Help for Installing HackerMode 2.0

Contents

Necessary Accounts and Hosting Info

- An Amazon Developer account. This is required to create and configure Alexa skills.
- An Amazon Web Services (AWS) account. This is required for hosting a skill on AWS Lambda
- A PHP hosting account pointed at a domain that you own. GoDaddy or a million other hosting accounts provide PHP in a very basic hosting site. You can even set one up on your Amazon account which would be a nice touch. You can use AWS LightSail which gives you cheap PHP hosting slightly more than half the cost of GoDaddy.
- A domain name for your site and TLS binding.
- A TLS certificate because Alexa won't talk to a http site. Prior to getting a certificate you need to have a domain pointing at your PHP site for your certificate to be associated with.

  GoDaddy will allow you to purchase a certificate, or give you complex instructions on how to install Let's Encrypt, but don't expect them to make it too easy for you to get a free cert. The instructions I've seen are hard to follow and sometimes are out-dated. If you don't use Let's Encrypt a cert may set you back $70+ USD or somewhere around that price. If you use Amazon LightSail which is an excellent choice you will need to:

  1. create your LightSail LAMP instance,
  2. create a load balancer, https://lightsail.aws.amazon.com/ls/docs/en_us/articles/verify-tls-ssl-certificate-using-dns-cname-https
     a. create a certificate,
     b. verify domain ownership,
     c. attach the resulting certificate to load balancer
- You will also need a Kali instance or basis for now. This is untested on other platforms.

NOTE: (Feel free to contribute to a queue replacement to add extra security and reduce the overhead of owning HackerMode 2.)

Once you have Kali, these three accounts (Alexa, AWS, PHP hosting) and your domain name pointed at your PHP site, and a certificate for the domain, you can begin the process of actually installing the parts of HackerMode 2.0!

Step One of Installation (PHP setup)

1. In the HTML root or sub directory of your PHP hosting site, drop the PHP files.
2. Modify their names to some name that ONLY you are aware of. (Unless you want other people who are aware of the project to hit URLs on your site and send random crap in) Note the filenames you changed them to, as you will need to tell Alexa and Kali later where to find the queue files.
3. Each of the PHP files includes a relative path to other PHP files. Ensure the PHP code reflects the new file names you picked in step 2.
4. Log files are created at run-time. Ensure the PHP code is updated to reflect the new names of log files and the new names of log files being written to. It would be wise to change the names of log files so that they are not publicly known. These log file locations are included in the PHP files (Kali_Read.php, Kali2Alexa.php, Alexa_Read.php, and Alexa2Kali.php)
5. You can further secure the queue PHP files by IP address whitelisting.

Step Two of Installation (Securing PHP files)

Apply whatever security settings and headers to your PHP files as you feel appropriate. The PHP queue does some basic things for security but web hygiene is needed to stay safe ultimately. Keep your queue file names secret so as to not let others write to your queue. You may want to white-list the IP of the Kali box so that only it and AWS resources can call the queue. Future versions will support authentication and signing, but that is not in the current release. You will typically want your Apache server serving pages with the HTTP headers: ***X-Content-Type-Options: nosniff*** and ***X-XSS-Protection: 1; mode=block*** headers turned on in httpd.conf or by adjusting the PHP files directly. You will probably also want the header ***X-Frame-Options DENY*** header turned on. Follow this guide for how to set these headers in the httpd.conf file of Apache. https://geekflare.com/http-header-implementation/ There are other places to set headers in PHP and I'm not a hosting expert so you may need to do some consulting of the Google…

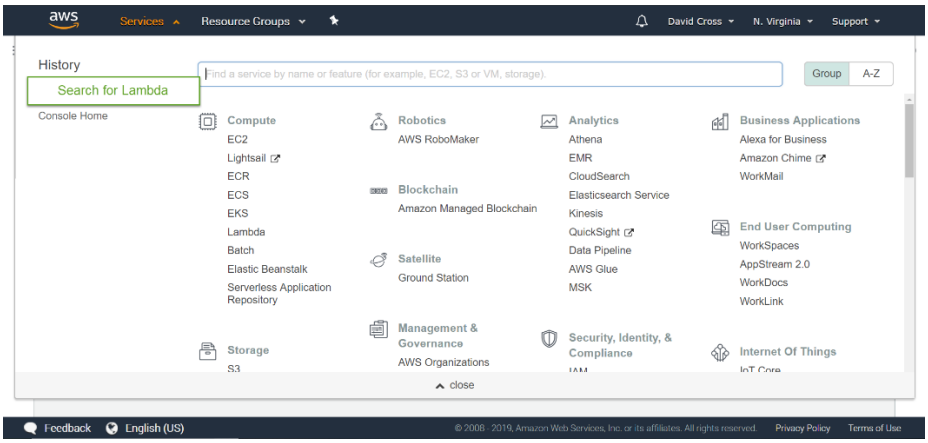
Step Three of Installation (Setting up music files)

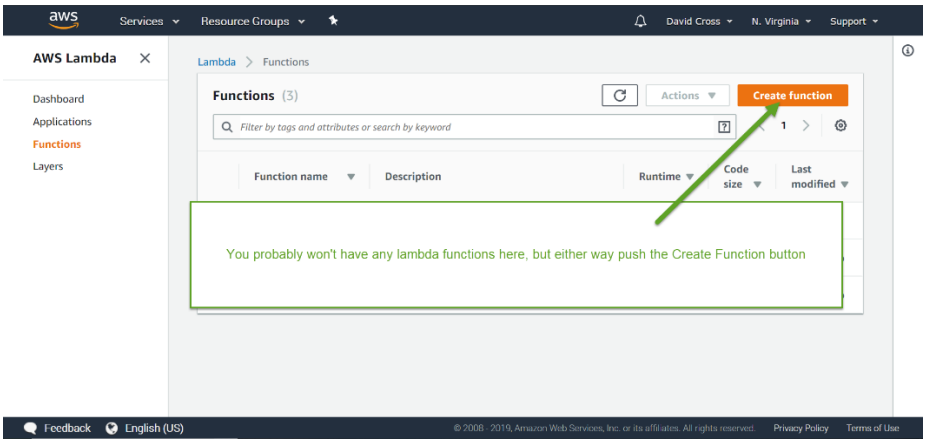
Create a directory on your PHP site and put the .mp3 files in it. Note the URL to the MP3 files as you will need it later when configuring your lambda function. Both of the songs I found were royalty free and don't require attribution, but the Rick Roll version is by CutieSMG. You may want to replace the default songs with your own music anyway. Audacity and the LAME encoder will help you create Alexa-style MP3s. Remember that when you cut MP3s for Alexa that you need to encode them with FFMPEG or using the site that comes up on Google that uses the FFMPEG encoder for you. Music played by Alexa in this format has to be less than 90 seconds or it won't play. There won't even be an error, they just won't work and you'll be scratching your head for days trying to figure that one out.

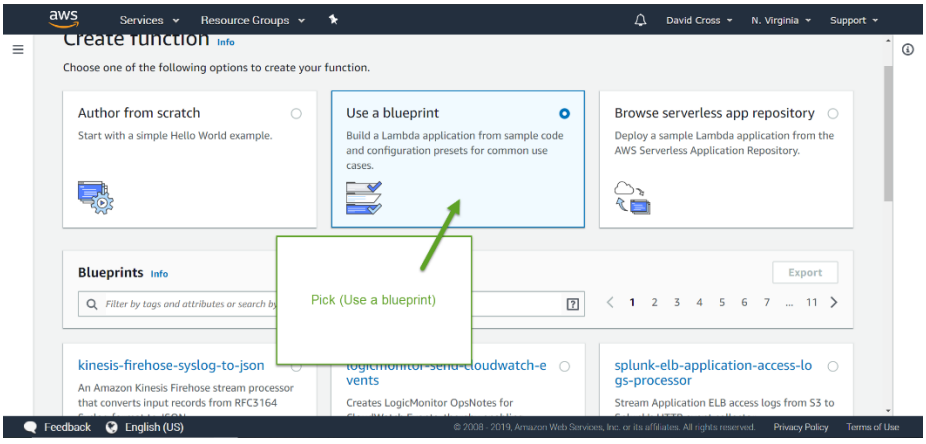
Step Four of Installation (Creating your Lambda)

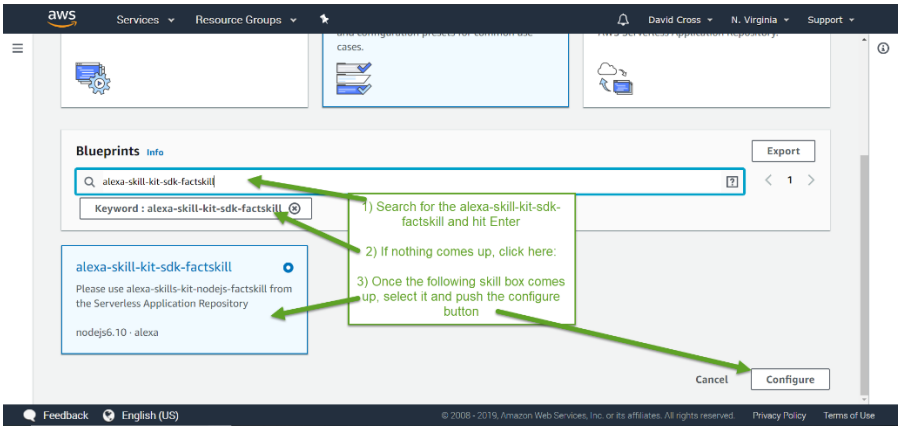
Step Four phase 1

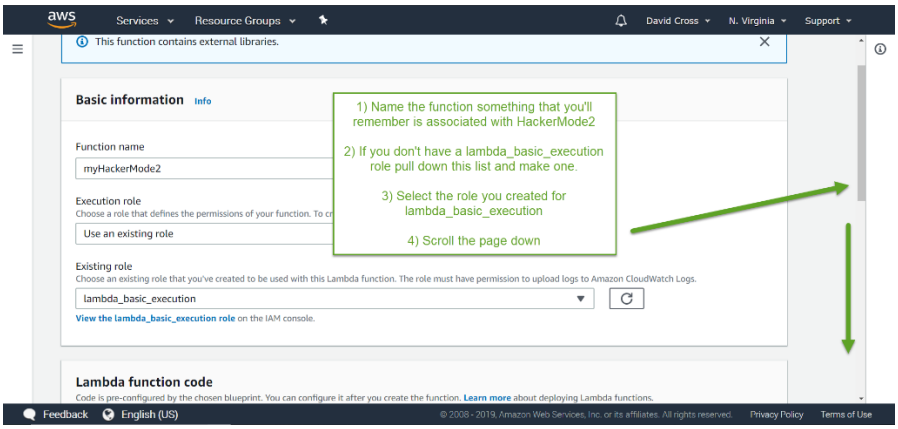
Create the Lambda function. Ensure you are logged into AWS and then search for Lambda.

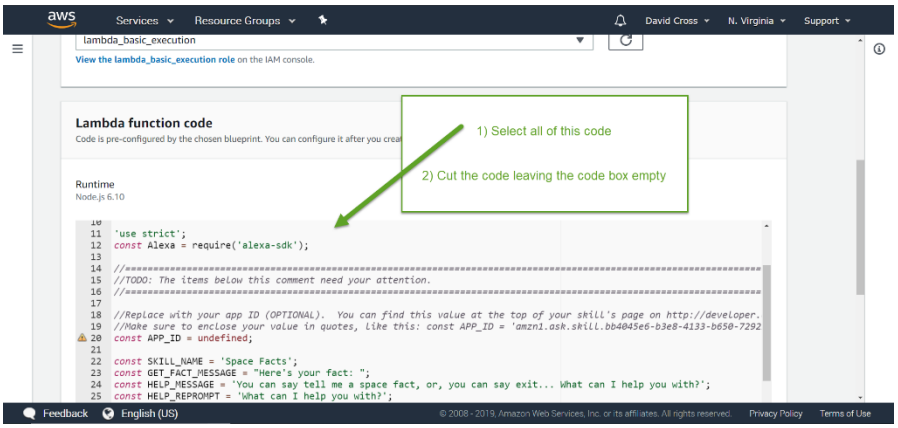


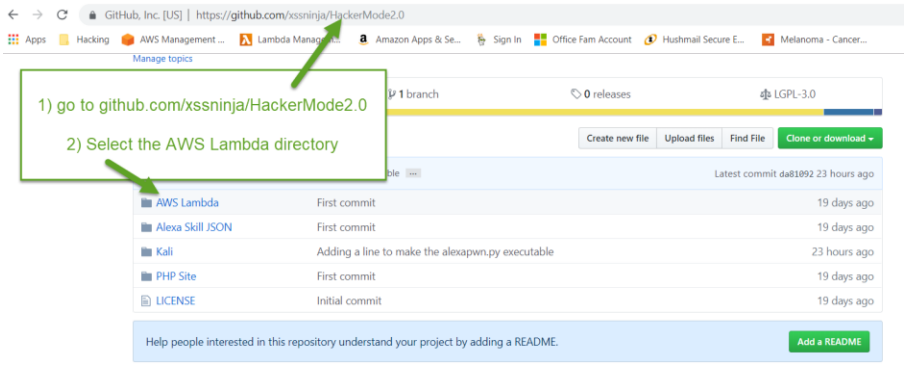


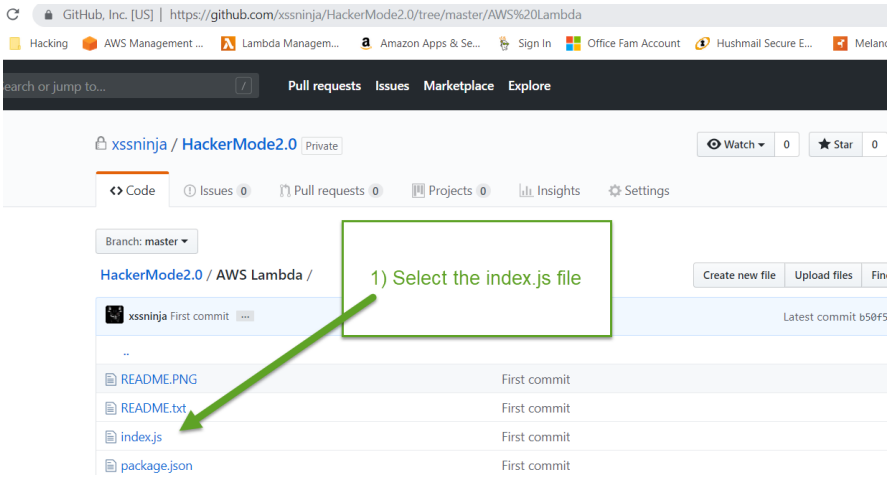



Services
Resource Groups
David Cross
N. Virginia
Support
Blueprints
Info
Export
alexa-skill-kit-sdk-factskill
Keyword : alexa-skill-kit-sdk-factskill
1) Search for the alexa-skill-kit-sdk-factskill and hit Enter
2) If nothing comes up, click here:
3) Once the following skill box comes up, select it and push the configure button
alexa-skill-kit-sdk-factskill
Please use alexa-skills-kit-nodejs-factskill from the Serverless Application Repository
nodejs6.10 · alexa
Cancel
Configure
Feedback
English (US)

