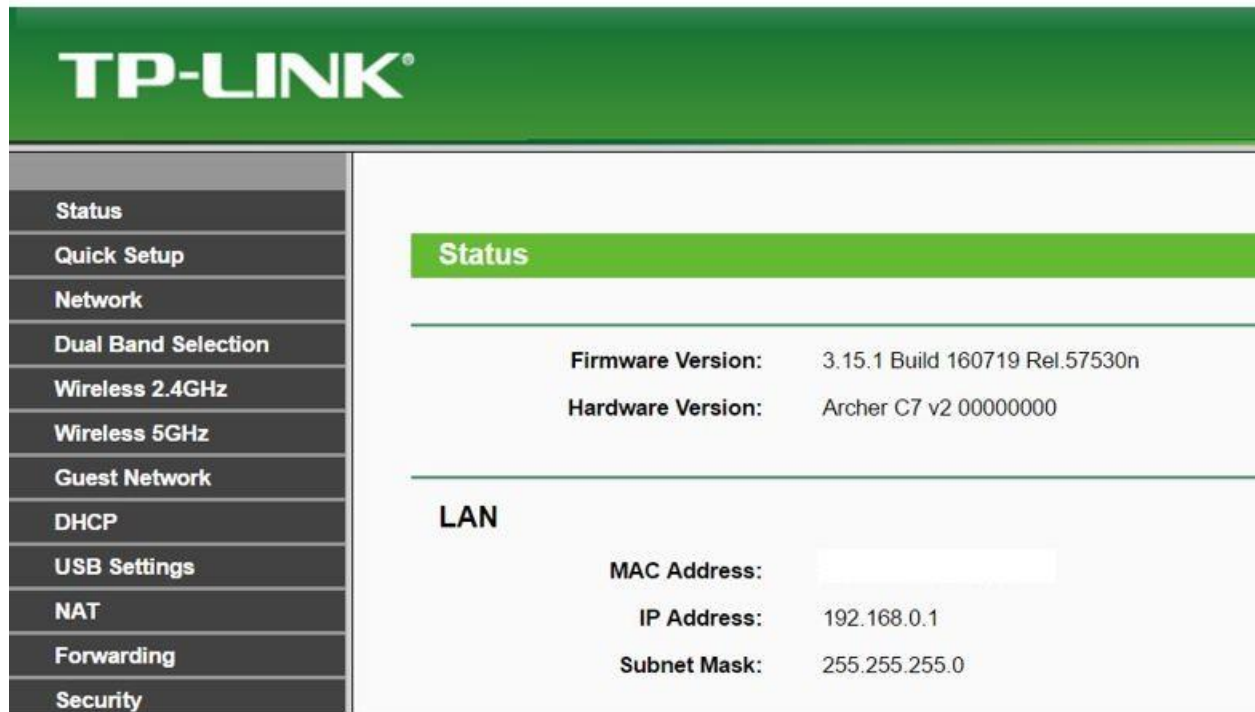
C:\Users\Harry Potter>
```

- g. Write that number down.
- h. Also write down the number next to IPv4 address (2 above Default Gateway). This will be very similar to the default gateway number above, but with a small difference in the final number.
- i. Open a browser and type that first (default gateway) number in the web address bar and hit enter.
 - i. For my example: it is 192.168.1.1
- j. This will take you to the admin page of your router or internet gateway. My example below from TP-Link (yours will be different, based on brand/model).
 - i. It is often password protected and hopefully you know this. If you've never accessed this, chances are it's the default for your router brand and model (google 'default admin login and password for xxxx xxxx router' and it usually comes up).

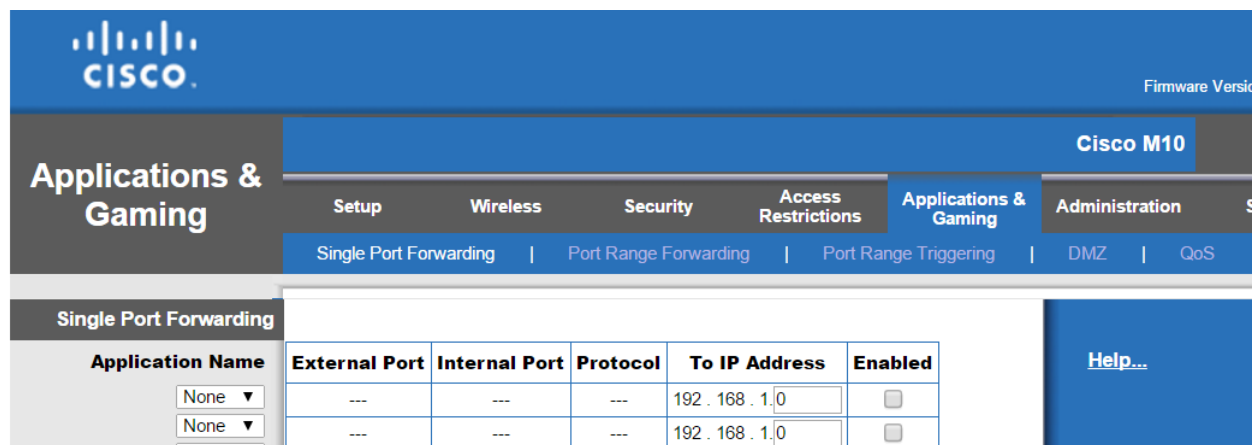
TP-Link takes you to a landing page where you enter a login. Cisco however, gives a pop-up asking for the login and password. You will get one of the two.



- k. Once you log in,
- You will come to a home page based on your router.
- l. From there, you are looking for a 'forwarding' option.
- m. Within Tp-Link, this is an option on the left



- i. In Cisco however, it is within 'Applications & Gaming'.



- n. If you aren't familiar, I'd either google or keep clicking through the options until you come across the section.
- o. Once you find, it will look similar to this:

The image shows the Cisco M10 web interface for 'Applications & Gaming' configuration. The top navigation bar includes 'Setup', 'Wireless', 'Security', 'Access Restrictions', 'Applications & Gaming' (selected), and 'Administration'. Below this, sub-navigation links are 'Single Port Forwarding', 'Port Range Forwarding', 'Port Range Triggering', 'DMZ', and 'QoS'. The 'Single Port Forwarding' section is active, showing a table with two entries. Each entry has a dropdown for 'Application Name' (set to 'None'), and fields for 'External Port', 'Internal Port', 'Protocol', 'To IP Address', and an 'Enabled' checkbox. The 'To IP Address' field is set to '192.168.1.0'.

Application Name	External Port	Internal Port	Protocol	To IP Address	Enabled
None ▼	---	---	---	192.168.1.0	<input type="checkbox"/>
None ▼	---	---	---	192.168.1.0	<input type="checkbox"/>

Or this:

The image shows the TP-Link 'Virtual Servers' configuration page. It features a green header with the title 'Virtual Servers' and a note: 'Note: Make sure the nat is enable if you want the Virtual Servers configuration take effect'. Below the note is a table with columns: 'ID', 'Service Port', 'Internal Port', 'IP Address', 'Protocol', 'Status', and 'Modify'. Under the table, there are four buttons: 'Add New...', 'Enable All', 'Disable All', and 'Delete All'. At the bottom, there are 'Previous' and 'Next' navigation buttons.

ID	Service Port	Internal Port	IP Address	Protocol	Status	Modify
<div> Add New... Enable All Disable All Delete All </div>						

- p. You now are going to set up port forwarding for Alexa to be able to access your Node server (steps later on) and then the node server be able to send commands to the Roku
 - i. You should have an option for External (or Service) port, Internal Port, Protocol and 'IP address' – see above for Cisco version, TP-link version below:

Add or Modify a Virtual Server Entry

Service Port:	<input type="text"/>	(XX-XX or XX)
Internal Port:	<input type="text"/>	(XX, Only valid for single Service Port or leave it blank)
IP Address:	<input type="text"/>	
Protocol:	<input type="text" value="ALL"/>	▼
Status:	<input type="text" value="Enabled"/>	▼
Common Service Port:	<input type="text" value="--Select One--"/>	

ii. In External/Service port, this is the port **you** choose

1. Here is a full list of TCP port numbers – there are thousands. I am not 100% positive, but pretty sure you can choose any one of them as your port number, so long as TCP is listed next to it - (this helps make it a bit more secure, since there are so many for someone to try to 'guess' or somehow get ahold of).

https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

- iii. In internal port, if needed to be filled in, type the same port as external
- iv. Finally, in IP – *you are typing the IP of the computer you will be running the node server on (we will do this part later).*
 1. *This is the IP4V IP from above*