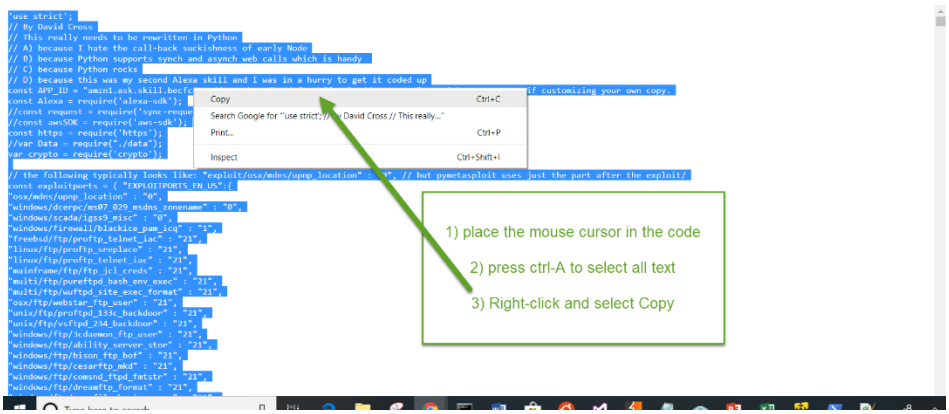

Services
Resource Groups
David Cross
N. Virginia
Support
This function contains external libraries.
Basic information
Info
1) Name the function something that you'll remember is associated with HackerMode2
2) If you don't have a lambda_basic_execution role pull down this list and make one.
3) Select the role you created for lambda_basic_execution
4) Scroll the page down
Function name
myHackerMode2
Execution role
Use an existing role
Existing role
lambda_basic_execution
View the lambda_basic_execution role on the IAM console.
Lambda function code
Feedback
English (US)

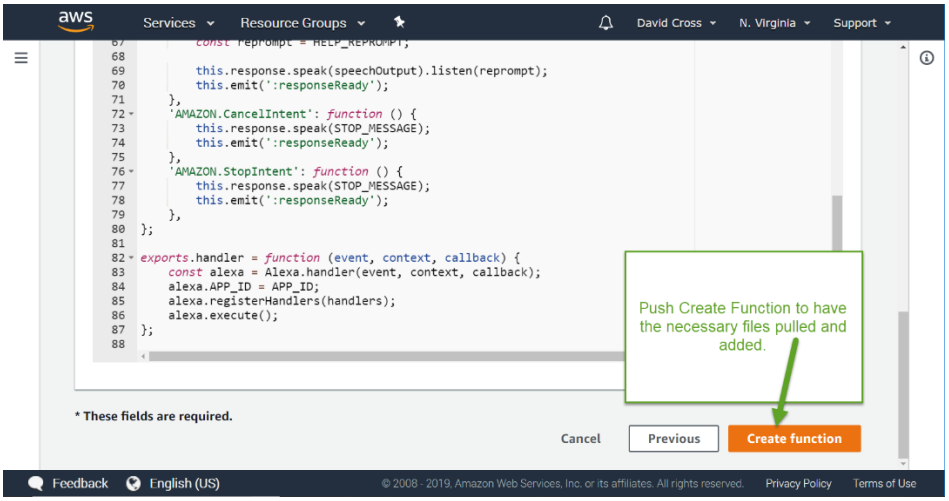

Services
Resource Groups
David Cross
N. Virginia
Support
lambda_basic_execution
View the lambda_basic_execution role on the IAM console.
Lambda function code
1) Select all of this code
2) Cut the code leaving the code box empty
Runtime
Node.js 6.10
'use strict';
const Alexa = require('alexa-sdk');
//TODO: The items below this comment need your attention.
//Replace with your app ID (OPTIONAL). You can find this value at the top of your skill's page on http://developer.
//Make sure to enclose your value in quotes, like this: const APP_ID = 'amzn1.ask.skill.bb4045e6-b3e8-4133-b650-7292
const APP_ID = undefined;
const SKILL_NAME = 'Space Facts';
const GET_FACT_MESSAGE = "Here's your fact: ";
const HELP_MESSAGE = 'You can say tell me a space fact, or, you can say exit... What can I help you with?';
const HELP_REPROMPT = 'What can I help you with?';
Feedback
English (US)

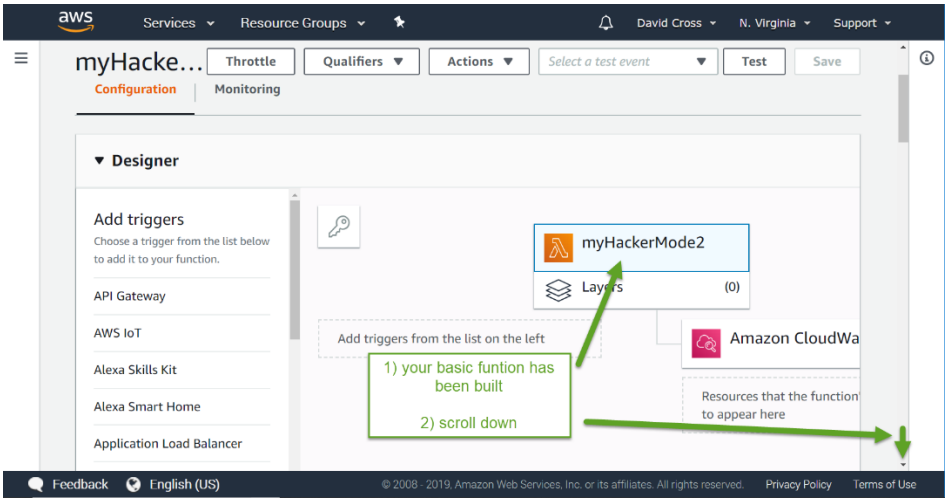

1) go to github.com/xssninja/HackerMode2.0
2) Select the AWS Lambda directory
AWS Lambda
Alexa Skill JSON
Kali
PHP Site
LICENSE

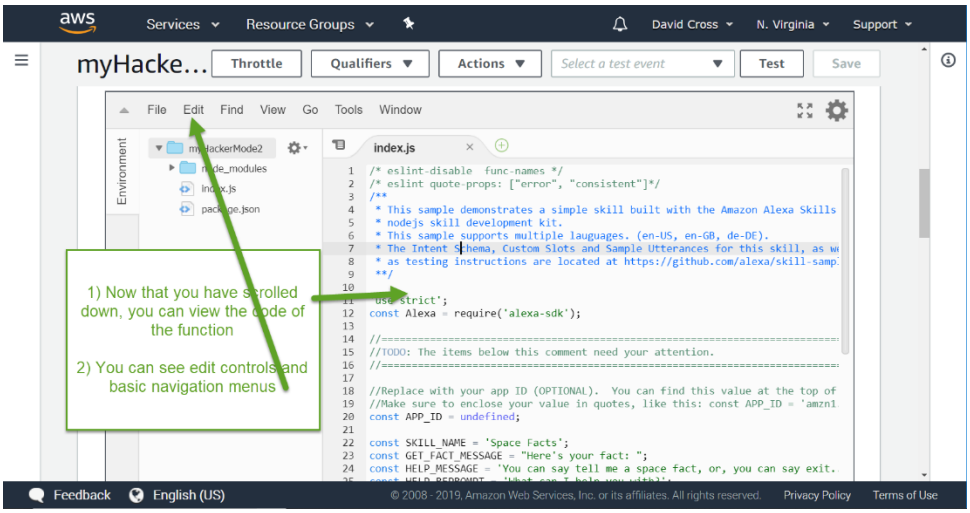

xssninja / HackerMode2.0
1) Select the index.js file
README.PNG
README.txt
index.js
package.json

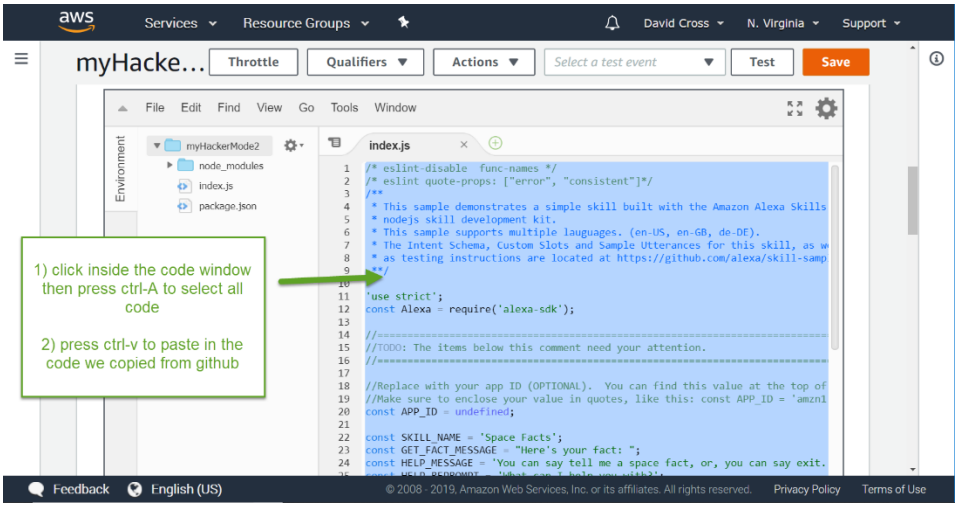

Click Raw

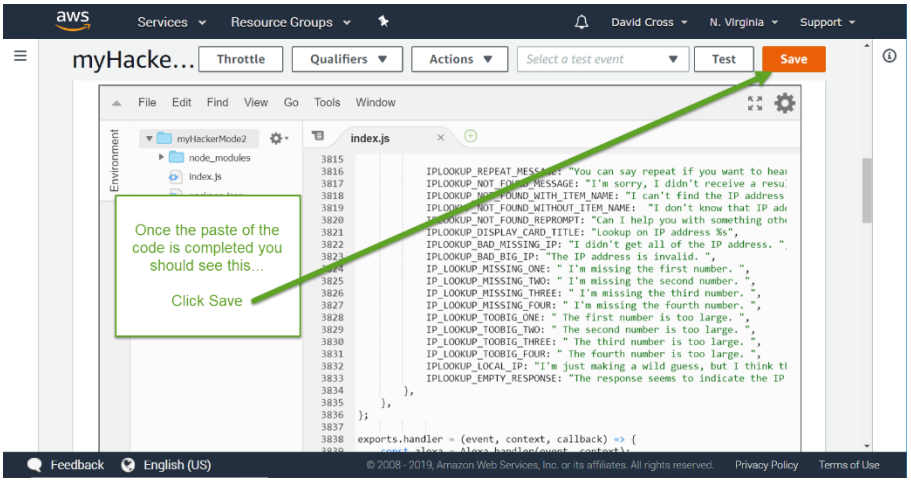

1) place the mouse cursor in the code
2) press ctrl-A to select all text
3) Right-click and select Copy


Push Create Function to have the necessary files pulled and added.
* These fields are required.
Cancel
Previous
Create function


myHacke...
Throttle
Qualifiers
Actions
Select a test event
Test
Save
Configuration
Monitoring
Designer
Add triggers
Choose a trigger from the list below to add it to your function.
API Gateway
AWS IoT
Alexa Skills Kit
Alexa Smart Home
Application Load Balancer
myHackerMode2
Layers
(0)
Add triggers from the list on the left
Amazon CloudWa
1) your basic funtion has been built
2) scroll down
Resources that the function to appear here

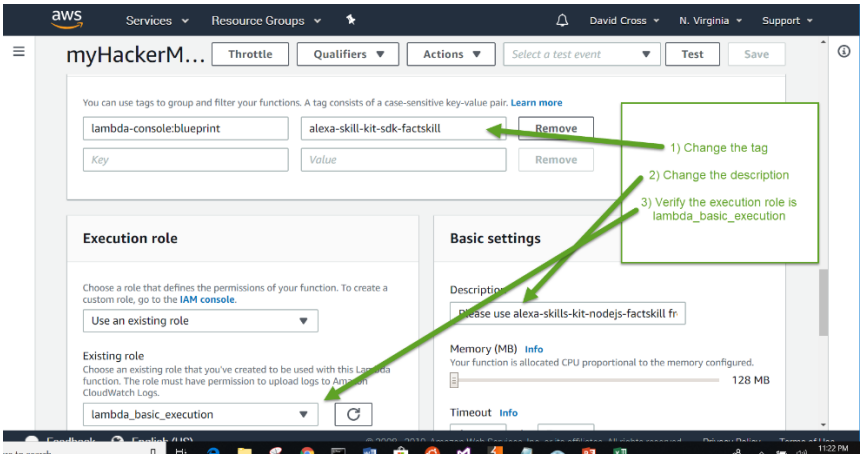

1) Now that you have scrolled down, you can view the code of the function
2) You can see edit controls and basic navigation menus

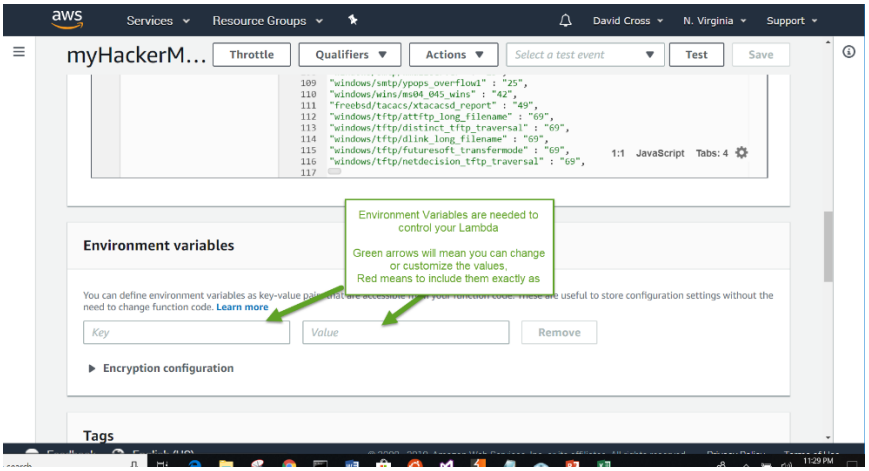

1) click inside the code window then press ctrl-A to select all code
2) press ctrl-v to paste in the code we copied from github

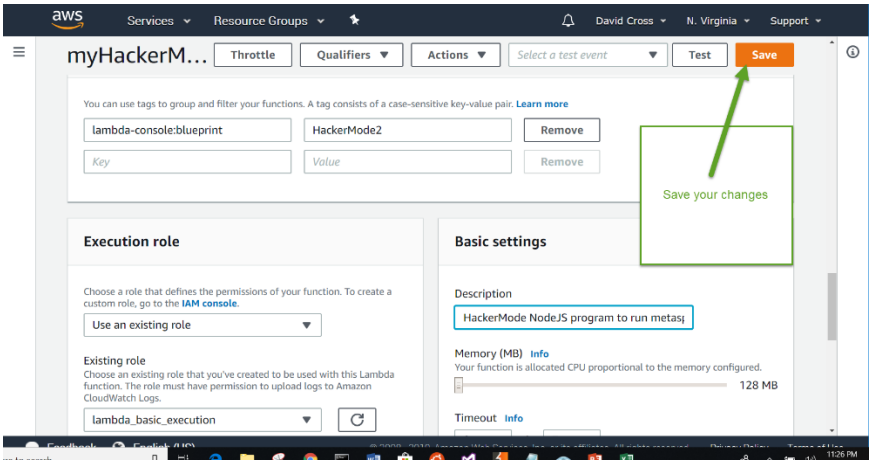

Once the paste of the code is completed you should see this...
Click Save

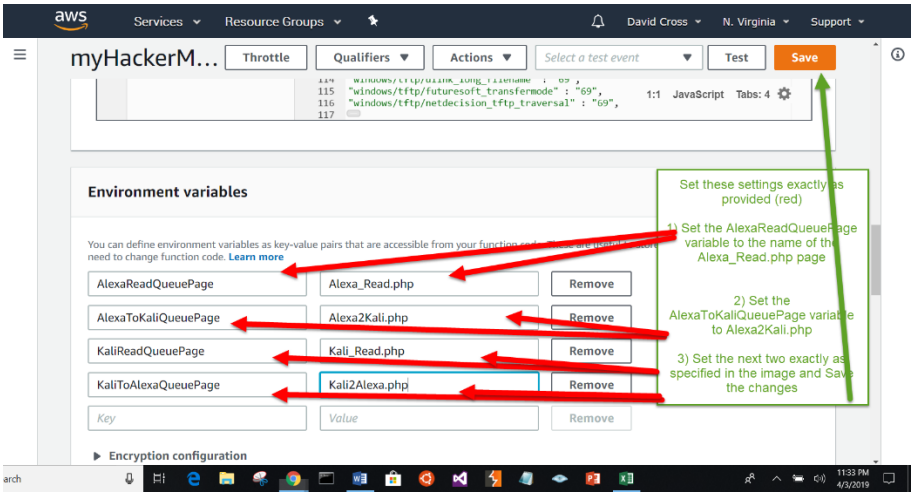
Step Four phase 2 (Configure your Lambda)

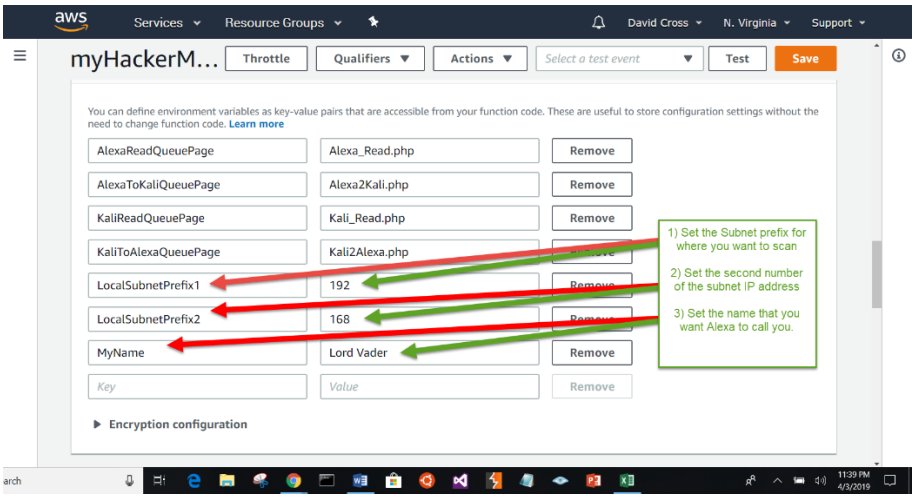
Configure the Lambda function.

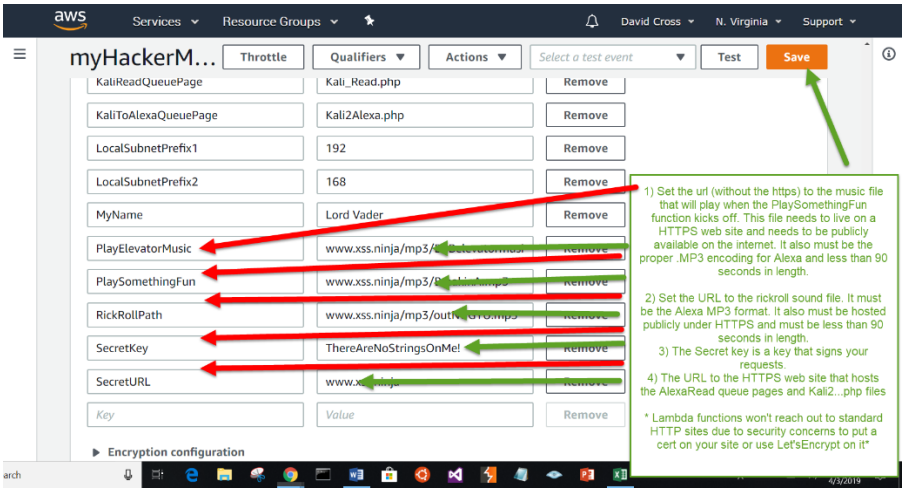




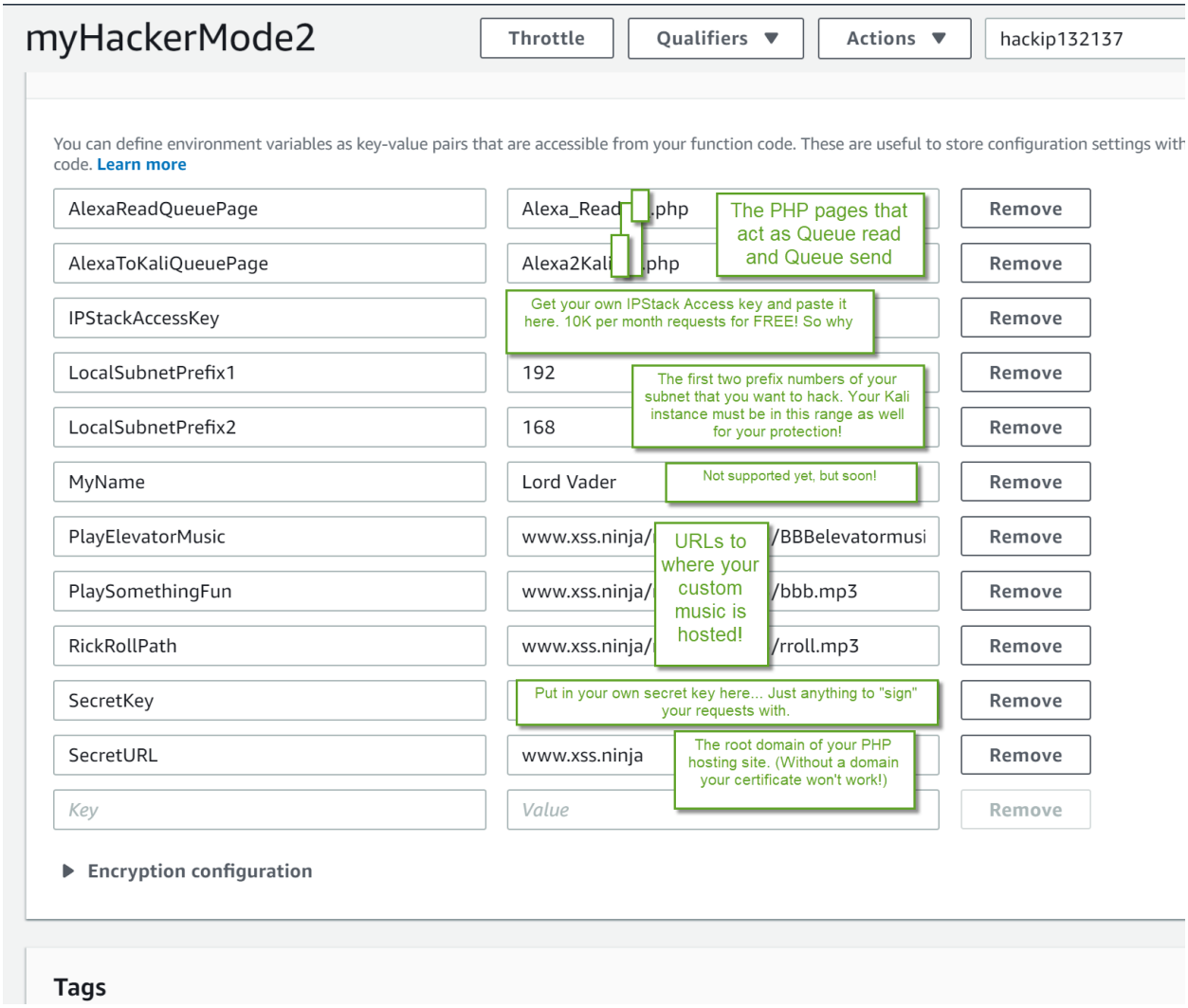



Services
Resource Groups
David Cross
N. Virginia
Support
myHackerM...
Throttle
Qualifiers
Actions
Select a test event
Test
Save
"windows/tftp/futuresoft_transfermode" : "69",
"windows/tftp/netdecision_tftp_traversal" : "69",
1:1 JavaScript Tabs: 4
Environment variables
You can define environment variables 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
AlexaReadQueuePage
Alexa_Read.php
AlexaToKaliQueuePage
Alexa2Kali.php
KaliReadQueuePage
Kali_Read.php
KaliToAlexaQueuePage
Kali2Alexa.php
Key
Value
Remove
Encryption configuration
Set these settings exactly as provided (red)
1) Set the AlexaReadQueuePage variable to the name of the Alexa_Read.php page
2) Set the AlexaToKaliQueuePage variable to Alexa2Kali.php
3) Set the next two exactly as specified in the image and Save the changes


Services
Resource Groups
David Cross
N. Virginia
Support
myHackerM...
Throttle
Qualifiers
Actions
Select a test event
Test
Save
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
AlexaReadQueuePage
Alexa_Read.php
AlexaToKaliQueuePage
Alexa2Kali.php
KaliReadQueuePage
Kali_Read.php
KaliToAlexaQueuePage
Kali2Alexa.php
LocalSubnetPrefix1
192
LocalSubnetPrefix2
168
MyName
Lord Vader
Key
Value
Remove
Encryption configuration
1) Set the Subnet prefix for where you want to scan
2) Set the second number of the subnet IP address
3) Set the name that you want Alexa to call you.


Services
Resource Groups
David Cross
N. Virginia
Support
myHackerM...
Throttle
Qualifiers
Actions
Select a test event
Test
Save
KaliReadQueuePage
Kali_Read.php
KaliToAlexaQueuePage
Kali2Alexa.php
LocalSubnetPrefix1
192
LocalSubnetPrefix2
168
MyName
Lord Vader
PlayElevatorMusic
PlaySomethingFun
RickRollPath
SecretKey
ThereAreNoStringsOnMe!
SecretURL
Key
Value
Remove
Encryption configuration
1) Set the url (without the https) to the music file that will play when the PlaySomethingFun function kicks off. This file needs to live on a HTTPS web site and needs to be publicly available on the internet. It also must be the proper .MP3 encoding for Alexa and less than 90 seconds in length.
2) Set the URL to the rickroll sound file. It must be the Alexa MP3 format. It also must be hosted publicly under HTTPS and must be less than 90 seconds in length.
3) The Secret key is a key that signs your requests.
4) The URL to the HTTPS web site that hosts the AlexaRead queue pages and Kali2...php files
* Lambda functions won't reach out to standard HTTP sites due to security concerns to put a cert on your site or use Let'sEncrypt on it*

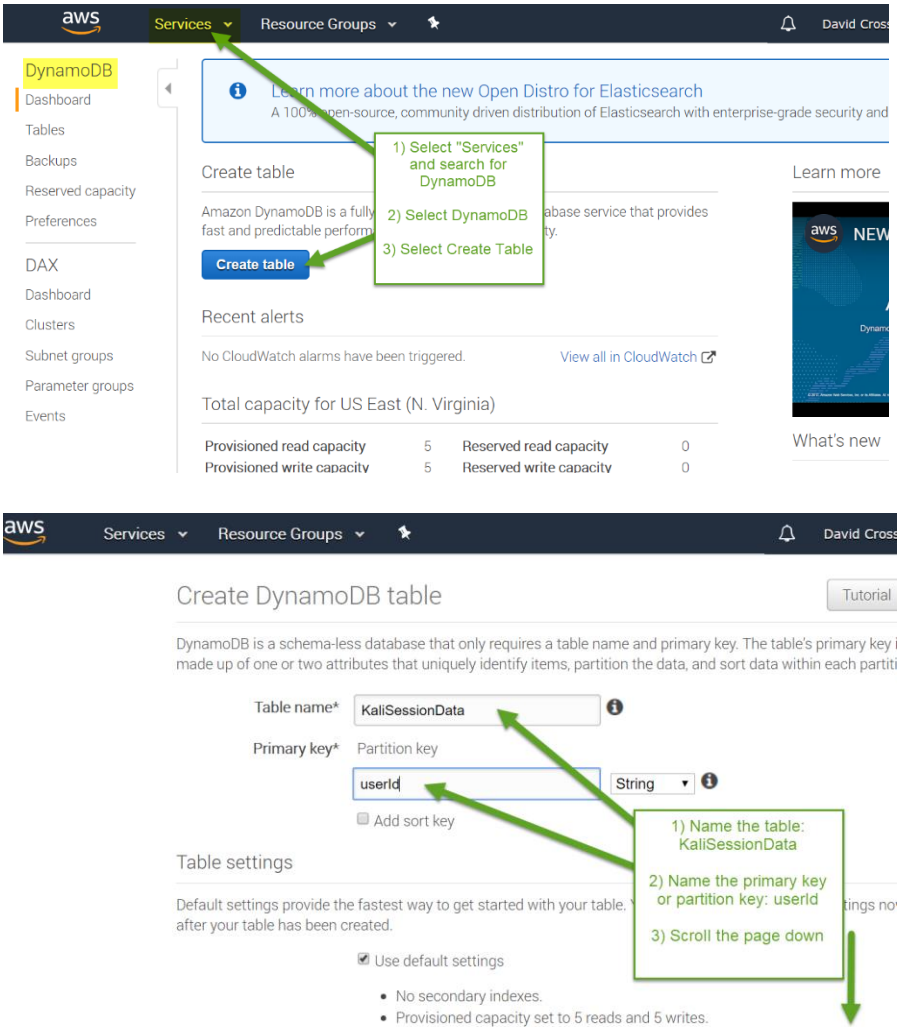
When you're all done it should look something like this minus the green boxes! (The green boxes indicate information that is sensitive!

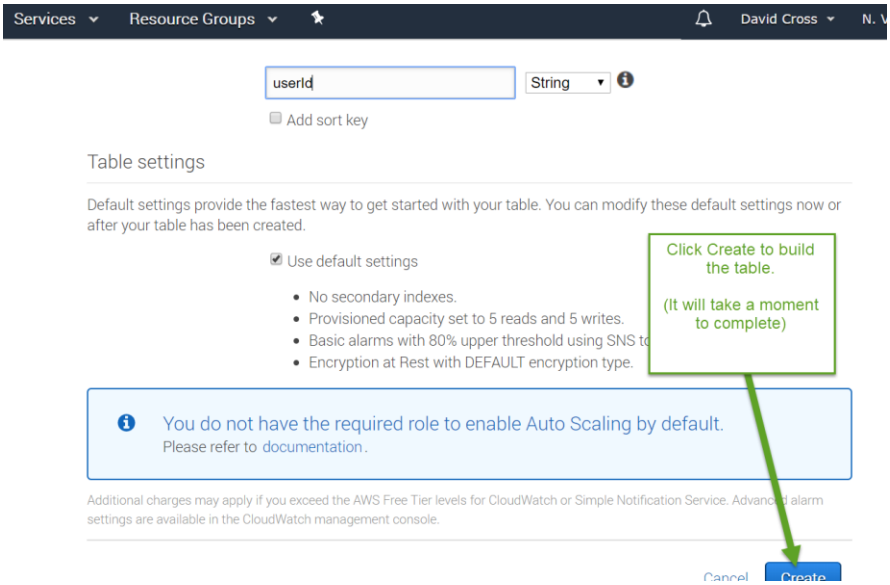


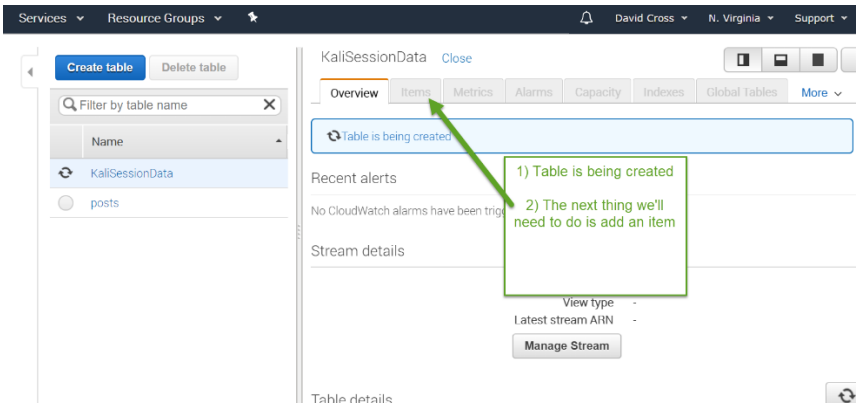
In the image above, note that you probably want your queue pages to be named something unique to your install. (Security by obscurity!) And if you want to use the IP Lookup ability, you'll want to register at https://ipstack.com/quickstart and get an IPStack Access Key. You will enter that secret value in the environment variable as shown above… (Don't post it anywhere or they'll block you!)