```
Node.js command prompt
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Wireless LAN adapter Wi-Fi:
Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . : fe80::c78:9056:
IPv4 Address. . . . . : 192.168.1.103
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1
Ethernet adapter Bluetooth Network Connection:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Tunnel adapter Reusable ISATAP Interface {4A5DE2AA-EDC6-47C
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Tunnel adapter Local Area Connection* 14:
Connection-specific DNS Suffix . :
IPv6 Address. . . . . : 2001:0:4137:5
Link-local IPv6 Address . . . . : fe80::28d1:3a
Default Gateway . . . . . : ::
C:\Users\Harry Potter>
```

- v. If you have the option to change 'protocol', set it to TCP – but leaving it as Both is also no issue (I have it set as both).
- vi. Finally, hit 'enable' or save. This will now allow Alexa to talk to your computer.

37. Now we can install the node server.

38. Head to <http://www.nodejs.org>

39. Download the current version with latest features (although 'recommended' version also works fine).

Node.js® is a JavaScript runtime built on [Chrome's V8 JavaScript engine](#).

Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js' package ecosystem, [npm](#), is the largest ecosystem of open source libraries in the world.

Download for Windows (x64)

v6.9.2 LTS

Recommended For Most Users

v7.3.0 Current

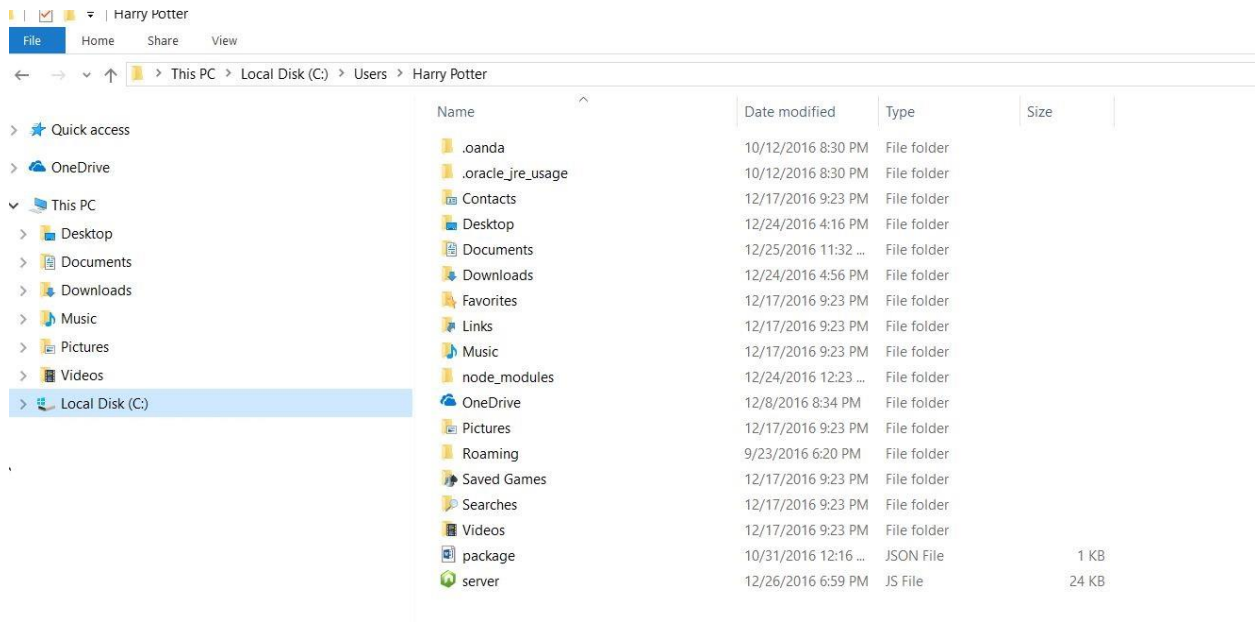
Latest Features

[Other Downloads](#) | [Changelog](#) | [API Docs](#)

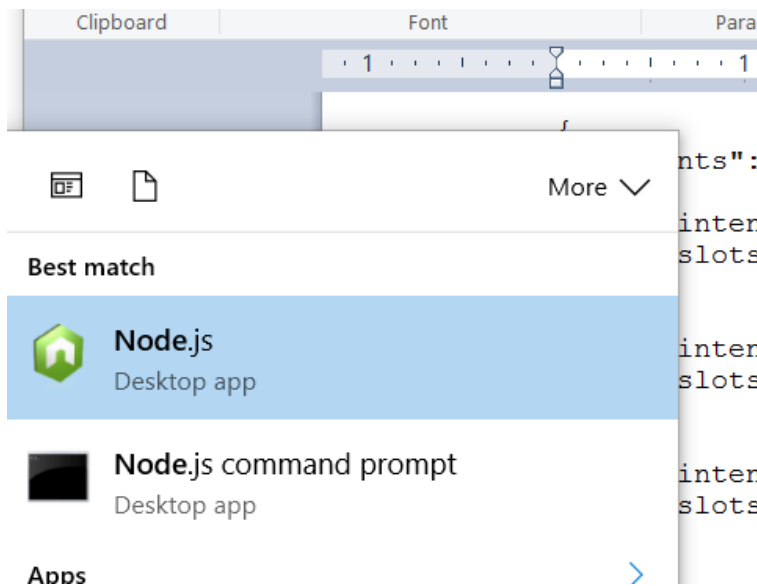
[Other Downloads](#) | [Changelog](#) | [API Docs](#)

40. Open and install like any other program.

- a. Hit next all the way through (this is a clean install – nothing special is asked to be installed or something stupid – only installs the program).
- b. Now, we need to head to the files you downloaded from github, in the folder 'Node Server file for Computer'
 - i. The two files, 'package' and 'server' need to be placed in your 'users' folder
 - ii. For windows, this means: My Computer or This PC (depending on the Windows)
 1. Then Local Disk C:
 2. Then Users
 3. Then choose the user folder you are currently on (most people are just one – it is the same as your computer name)
 4. Within that, folder, paste those two files.

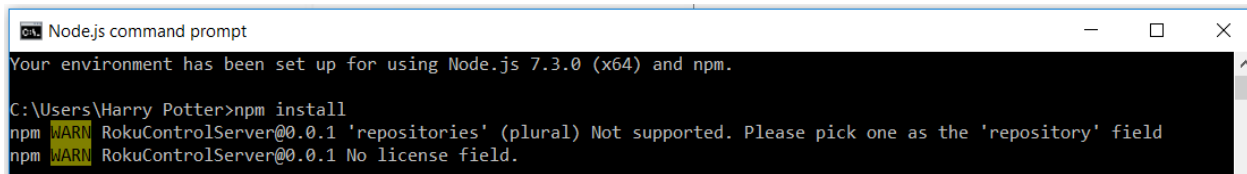


- c. Once completed, go to Start Menu again, similar to how we have done for command prompt (search part at bottom)
 - i. Search for node
 1. You will see multiple options
 2. Click on: Node.js command prompt



- d. Once you open Node.js command prompt, it will look very similar to the command prompt you have been using.
- e. Type : npm install
- f. Hit enter

- g. You should see this, so long as those files were copied to your user folder:



```
Node.js command prompt
Your environment has been set up for using Node.js 7.3.0 (x64) and npm.

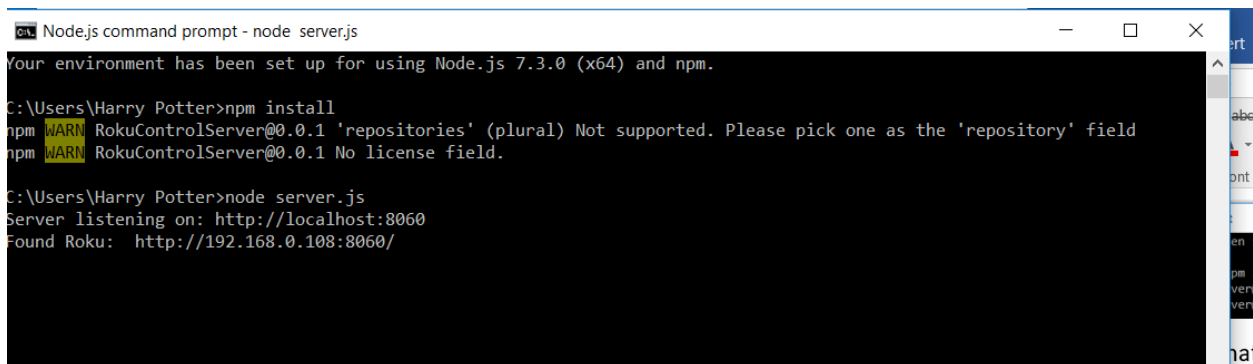
C:\Users\Harry Potter>npm install
npm WARN RokuControlServer@0.0.1 'repositories' (plural) Not supported. Please pick one as the 'repository' field
npm WARN RokuControlServer@0.0.1 No license field.
```

41. Leaving that command prompt open...for the sake of this being easy to narrow down....**if you have more than one Roku....unplug all that you are not setting this up for!**

- a. If you are wanting set up more than one, still do this – we will get to that later.

42. now type: node server.js

- a. You will now see something like this:



```
Node.js command prompt - node server.js
Your environment has been set up for using Node.js 7.3.0 (x64) and npm.

C:\Users\Harry Potter>npm install
npm WARN RokuControlServer@0.0.1 'repositories' (plural) Not supported. Please pick one as the 'repository' field
npm WARN RokuControlServer@0.0.1 No license field.