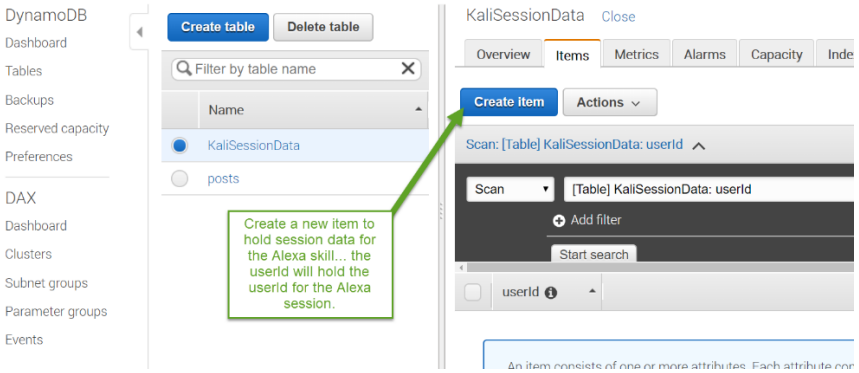
Step Four Phase 3 (Creating Dynamo DB)

Create Dynamo DB and rights for session preservation.

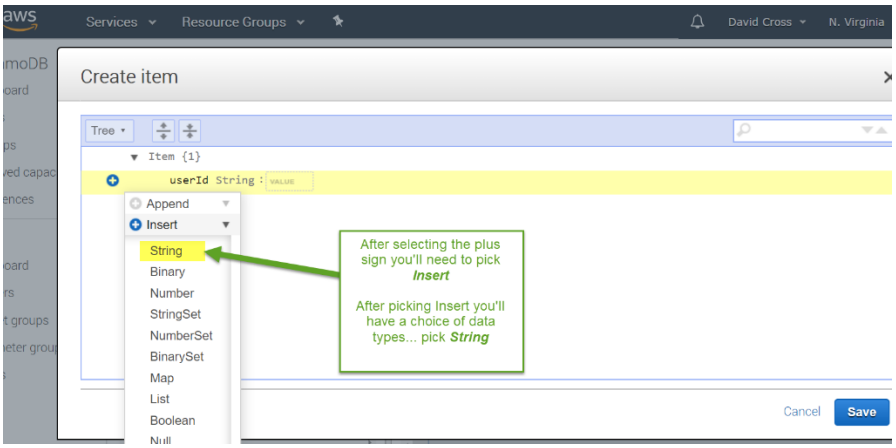


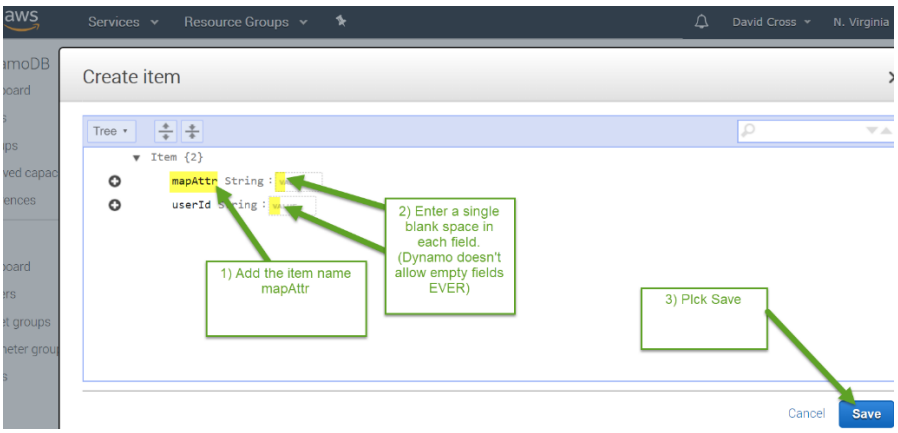

Services
Resource Groups
David Cross
userId
String
Add sort key
Table settings
Default settings provide the fastest way to get started with your table. You can modify these default settings now or after your table has been created.
Use default settings
No secondary indexes.
Provisioned capacity set to 5 reads and 5 writes.
Basic alarms with 80% upper threshold using SNS t
Encryption at Rest with DEFAULT encryption type.
Click Create to build the table.
(It will take a moment to complete)
You do not have the required role to enable Auto Scaling by default.
Please refer to documentation.
Additional charges may apply if you exceed the AWS Free Tier levels for CloudWatch or Simple Notification Service. Advanced alarm settings are available in the CloudWatch management console.
Cancel
Create

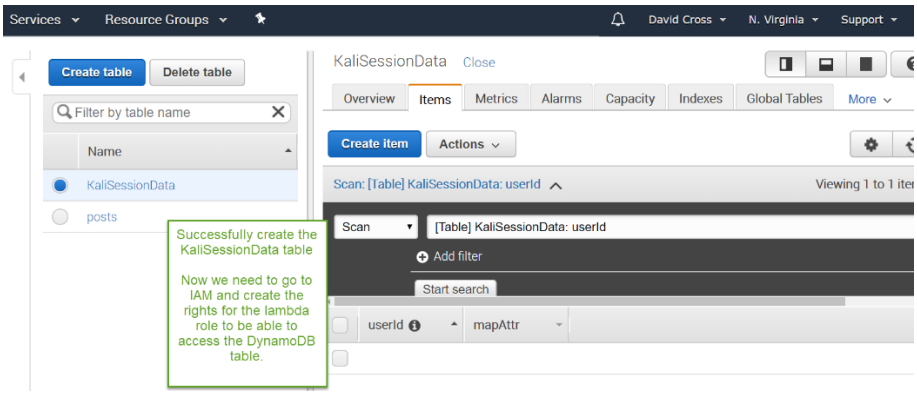

Services
Resource Groups
David Cross
N. Virginia
Support
Create table
Delete table
Filter by table name
Name
KaliSessionData
posts
KaliSessionData
Close
Overview
Items
Metrics
Alarms
Capacity
Indexes
Global Tables
More
Table is being created
Recent alerts
No CloudWatch alarms have been trig
1) Table is being created
2) The next thing we'll need to do is add an item
Stream details
View type
Latest stream ARN
Manage Stream
Table details

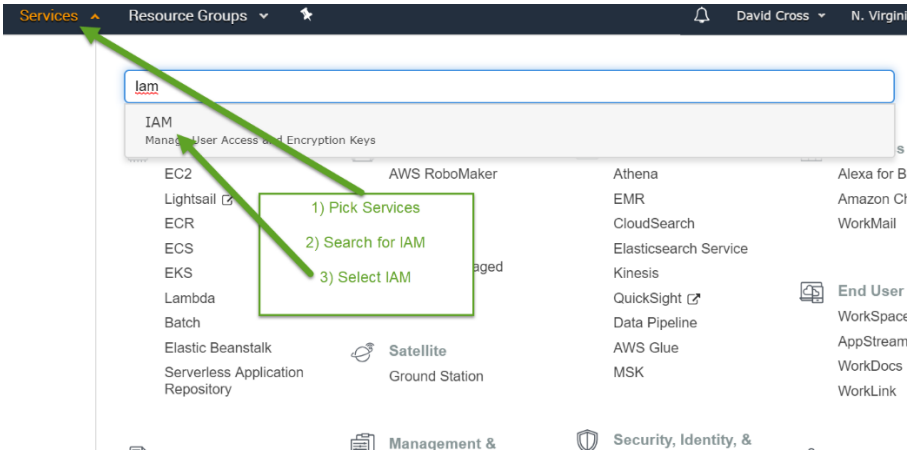

DynamoDB
Dashboard
Tables
Backups
Reserved capacity
Preferences
DAX
Dashboard
Clusters
Subnet groups
Parameter groups
Events
Create table
Delete table
Filter by table name
Name
KaliSessionData
posts
Create a new item to hold session data for the Alexa skill... the userId will hold the userId for the Alexa session.
KaliSessionData
Close
Overview
Items
Metrics
Alarms
Capacity
Create item
Actions
Scan: [Table] KaliSessionData: userId
Scan
[Table] KaliSessionData: userId
Add filter
Start search
userId

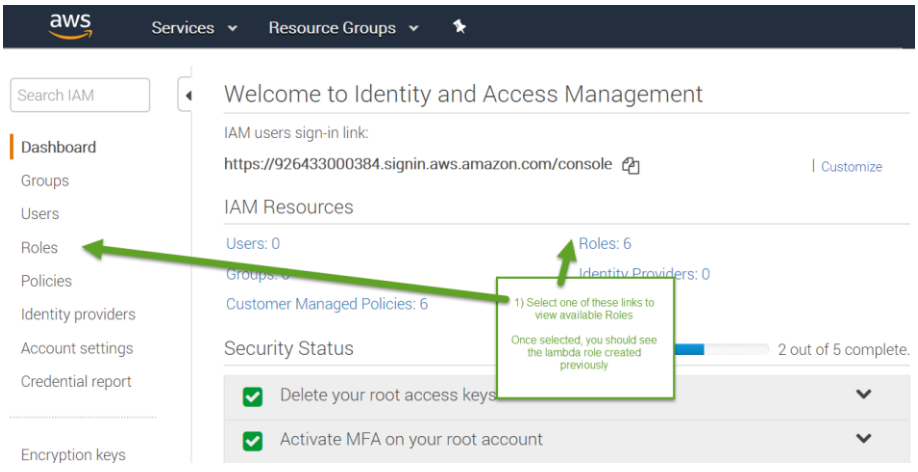

Services
Resource Groups
David Cross
N. Virginia
Create item
Tree
Item {1}
userId String :
After clicking "Create Item" we get this screen and to add another data item we need to click the plus sign here
Cancel
Save

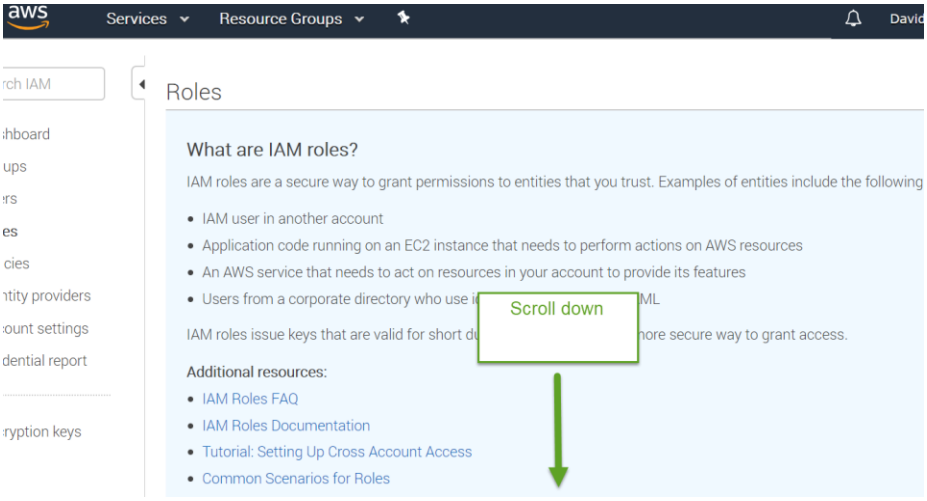

Services
Resource Groups
David Cross
N. Virginia
Create item
Tree
Item {1}
userId String :
Append
Insert
String
Binary
Number
StringSet
NumberSet
BinarySet
Map
List
Boolean
After selecting the plus sign you'll need to pick Insert
After picking Insert you'll have a choice of data types... pick String
Cancel
Save

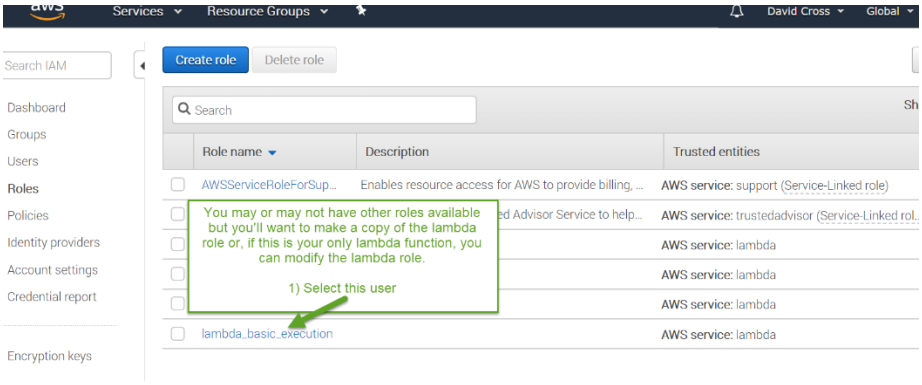

Services
Resource Groups
David Cross
N. Virginia
Create item
Tree
Item {2}
mapAttr String :
userId String :
1) Add the item name mapAttr
2) Enter a single blank space in each field. (Dynamo doesn't allow empty fields EVER)
3) Pick Save
Cancel
Save

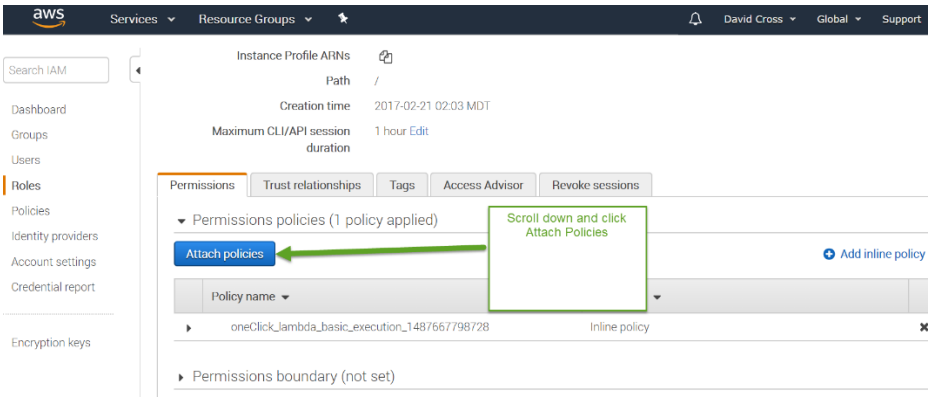

Services
Resource Groups
David Cross
N. Virginia
Support
Create table
Delete table
Filter by table name
Name
KaliSessionData
posts
KaliSessionData
Close
Overview
Items
Metrics
Alarms
Capacity
Indexes
Global Tables
More
Create item
Actions
Scan: [Table] KaliSessionData: userId
Viewing 1 to 1 ite
Scan
[Table] KaliSessionData: userId
Add filter
Start search
userId
mapAttr
Successfully create the KaliSessionData table
Now we need to go to IAM and create the rights for the lambda role to be able to access the DynamoDB table.

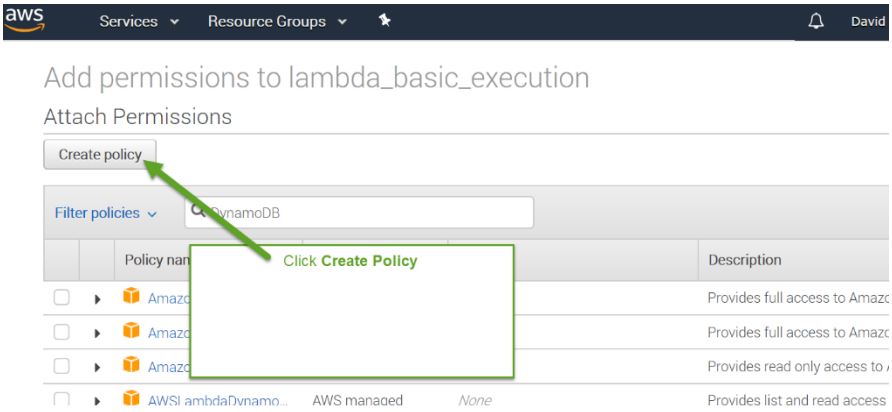

Services
Resource Groups
David Cross
N. Virgini
Iam
IAM
Manage User Access and Encryption Keys
EC2
Lightsail
ECR
ECS
EKS
Lambda
Batch
Elastic Beanstalk
Serverless Application Repository
AWS RoboMaker
Satellite
Ground Station
Athena
EMR
CloudSearch
Elasticsearch Service
Kinesis
QuickSight
Data Pipeline
AWS Glue
MSK
Alexa for B
Amazon Ch
WorkMail
End User
WorkSpace
AppStream
WorkDocs
WorkLink
1) Pick Services
2) Search for IAM
3) Select IAM
Management &
Security, Identity, &

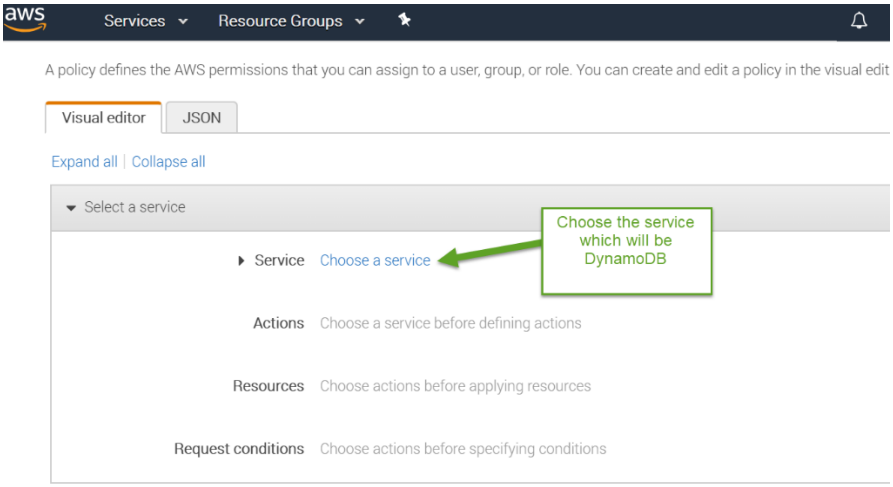

aws
Services
Resource Groups
Search IAM
Dashboard
Groups
Users
Roles
Policies
Identity providers
Account settings
Credential report
Encryption keys
Welcome to Identity and Access Management
IAM users sign-in link:
https://926433000384.signin.aws.amazon.com/console
Customize
IAM Resources
Users: 0
Roles: 6
Identity Providers: 0
Customer Managed Policies: 6
1) Select one of these links to view available Roles
Once selected, you should see the lambda role created previously
Security Status
2 out of 5 complete.
Delete your root access keys
Activate MFA on your root account

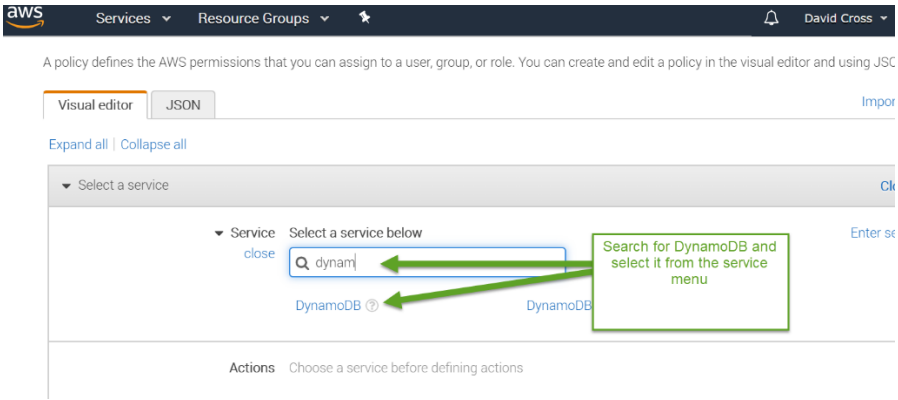

Services
Resource Groups
Roles
What are IAM roles?
IAM roles are a secure way to grant permissions to entities that you trust. Examples of entities include the following
IAM user in another account
Application code running on an EC2 instance that needs to perform actions on AWS resources
An AWS service that needs to act on resources in your account to provide its features
Users from a corporate directory who use
Scroll down
IAM roles issue keys that are valid for short
Additional resources:
IAM Roles FAQ
IAM Roles Documentation
Tutorial: Setting Up Cross Account Access
Common Scenarios for Roles

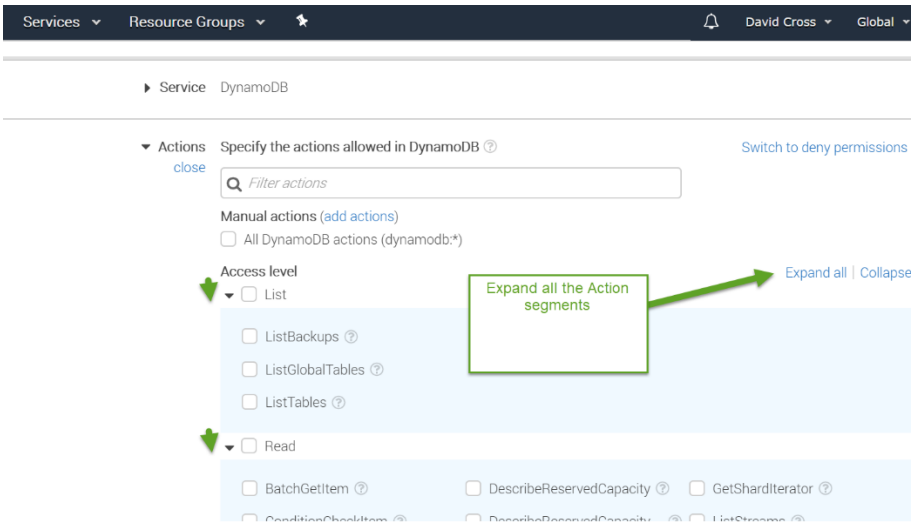

Services
Resource Groups
David Cross
Global
Create role
Delete role
Search
Role name
Description
Trusted entities
AWSServiceRoleForSup...
Enables resource access for AWS to provide billing, ...
AWS service: support (Service-Linked role)
AWS service: trustedadvisor (Service-Linked rol...
AWS service: lambda
You may or may not have other roles available but you'll want to make a copy of the lambda role or, if this is your only lambda function, you can modify the lambda role.
1) Select this user
lambda_basic_execution
Dashboard
Groups
Users
Roles
Policies
Identity providers
Account settings
Credential report
Encryption keys

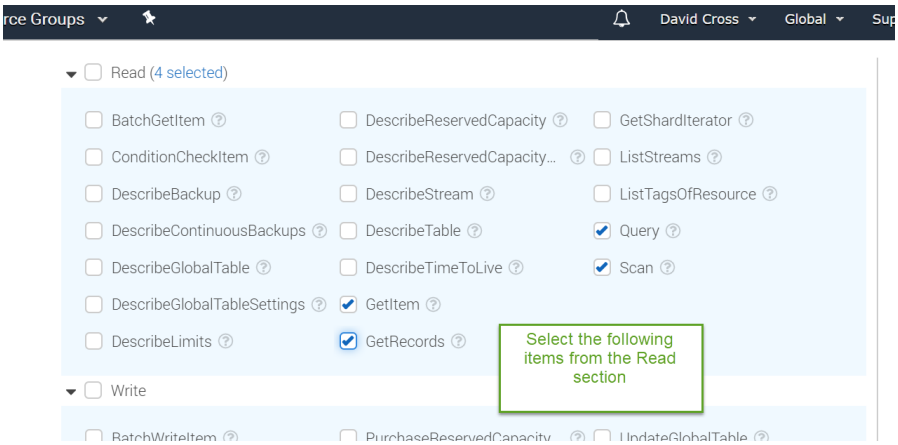

Services
Resource Groups
David Cross
Global
Support
Instance Profile ARNs
Path
/
Creation time
2017-02-21 02:03 MDT
Maximum CLI/API session duration
1 hour Edit
Permissions
Trust relationships
Tags
Access Advisor
Revoke sessions
Permissions policies (1 policy applied)
Scroll down and click Attach Policies
Attach policies
Add inline policy
Policy name
oneClick_lambda_basic_execution_1487667798728
Inline policy
Permissions boundary (not set)

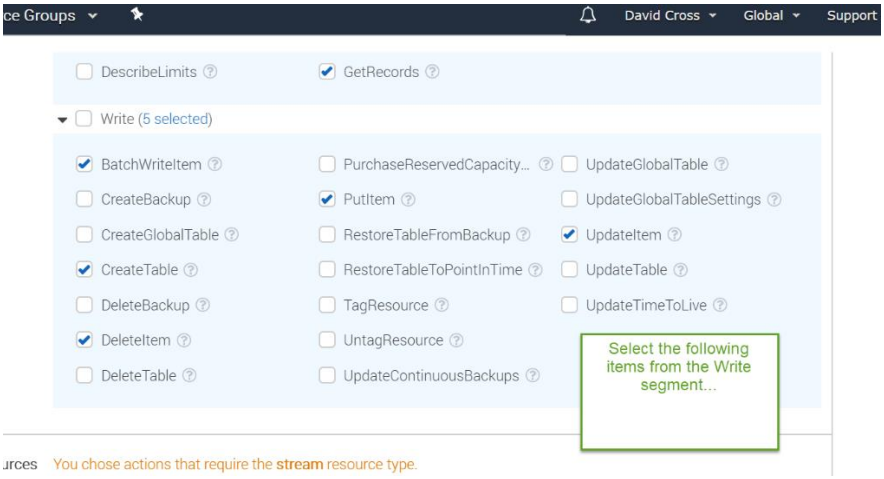

Services
Resource Groups
David
Add permissions to lambda_basic_execution
Attach Permissions
Create policy
Filter policies
DynamoDB
Click Create Policy
Policy nan
Description
Provides full access to Amazo
Provides full access to Amazo
Provides read only access to
AWSLambdaDynamo...
AWS managed
None
Provides list and read access

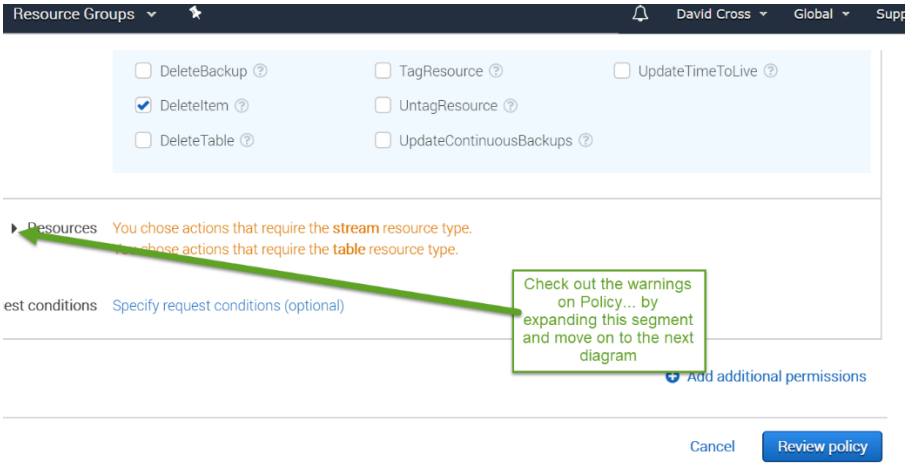

Services
Resource Groups
A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual edit
Visual editor
JSON
Expand all | Collapse all
Select a service
Service
Choose a service
Choose the service which will be DynamoDB
Actions
Choose a service before defining actions
Resources
Choose actions before applying resources
Request conditions
Choose actions before specifying conditions

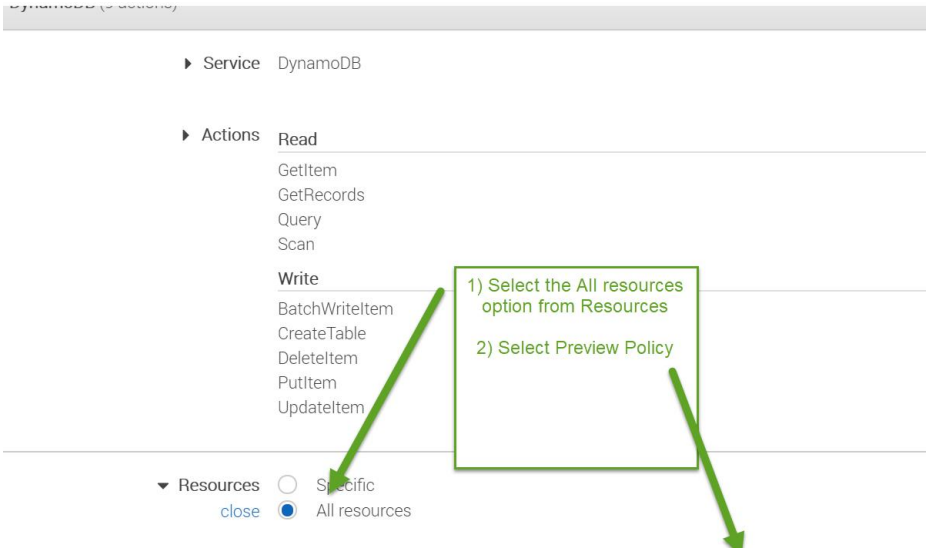

Services
Resource Groups
David Cross
A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSC
Visual editor
JSON
Expand all | Collapse all
Select a service
Service
Select a service below
close
dynam
DynamoDB
DynamoDB
Search for DynamoDB and select it from the service menu
Actions
Choose a service before defining actions