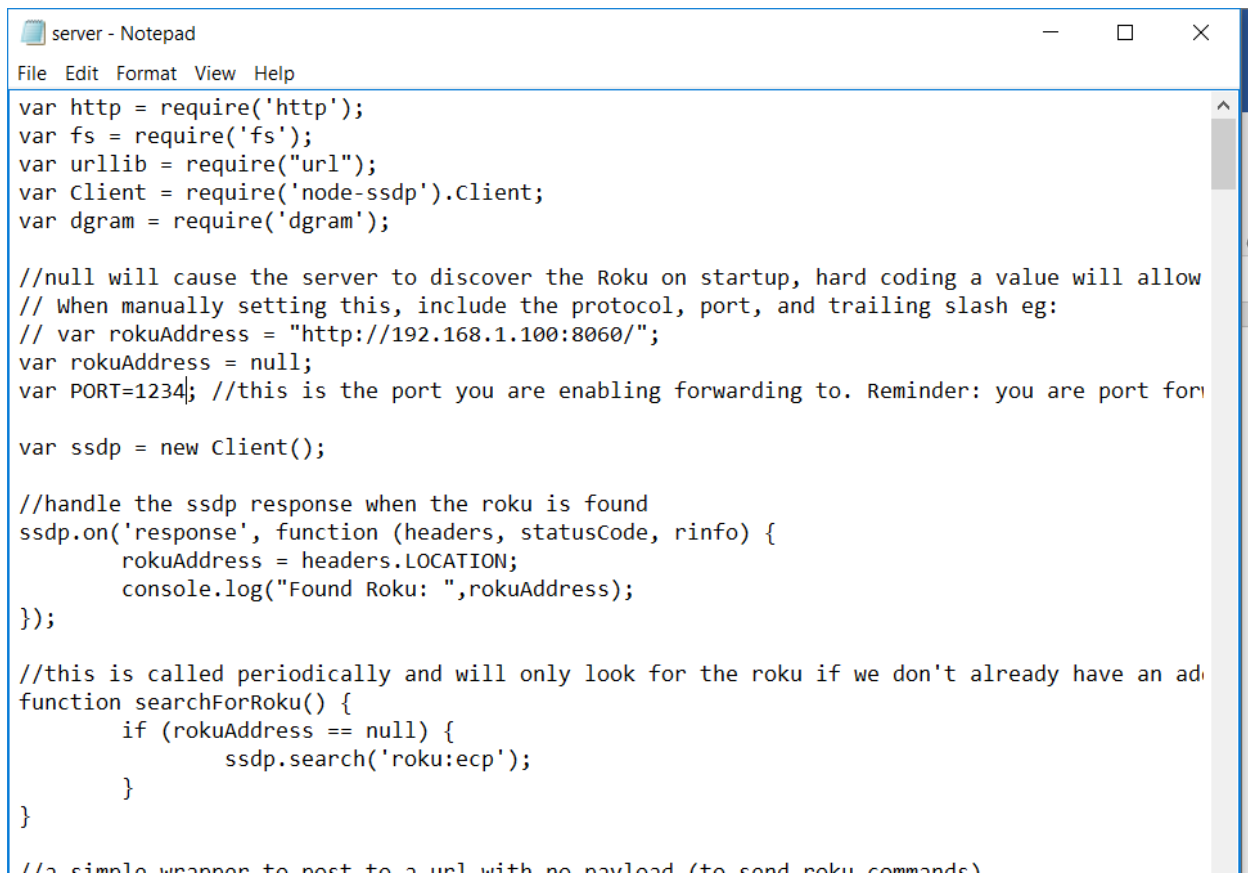
C:\Users\Harry Potter>node server.js
Server listening on: http://localhost:8060
Found Roku: http://192.168.0.108:8060/
```

- b. This will show you the only roku you have plugged in and connected.
- i. If others are plugged in, it isn't *as* simple to know which IP belongs to which Roku
 - ii. Plus, it saves steps in this guide to find out in another way ☺
- c. Write down this IP address!

43. **Now.....let's head back to where you saved those two server files.**

44. Right click on the 'server.js' file and hit 'open with' and choose 'Notepad'

- a. You can choose something different than Notepad, if you know what we are doing, but everyone should have Notepad as a default.
- b. You will get something similar to this:



```
server - Notepad
File Edit Format View Help
var http = require('http');
var fs = require('fs');
var url = require('url');
var Client = require('node-ssdp').Client;
var dgram = require('dgram');

//null will cause the server to discover the Roku on startup, hard coding a value will allow
// When manually setting this, include the protocol, port, and trailing slash eg:
// var rokuAddress = "http://192.168.1.100:8060/";
var rokuAddress = null;
var PORT=1234; //this is the port you are enabling forwarding to. Reminder: you are port forwarding
var ssdp = new Client();

//handle the ssdp response when the roku is found
ssdp.on('response', function (headers, statusCode, rinfo) {
    rokuAddress = headers.LOCATION;
    console.log("Found Roku: ", rokuAddress);
});

//this is called periodically and will only look for the roku if we don't already have an address
function searchForRoku() {
    if (rokuAddress == null) {
        ssdp.search('roku:ecp');
    }
}

//a simple wrapper to post to a url with no payload (to send roku commands)
```

45. Remember that port you chose a few steps back?

Change the number 1234 (highlighted yellow below) to your number:

```
server - Notepad
File Edit Format View Help
var http = require('http');
var fs = require('fs');
var url = require('url');
var Client = require('node-ssdp').Client;
var dgram = require('dgram');

//null will cause the server to discover the Roku on startup, hard coding a value will allow
// When manually setting this, include the protocol, port, and trailing slash eg:
// var rokuAddress = "http://192.168.1.100:8060/";
var rokuAddress = null;
var PORT=1234; //this is the port you are enabling forwarding to. Reminder: you are port forwarding
var ssdp = new Client();

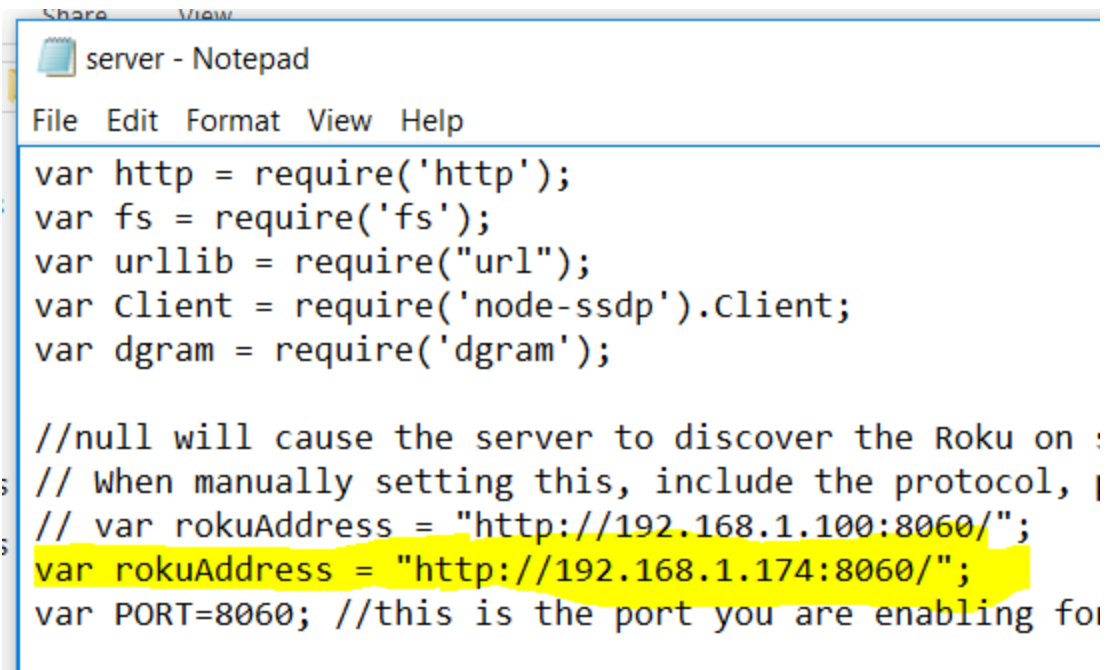
//handle the ssdp response when the roku is found
ssdp.on('response', function (headers, statusCode, rinfo) {
    rokuAddress = headers.LOCATION;
    console.log("Found Roku: ", rokuAddress);
});

//this is called periodically and will only look for the roku if we don't already have an address
function searchForRoku() {
    if (rokuAddress == null) {
        ssdp.search('roku:ecp');
    }
}

//simple wrapper to post to a url with no payload (to send roku commands)
```

46. IF YOU ONLY HAVE ONE ROKU in your house, you can hit file, save and close out

- a. If you have multiple Rokus....we need to do one extra step.
- b. Remember that IP address you **just** wrote down? Under: "Found Roku: <http://192.xxx.x.xxx>?"
- c. Next to var rokuAddress = null; change it to (WITH quotation makes) your IP address: (example): "http://192.168.0.108:8060/"
 - i. Make sure to include the http:// AND the last / in the end – and all in quotation marks, as shown below:



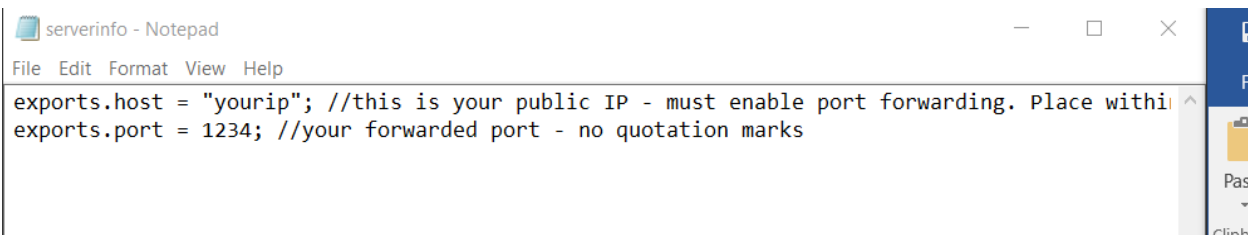
```
var http = require('http');
var fs = require('fs');
var url = require('url');
var Client = require('node-ssdp').Client;
var dgram = require('dgram');

//null will cause the server to discover the Roku on :
// When manually setting this, include the protocol, |
// var rokuAddress = "http://192.168.1.100:8060/";
var rokuAddress = "http://192.168.1.174:8060/";
var PORT=8060; //this is the port you are enabling for
```

47.If you only had one Roku, you can leave this as null and it will auto-find the Roku. The problem arises when you have multiple – you will want the specific IP address so it knows which one to send a command to.

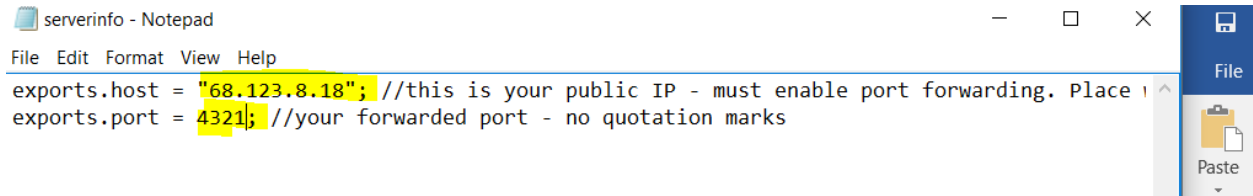
48.Nowwww, let's go to the files you downloaded, the folder: "AWS Files – Lambda"

- a. Right click on the file 'serverinfo'
- b. Open with Notepad again
- c. You will see this:



```
exports.host = "yourip"; //this is your public IP - must enable port forwarding. Place within
exports.port = 1234; //your forwarded port - no quotation marks
```

- d. Replace "yourip" with the IP address you wrote down in the **very beginning** – from whatismyip.com
 - i. This is your public IP. Leave OUT the http:// - ONLY the number
- e. For the second line, replace the numbers 1234 with the port YOU CHOSE!
- f. Something like below:



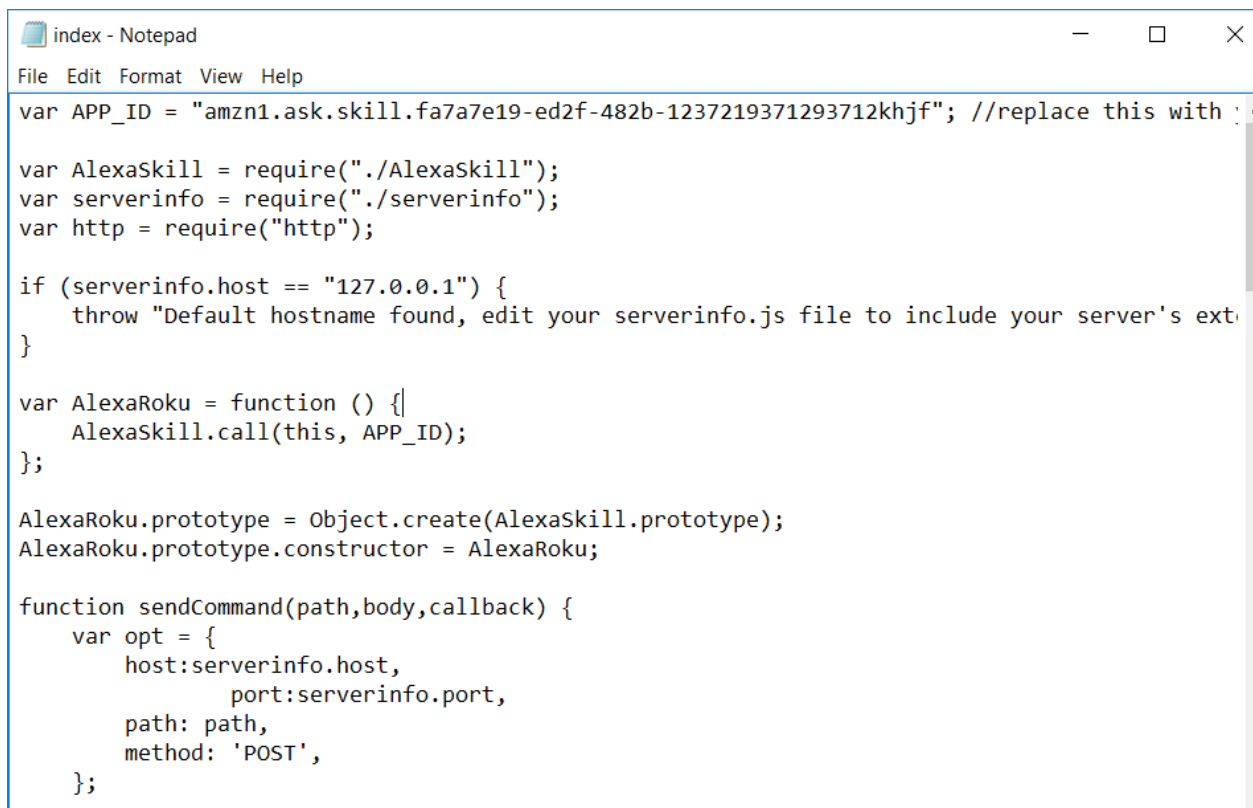
```
serverinfo - Notepad
File Edit Format View Help
exports.host = "68.123.8.18"; //this is your public IP - must enable port forwarding. Place
exports.port = 4321; //your forwarded port - no quotation marks
```

49. Go to file, hit save.

50. WE ARE ALMOST DONE!

51. In the same folder, "AWS Files – Lambda"

- a. Right click 'index'
- b. 'Open with' notepad
- c. You'll see something like this:



```
index - Notepad
File Edit Format View Help
var APP_ID = "amzn1.ask.skill.fa7a7e19-ed2f-482b-1237219371293712khjf"; //replace this with
var AlexaSkill = require("./AlexaSkill");
var serverinfo = require("./serverinfo");
var http = require("http");

if (serverinfo.host == "127.0.0.1") {
  throw "Default hostname found, edit your serverinfo.js file to include your server's ext
}

var AlexaRoku = function () {
  AlexaSkill.call(this, APP_ID);
};

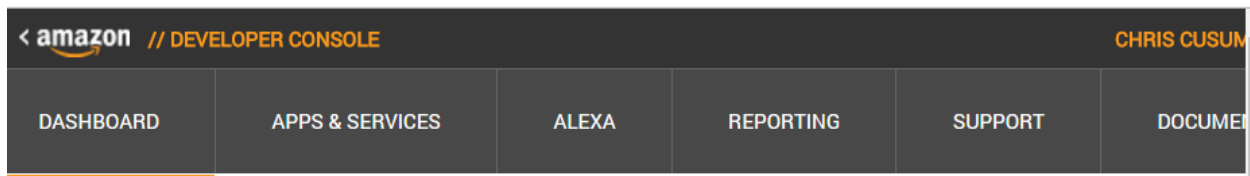
AlexaRoku.prototype = Object.create(AlexaSkill.prototype);
AlexaRoku.prototype.constructor = AlexaRoku;

function sendCommand(path,body,callback) {
  var opt = {
    host:serverinfo.host,
    port:serverinfo.port,
    path: path,
    method: 'POST',
  };
};
```

52. This is where we head to the internet: <http://developer.amazon.com>

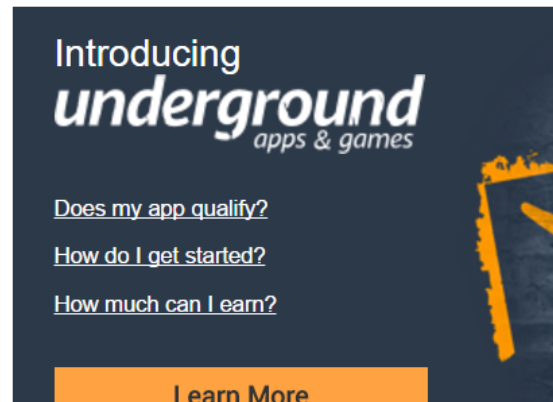
53.Top right, hit sign in and sign in

54.Similar to before, you should see Alexa at the top: click that.



Developer Communications

Announcements	Notifications
Introducing the ASK Built-in Library...	Dec 7, 2016
Announcing the 12 Sponsored Team...	Nov 14, 2016
Announcing the List Skill API	Oct 13, 2016
New Hands-Free Capabilities from t...	Oct 6, 2016
Introducing the Flash Briefing Skill A...	Oct 3, 2016




55. Click Get Started under Alexa Skills Kit

56.You should now see the skill you created in the beginning of this guide.

Click that:


Building Alexa Skills with the Alexa Skills Kit [Add a New Skill](#)

To learn more about building Alexa skills, see [Getting Started with the Alexa Skills Kit](#). To start building an Alexa skill for free using AWS Lambda, see [Creating an AWS Lambda Function for a Custom Skill](#). We encourage you to visit the [Alexa Developer Forum](#) to collaborate with Alexa team members and fellow Alexa developers.

Your skills		Skill Metrics				
Skill Name	Language	Skill Type	Modified	Status	Actions	
 Roku	English (U.S.)	Custom	12/26/16	Development	Edit	Delete

57.On the next page, you will see your 'Application Id'

- We need this for that index file you have open on your computer right now.
- Highlight it and copy (as circled in green below – some blocked out for privacy):

English (U.S.)  Add New Language

Skill Type
Define a custom interaction model or use one of the predefined skill APIs. [Learn more](#) Custom

Language
Language of your skill English (U.S.)

Application Id
The ID for this skill amzn1.ask.skill.fa7a7e19-~~XXXXXXXXXXXX~~-a08ed5-XXXX

Name
Name of the skill that is displayed to customers in the Alexa app. Must be between 2-50 characters. Roku

Invocation Name
The name customers use to activate the skill. For example, "Alexa ask Tide Pooler..." [Invocation Name Guidelines](#) roku

58. Head back to that index file in Notepad

59. In the very first line, next to `var APP_ID = "`, you need to paste your Application ID

- Make sure to leave quotation marks – but also, no spaces before or after the beginning 'a' and the last number.
- Example below:

index - Notepad

File Edit Format View Help

```
var APP_ID = "amzn1.ask.skill.fa7a2893729372.32432293712khjf"; //replace this with your app ID to make use of APP_ID verification - leave quotation marks in
var AlexaSkill = require("../AlexaSkill");
var serverinfo = require("../serverinfo");
var http = require("http");