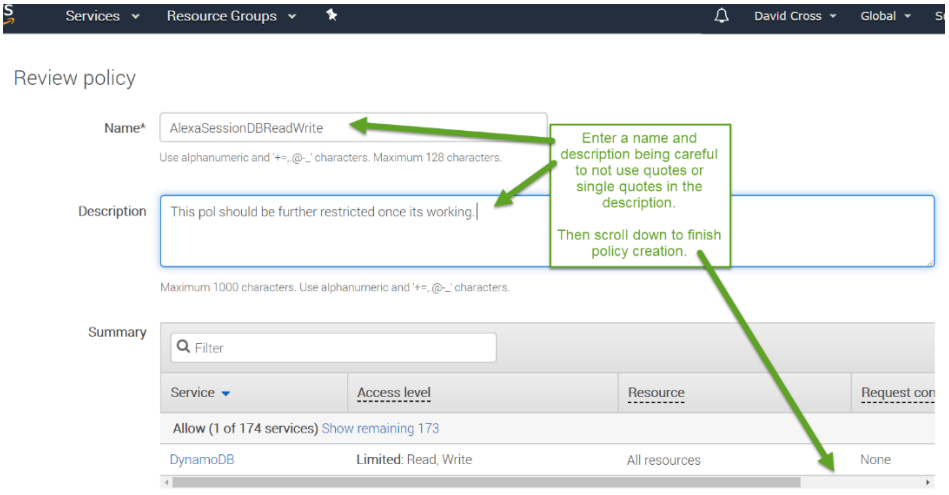

Services
Resource Groups
David Cross
Global
Service DynamoDB
Actions
Specify the actions allowed in DynamoDB
Switch to deny permissions
close
Filter actions
Manual actions (add actions)
All DynamoDB actions (dynamodb:*)
Access level
List
Expand all the Action segments
Expand all | Collapse
ListBackups
ListGlobalTables
ListTables
Read
BatchGetItem
DescribeReservedCapacity
GetShardIterator

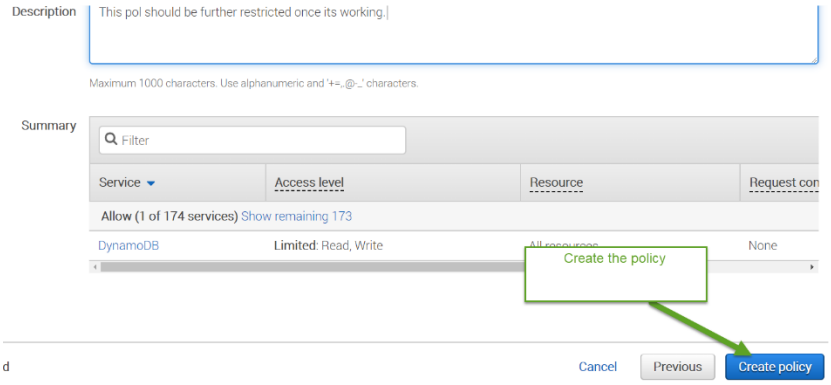

David Cross
Global
Read (4 selected)
BatchGetItem
DescribeReservedCapacity
GetShardIterator
ConditionCheckItem
DescribeReservedCapacity...
ListStreams
DescribeBackup
DescribeStream
ListTagsOfResource
DescribeContinuousBackups
DescribeTable
Query
DescribeGlobalTable
DescribeTimeToLive
Scan
DescribeGlobalTableSettings
GetItem
DescribeLimits
GetRecords
Select the following items from the Read section
Write

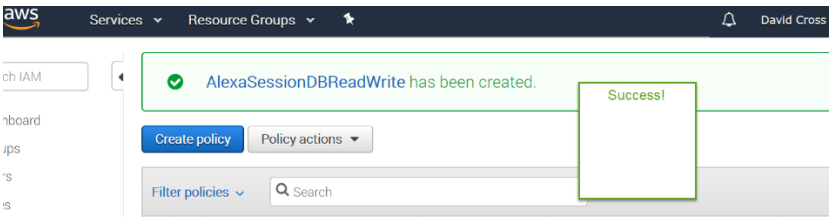

David Cross
Global
Support
DescribeLimits
GetRecords
Write (5 selected)
BatchWriteItem
PurchaseReservedCapacity...
UpdateGlobalTable
CreateBackup
PutItem
UpdateGlobalTableSettings
CreateGlobalTable
RestoreTableFromBackup
UpdateItem
CreateTable
RestoreTableToPointInTime
UpdateTable
DeleteBackup
TagResource
UpdateTimeToLive
DeleteItem
UntagResource
DeleteTable
UpdateContinuousBackups
Select the following items from the Write segment...
You chose actions that require the stream resource type.

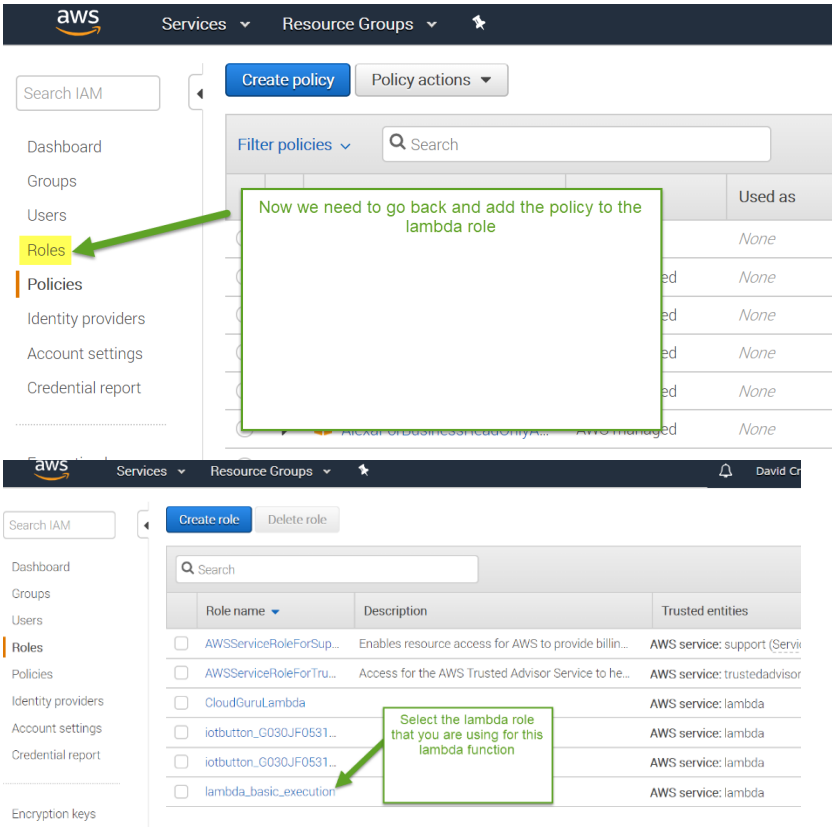

Resource Groups
David Cross
Global
DeleteBackup
TagResource
UpdateTimeToLive
DeleteItem
UntagResource
DeleteTable
UpdateContinuousBackups
Resources
You chose actions that require the stream resource type.
You chose actions that require the table resource type.
est conditions
Specify request conditions (optional)
Check out the warnings on Policy... by expanding this segment and move on to the next diagram
Add additional permissions
Cancel
Review policy

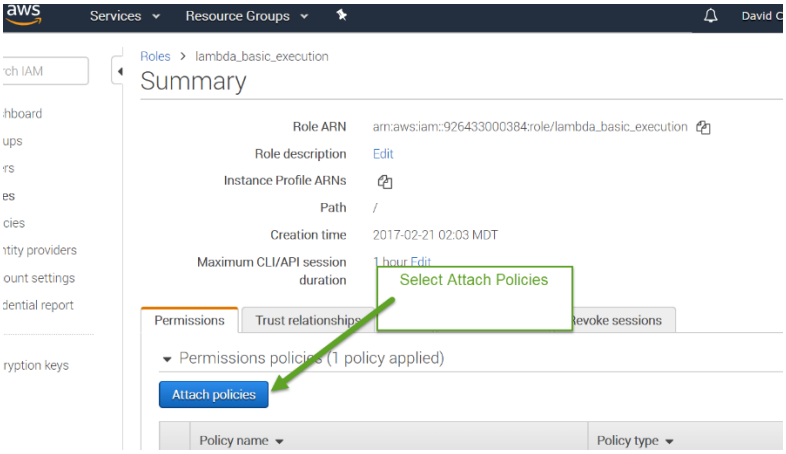

Service
DynamoDB
Actions
Read
GetItem
GetRecords
Query
Scan
Write
BatchWriteItem
CreateTable
DeleteItem
PutItem
UpdateItem
1) Select the All resources option from Resources
2) Select Preview Policy
Resources
close
Specific
All resources

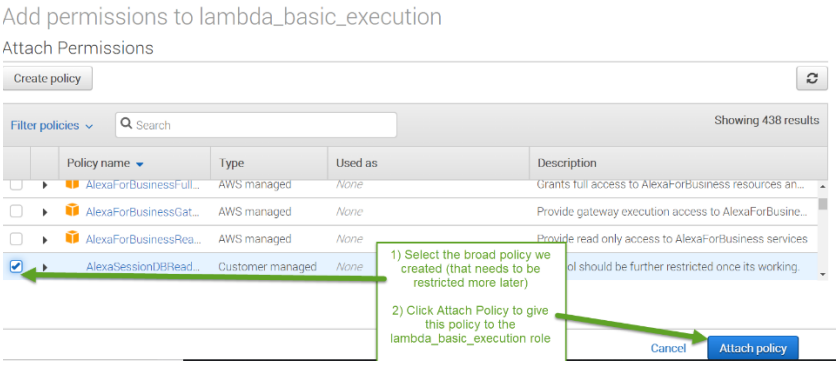

Services
Resource Groups
David Cross
Global
Review policy
Name*
AlexaSessionDBReadWrite
Use alphanumeric and '+=,.@-_' characters. Maximum 128 characters.
Description
This pol should be further restricted once its working.
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.
Enter a name and description being careful to not use quotes or single quotes in the description.
Then scroll down to finish policy creation.
Summary
Filter
Service
Access level
Resource
Request con
Allow (1 of 174 services) Show remaining 173
DynamoDB
Limited: Read, Write
All resources
None

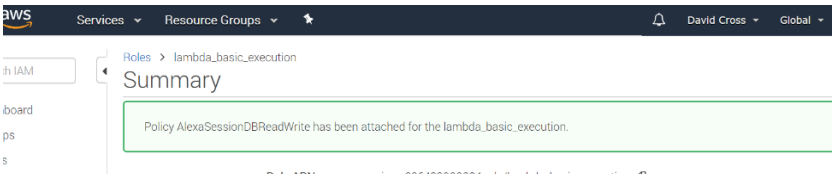

Description
This pol should be further restricted once its working.
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.
Summary
Filter
Service
Access level
Resource
Request con
Allow (1 of 174 services) Show remaining 173
DynamoDB
Limited: Read, Write
None
Create the policy
Cancel
Previous
Create policy


Services
Resource Groups
David Cross
AlexaSessionDBReadWrite has been created.
Success!
Create policy
Policy actions
Filter policies
Search


Services
Resource Groups
Search IAM
Dashboard
Groups
Users
Roles
Policies
Identity providers
Account settings
Credential report
Create policy
Policy actions
Filter policies
Search
Used as
None
Now we need to go back and add the policy to the lambda role
Create role
Delete role
Role name
Description
Trusted entities
AWSServiceRoleForSup...
Enables resource access for AWS to provide billin...
AWS service: support (Servi
AWSServiceRoleForTru...
Access for the AWS Trusted Advisor Service to he...
AWS service: trustedadvisor
CloudGuruLambda
AWS service: lambda
iotbutton_G030JF0531...
AWS service: lambda
iotbutton_G030JF0531...
AWS service: lambda
lambda_basic_execution
AWS service: lambda
Select the lambda role that you are using for this lambda function
Encryption keys


Services
Resource Groups
Roles > lambda_basic_execution
Summary
Role ARN
arn:aws:iam::926433000384:role/lambda_basic_execution
Role description
Edit
Instance Profile ARNs
Path
/
Creation time
2017-02-21 02:03 MDT
Maximum CLI/API session duration
1 hour Edit
Select Attach Policies
Permissions
Trust relationships
Revoke sessions
Permissions policies (1 policy applied)
Attach policies
Policy name
Policy type

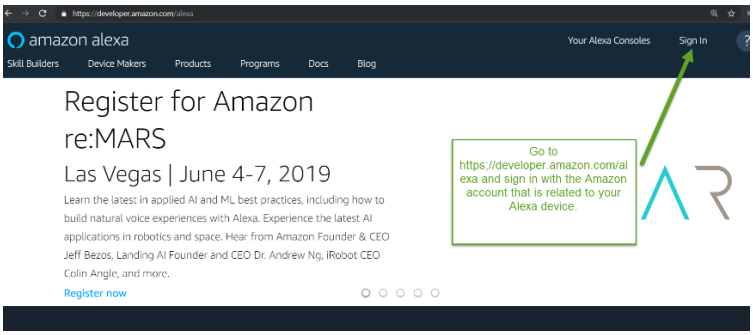

Add permissions to lambda_basic_execution
Attach Permissions
Create policy
Filter policies
Search
Showing 438 results
Policy name
Type
Used as
Description
AlexaForBusinessFull...
AWS managed
None
Grants full access to AlexaForBusiness resources an...
AlexaForBusinessGat...
AWS managed
None
Provide gateway execution access to AlexaForBusine...
AlexaForBusinessRea...
AWS managed
None
Provide read only access to AlexaForBusiness services
AlexaSessionDBRead...
Customer managed
None
ol should be further restricted once its working.
1) Select the broad policy we created (that needs to be restricted more later)
2) Click Attach Policy to give this policy to the lambda_basic_execution role
Cancel
Attach policy

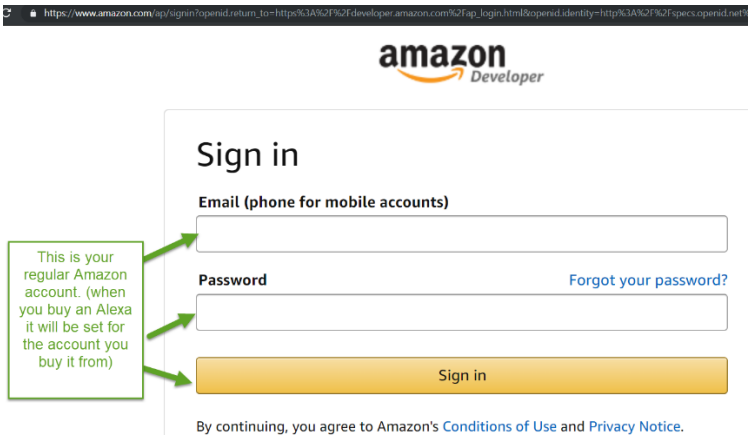

Services
Resource Groups
David Cross
Global
Roles > lambda_basic_execution
Summary
Policy AlexaSessionDBReadWrite has been attached for the lambda_basic_execution.

Step Five (Creating the Alexa Voice configuration)

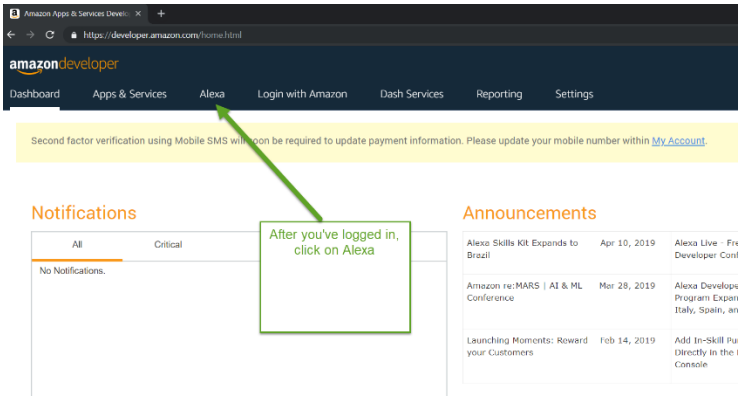
Step Five phase 1

Log into the Alexa Skill site using the account you set up in the pre-requisite steps.

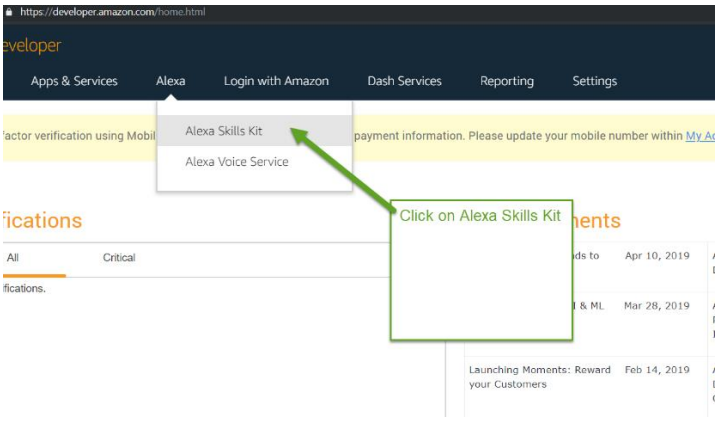




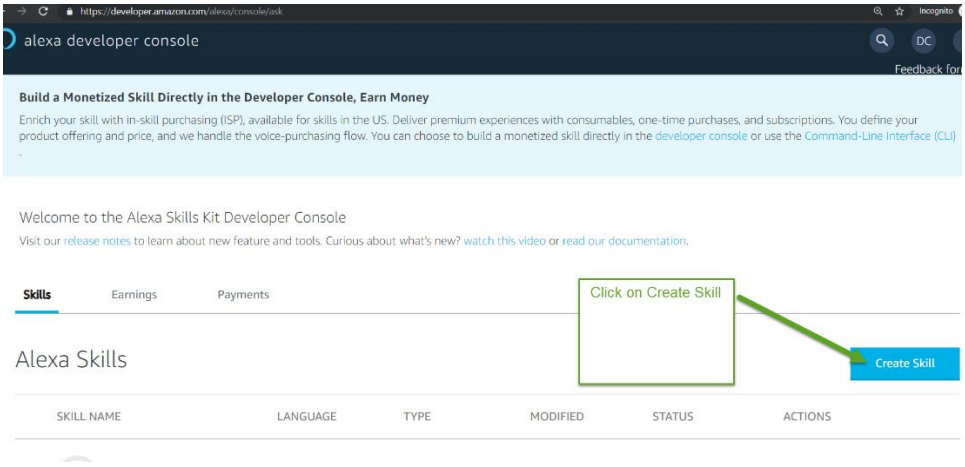
Click Alexa

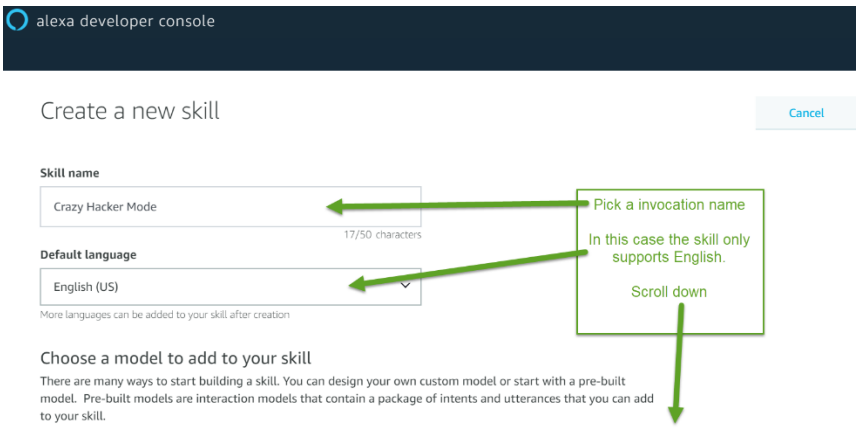


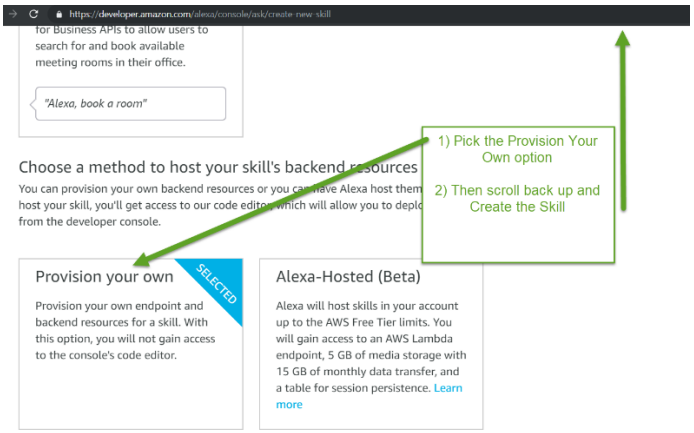
Go to the Alexa Skills Kit

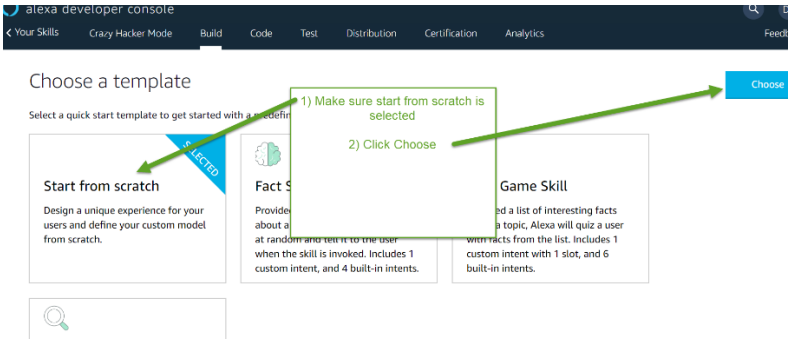


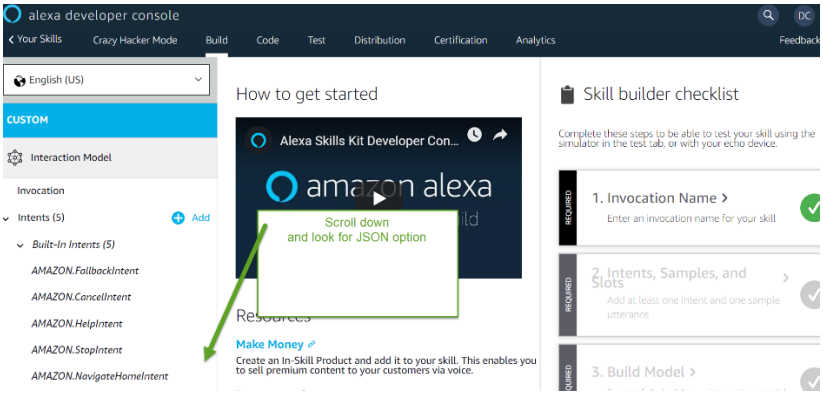
Create a skill as you see in the image below:

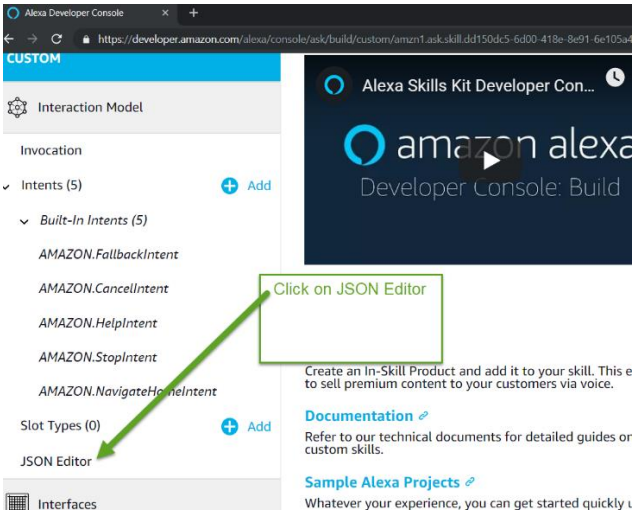


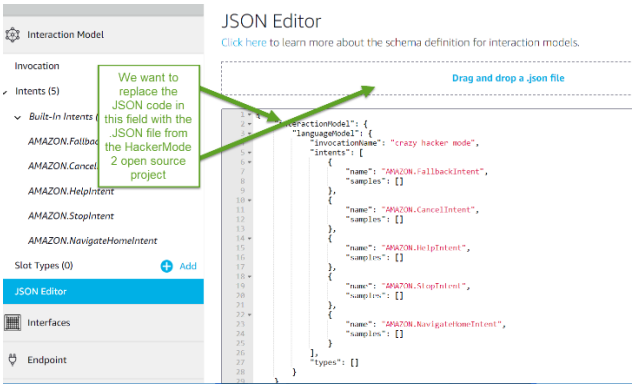


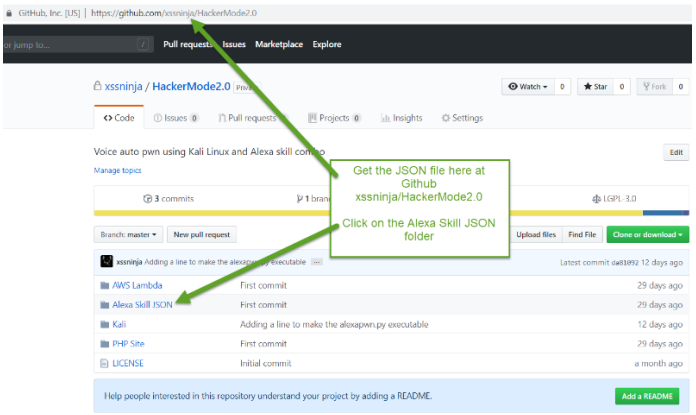

https://developer.amazon.com/alexa/console/ask/create-new-skill
for Business APIs to allow users to search for and book available meeting rooms in their office.
"Alexa, book a room"
1) Pick the Provision Your Own option
2) Then scroll back up and Create the Skill
Choose a method to host your skill's backend resources
Provision your own
SELECTED
Provision your own endpoint and backend resources for a skill. With this option, you will not gain access to the console's code editor.
Alexa-Hosted (Beta)
Alexa will host skills in your account up to the AWS Free Tier limits. You will gain access to an AWS Lambda endpoint, 5 GB of media storage with 15 GB of monthly data transfer, and a table for session persistence. Learn more

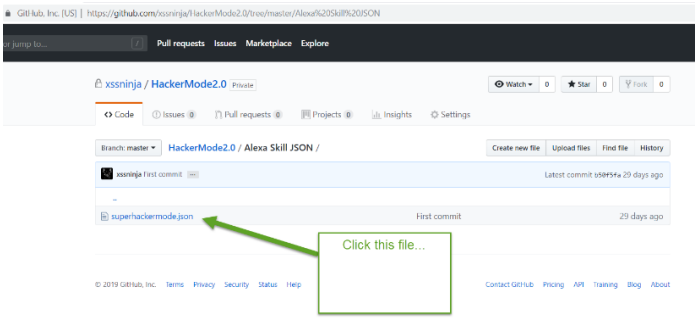

alexa developer console
Your Skills
Crazy Hacker Mode
Build
Code
Test
Distribution
Certification
Analytics
Choose a template
1) Make sure start from scratch is selected
2) Click Choose
Choose
Start from scratch
Design a unique experience for your users and define your custom model from scratch.
Game Skill
custom intent with 1 slot, and 6 built-in intents.

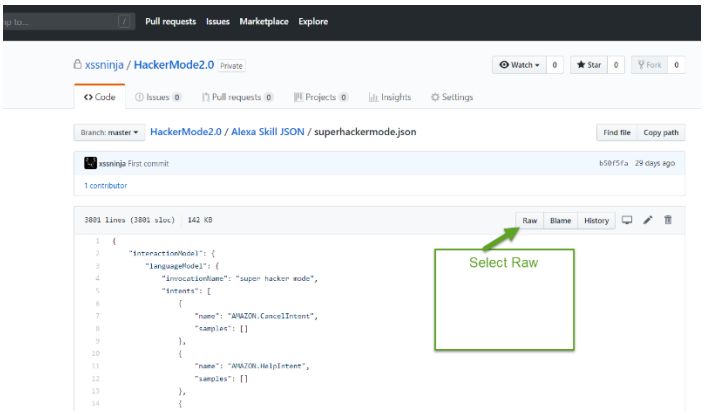

alexa developer console
Your Skills
Crazy Hacker Mode
Build
Code
Test
Distribution
Certification
Analytics
Feedback
English (US)
CUSTOM
Interaction Model
Invocation
Intents (5)
Add
Built-In Intents (5)
AMAZON.FallbackIntent
AMAZON.CancelIntent
AMAZON.HelpIntent
AMAZON.StopIntent
AMAZON.NavigateHomeIntent
How to get started
Alexa Skills Kit Developer Con...
amazon alexa
Scroll down and look for JSON option
Make Money
Create an In-Skill Product and add it to your skill. This enables you to sell premium content to your customers via voice.
Skill builder checklist
Complete these steps to be able to test your skill using the simulator in the test tab, or with your echo device.
1. Invocation Name
Enter an invocation name for your skill
2. Intents, Samples, and Slots
Add at least one intent and one sample utterance
3. Build Model


Alexa Developer Console
CUSTOM
Interaction Model
Invocation
Intents (5)
Add
Built-In Intents (5)
AMAZON.FallbackIntent
AMAZON.CancelIntent
AMAZON.HelpIntent
AMAZON.StopIntent
Slot Types (0)
Add
JSON Editor
Interfaces
Alexa Skills Kit Developer Con...
Developer Console: Build
Click on JSON Editor
Documentation
Refer to our technical documents for detailed guides or custom skills.
Sample Alexa Projects

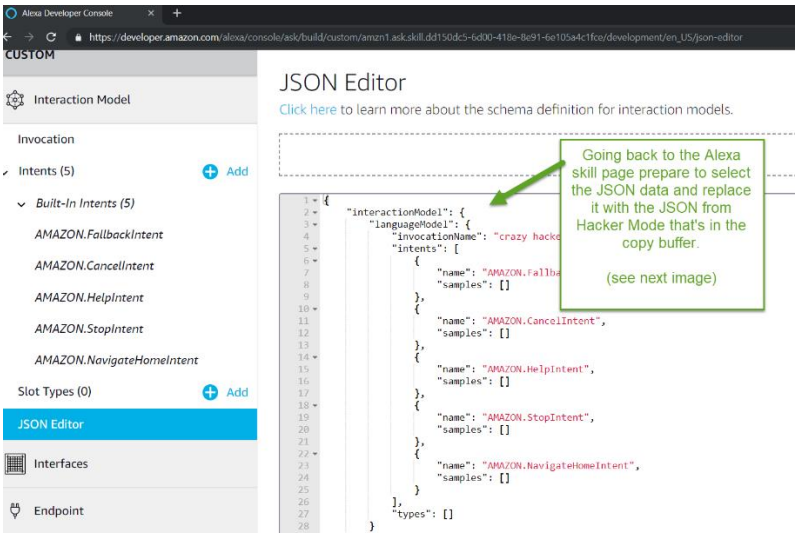

Interaction Model
Invocation
Intents (5)
JSON Editor
Interfaces
Endpoint
We want to replace the JSON code in this field with the JSON file from the HackerMode 2 open source project
JSON Editor
Click here to learn more about the schema definition for interaction models.
Drag and drop a .json file


Pull requests
Issues
Marketplace
Explore
xssninja / HackerMode2.0
Code
Issues
Pull requests
Projects
Insights
Settings
Get the JSON file here at Github xssninja/HackerMode2.0
Click on the Alexa Skill JSON folder
AWS Lambda
Alexa Skill JSON
Kali
PHP Site
LICENSE
Add a README

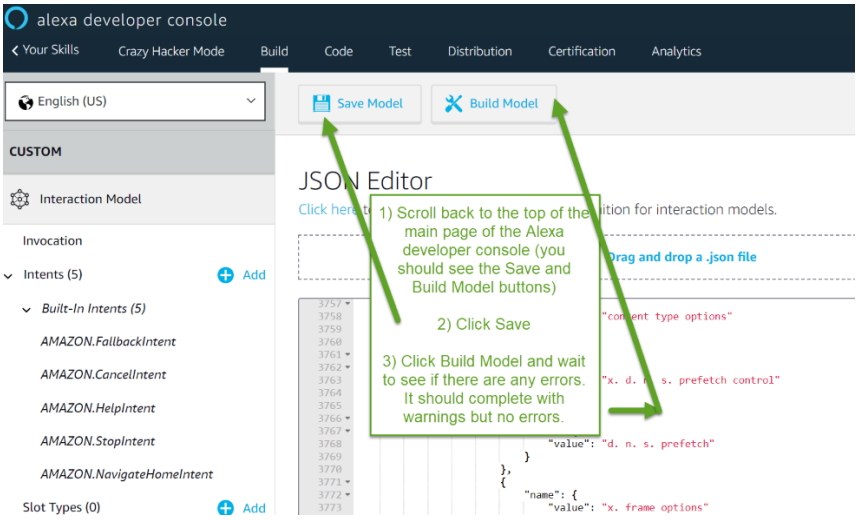

Pull requests Issues Marketplace Explore
xssninja / HackerMode2.0 Private
Code Issues Pull requests Projects Insights Settings
Branch: master HackerMode2.0 / Alexa Skill JSON /
superhackermode.json First commit 29 days ago
Click this file...

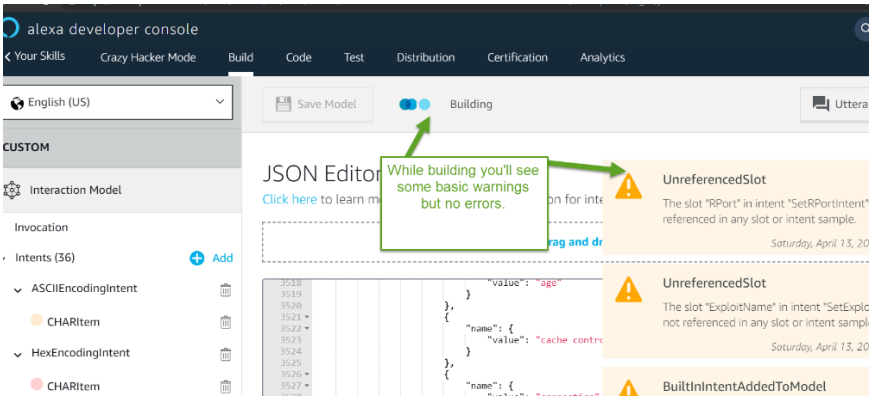

Pull requests Issues Marketplace Explore
xssninja / HackerMode2.0 Private
Code Issues Pull requests Projects Insights Settings
Branch: master HackerMode2.0 / Alexa Skill JSON / superhackermode.json
Find file Copy path
xssninja First commit b50f5fa 29 days ago
1 contributor
3801 lines (3801 sloc) 142 KB
Raw Blame History
Select Raw

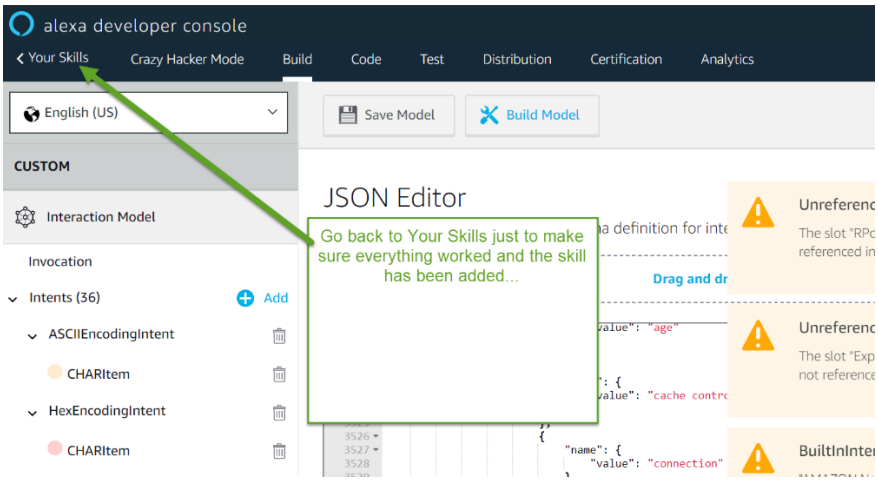

https://raw.githubusercontent.com/xssninja/HackerMode2.0/master/Alexa%20Skill%20JSON/superhackermode.json?token=AYxdoIDQ8RZM-DU
Press Ctrl-A and Ctrl-C to select all and copy to the clipboard

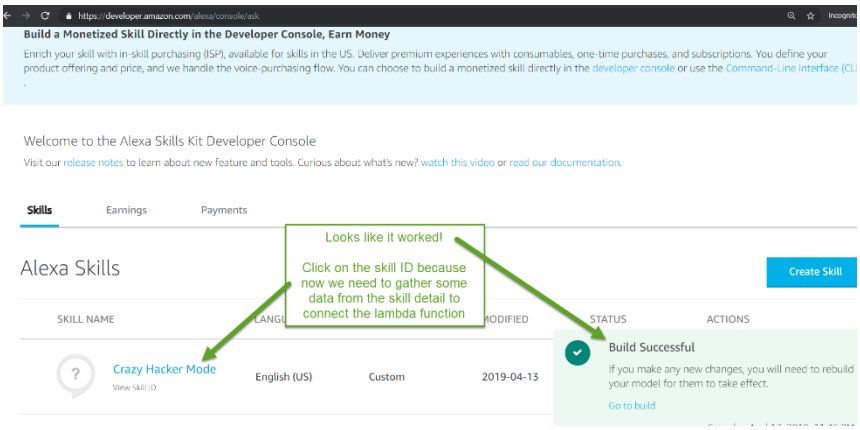

Alexa Developer Console
https://developer.amazon.com/alexa/console/ask/build/custom/amzn1.ask.skill.dd150dc5-6d00-418e-8e91-6e105a4c1fce/development/en_US/json-editor
CUSTOM
Interaction Model
Invocation
Intents (5) Add
Built-In Intents (5)
AMAZON.FallbackIntent
AMAZON.CancelIntent
AMAZON.HelpIntent
AMAZON.StopIntent
AMAZON.NavigateHomeIntent
Slot Types (0) Add
JSON Editor
Interfaces
Endpoint
JSON Editor
Click here to learn more about the schema definition for interaction models.
Going back to the Alexa skill page prepare to select the JSON data and replace it with the JSON from Hacker Mode that's in the copy buffer.
(see next image)


Interaction Model
Invocation
Intents (5) Add
Built-In Intents (5)
AMAZON.FallbackIntent
AMAZON.CancelIntent
AMAZON.HelpIntent
AMAZON.StopIntent
AMAZON.NavigateHomeIntent
Slot Types (0) Add
JSON Editor
Interfaces
Endpoint
JSON Editor
Click here to learn more about the schema definition for interaction models.
After selecting all the text in this box using Ctrl-A and replacing it with the JSON in the copy buffer using Ctrl-C, you should see a portion of the new JSON in your view
Drag and drop a .json file


alexa developer console
Your Skills
Crazy Hacker Mode
Build
Code
Test
Distribution
Certification
Analytics
English (US)
Save Model
Build Model
CUSTOM
Interaction Model
Invocation
Intents (5) Add
Built-In Intents (5)
AMAZON.FallbackIntent
AMAZON.CancelIntent
AMAZON.HelpIntent
AMAZON.StopIntent
AMAZON.NavigateHomeIntent
Slot Types (0) Add
JSON Editor
1) Scroll back to the top of the main page of the Alexa developer console (you should see the Save and Build Model buttons)
2) Click Save
3) Click Build Model and wait to see if there are any errors. It should complete with warnings but no errors.

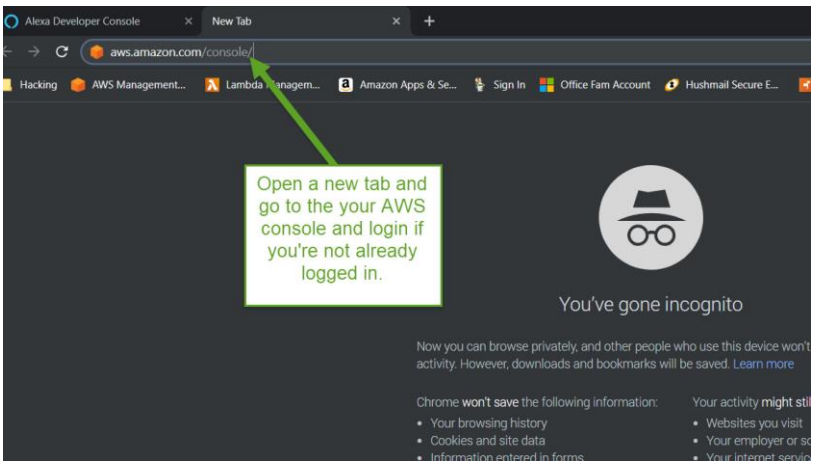

alexa developer console
Your Skills
Crazy Hacker Mode
Build
Code
Test
Distribution
Certification
Analytics
English (US)
Save Model
Building
CUSTOM
Interaction Model
Invocation
Intents (36)
Add
ASCIIEncodingIntent
CHARItem
HexEncodingIntent
CHARItem
JSON Editor
While building you'll see some basic warnings but no errors.
UnreferencedSlot
BuiltInIntentAddedToModel

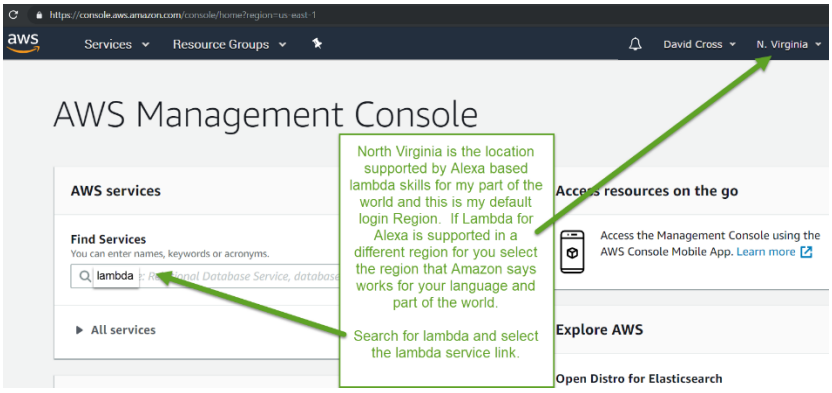

alexa developer console
Your Skills
Crazy Hacker Mode
Build
Code
Test
Distribution
Certification
Analytics
English (US)
Save Model
Build Model
CUSTOM
Interaction Model
Invocation
Intents (36)
Add
ASCIIEncodingIntent
CHARItem
HexEncodingIntent
CHARItem
JSON Editor
Go back to Your Skills just to make sure everything worked and the skill has been added...

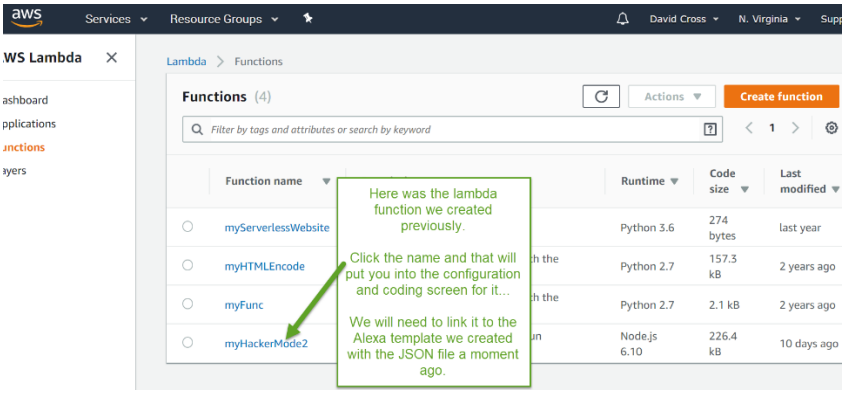

Build a Monetized Skill Directly in the Developer Console, Earn Money
Welcome to the Alexa Skills Kit Developer Console
Skills
Earnings
Payments
Alexa Skills
Create Skill
SKILL NAME
STATUS
ACTIONS
Looks like it worked!
Click on the skill ID because now we need to gather some data from the skill detail to connect the lambda function
Crazy Hacker Mode
English (US)
Custom
2019-04-13
Build Successful

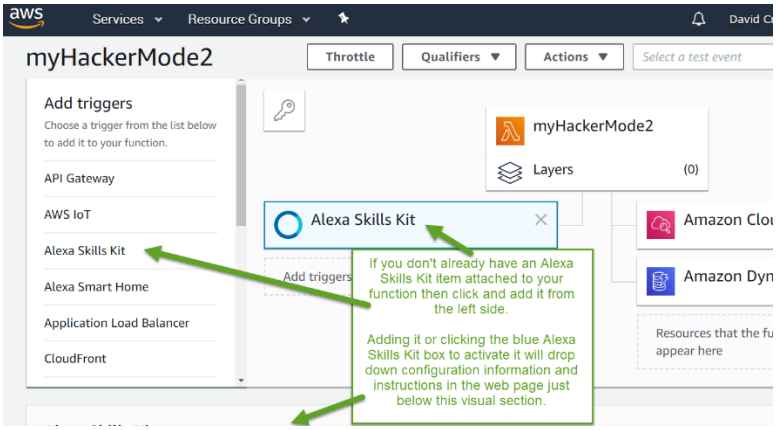
Step Five Phase 2 (Connecting your Lambda function and your Alexa skill together)

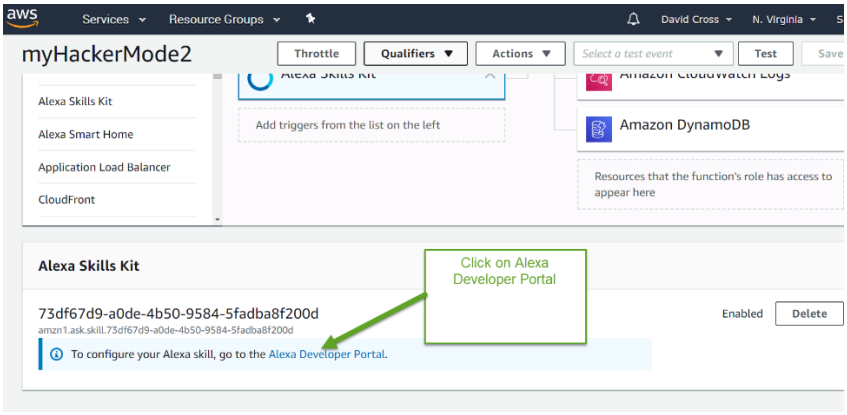
Log into the AWS console using your account created in the prerequisite steps.