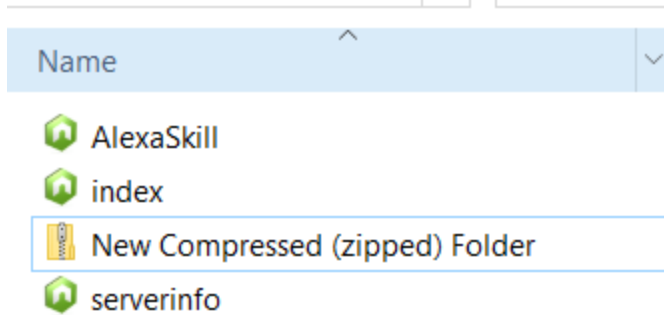
if (serverinfo.host == "127.0.0.1") {
```

60. We are so closed to being down!

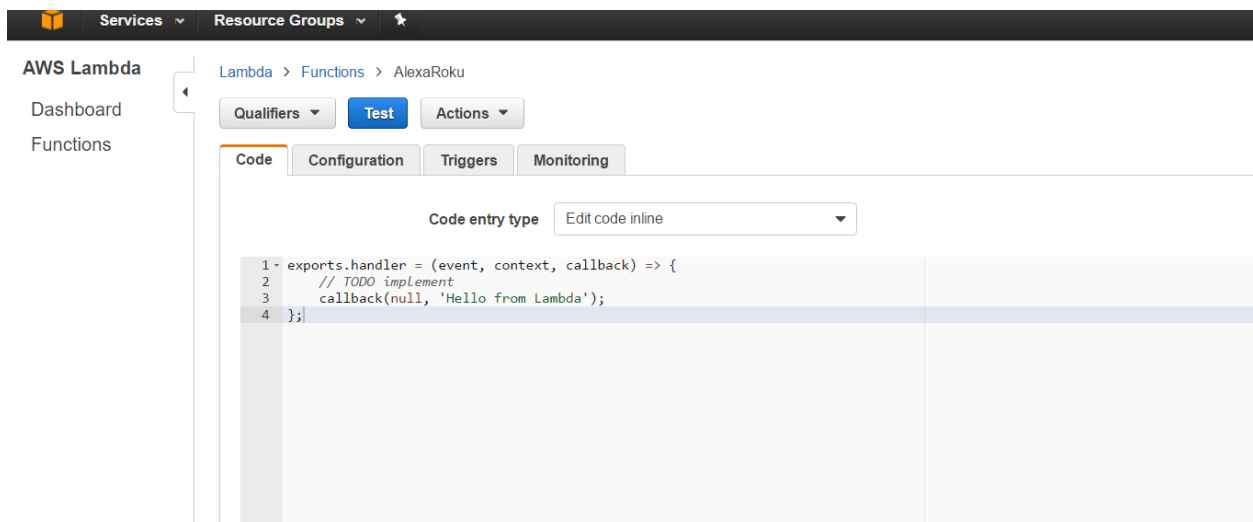
61. We now have all the files correctly edited so they are personalized to you.

62. In that same 'AWS Files – Lambda' folder

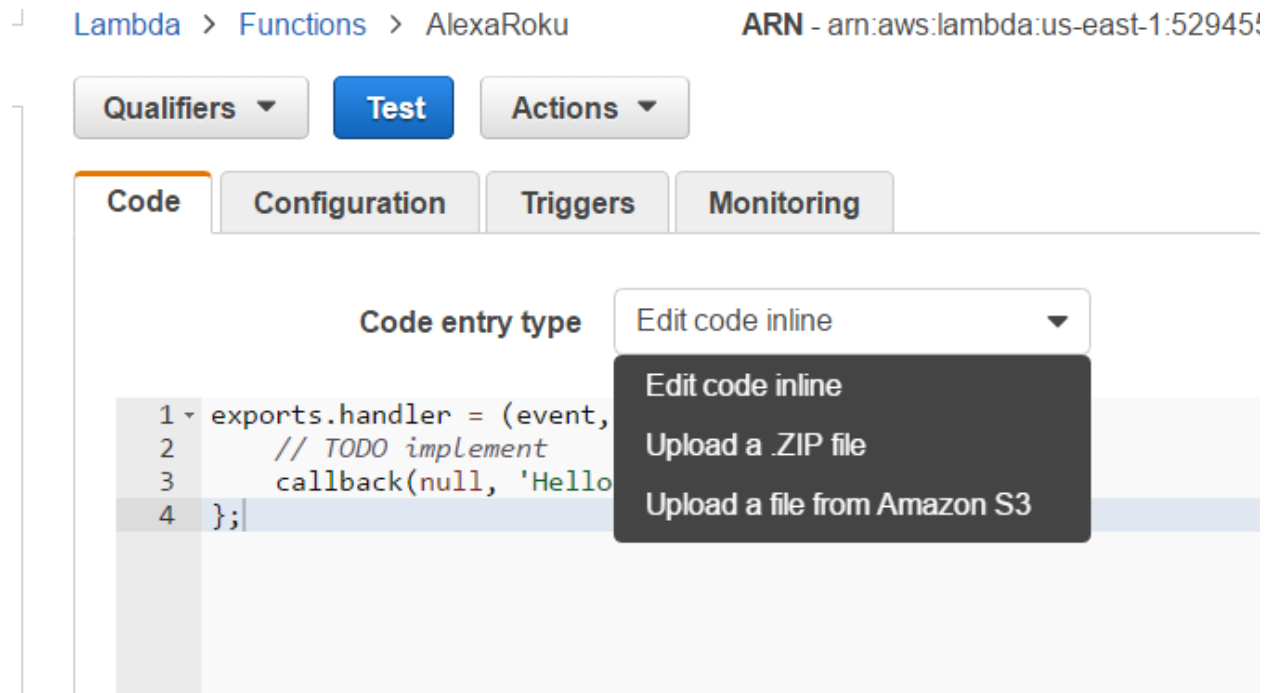
- Right click anywhere in that folder except for the files
- Hit 'New'
- Selected 'Compressed zip folder'
- A new folder-like (zip file) will appear



- e. Move all 3 of the files into that new zip file (Serverinfo, Index, AlexaSkill)
 - i. This will only copy them into it – not actually move them
 - f. Now click that file (New compressed (zipped) folder) to verify they are showing inside
63. If all 3 are in that zip file, we are now ready to upload online and hit the final 'run'!
64. Head to <https://aws.amazon.com> (same place as our Lambda server)
- a. Top right, 'sign into console'
 - b. Search for Lambda and select
 - c. You should see the skill we created, select it (all like before)
 - d. You will now be taken to their code page:



65. Under Code Entry type (drop down list right above the code box) – select it and choose 'upload a .ZIP file)



66.Hit the upload button and go to the zip file we just created (in the AWS – Lambda Server folder)!


67.Now hit Save at the top (Don't hit Save and test)

Lambda > Functions > AlexaRoku ARN - arn:aws:lam i:AlexaRoku

Qualifiers ▾ Save Save and test Actions ▾

Code Configuration Triggers Monitoring ?

Code entry type Upload a .ZIP file ▾

Function package*  Upload New Compressed (zipped) Folder.zip

For files larger than 10 MB, consider uploading via S3.

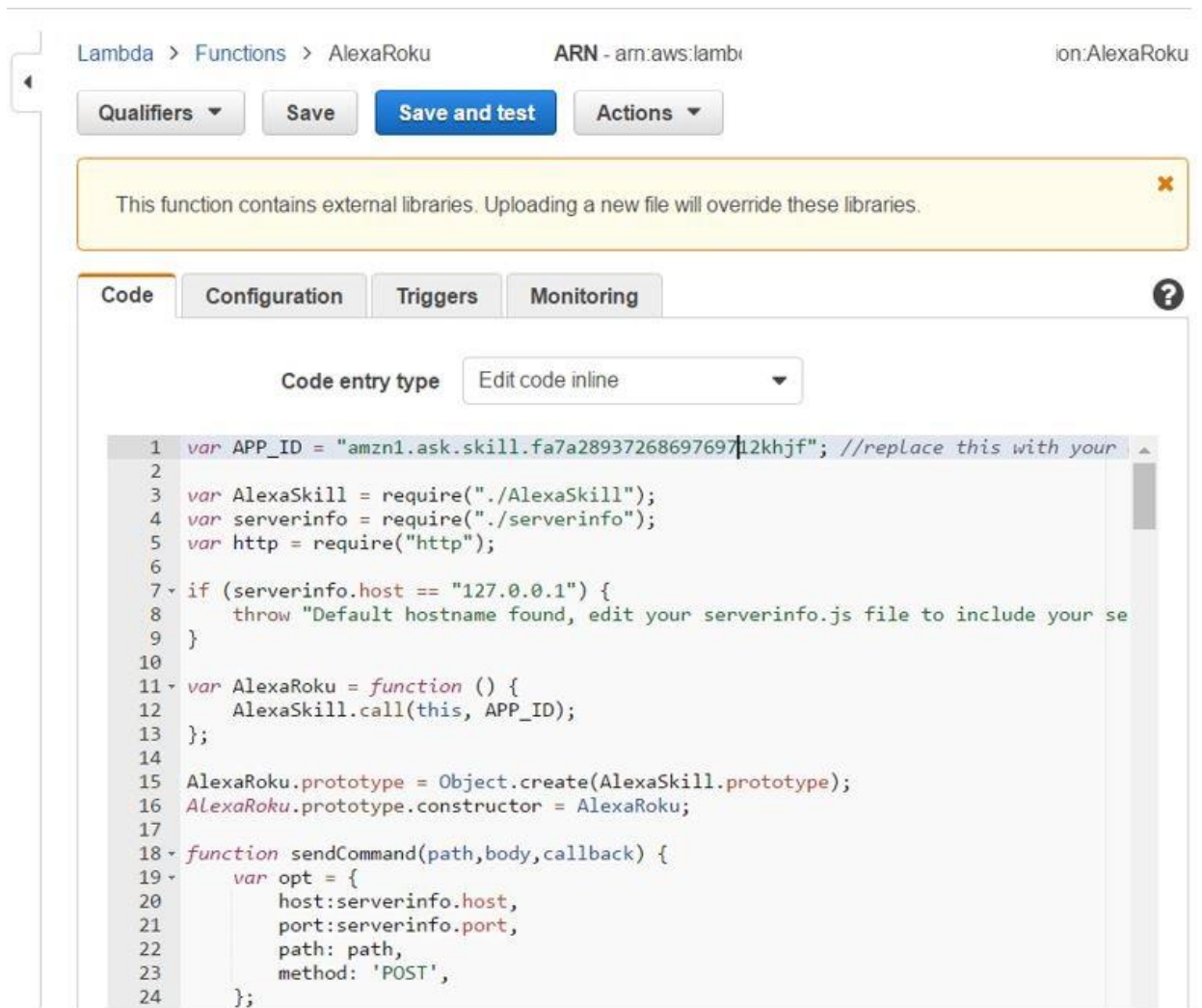
You can define Environment Variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. [Learn more](#). For storing sensitive information, we recommend encrypting values using KMS and the console's encryption helpers.

Enable encryption helpers ☐

Environment variables

Key	Value	✕
-----	-------	---

68. Your code should now appear like this:



69.Finally, we are ready for the last step!

70.To be safe and ensure no other command prompt is running at this point, close all your programs.

71.Once this is done, head to the start menu, bottom right, and search 'node'

72.Choose Node.js Command Prompt

73.Now type: node server.js

74.If all is well, you will still get this:

```
node server.js
Your environment has been set up for using Node.js 7.3.0 (x64) and npm.
C:\Users\Harry Potter>node server.js
Server listening on: http://localhost:8060
```

75. Now go test it.... 'Alexa, tell roku to go home'

76.If Alexa says 'going home' and the Roku takes you home....Congrats, it is done 😊

77.In addition, that command prompt file should show you the step it did:

```
node server.js
Your environment has been set up for using Node.js 7.3.0 (x64) and npm.
C:\Users\Harry Potter>node server.js
Server listening on: http://localhost:8060
Found Roku: http://192.168.0.108:8060/
Posting: http://192.168.0.108:168 /keypress/home
```

Now enjoy....

A few caveats I have discovered....

1. If you make a command and the Roku is turned off or disconnected, the node server gives an error and disconnects. You MUST open back up and type: node server.js (and hit enter) to get it going again. Will check a way to make that auto-reconnect....but for now.
2. If you want more than one Roku....you will need to do this method for each Roku, using a different 'Invocation name' (step 20 for reference) to call it.
 - a. I have not tested this, but don't see why it shouldn't work.
3. If you leave the 'server' file IP address line as 'null' (in your windows 'users' folder – steps 44-45) and you have more than one Roku connected, it likely will try to control them all or error out (have not tested).
4. If you get an error or something isn't working in these steps
 - a. It is likely one of the reasons:
 - i. The ARN ID wasn't correctly copied over to the Alexa Skill Developer website
 - ii. The App ID wasn't correctly copied into the server file
 - iii. Your IP address isn't correctly forwarded

- iv. The IP address or port number used in the Server or serverinfo file is not correctly inputted.

Remember, you MUST always say 'Alexa, Tell Roku.....' for it to begin the skill.

If you want to know the commands to invoke a specific function, reopen that 'Sample Utterances' file we opened earlier. The function names are mostly self-explanatory – so just see what the examples are right next to them to know how to get a specific function: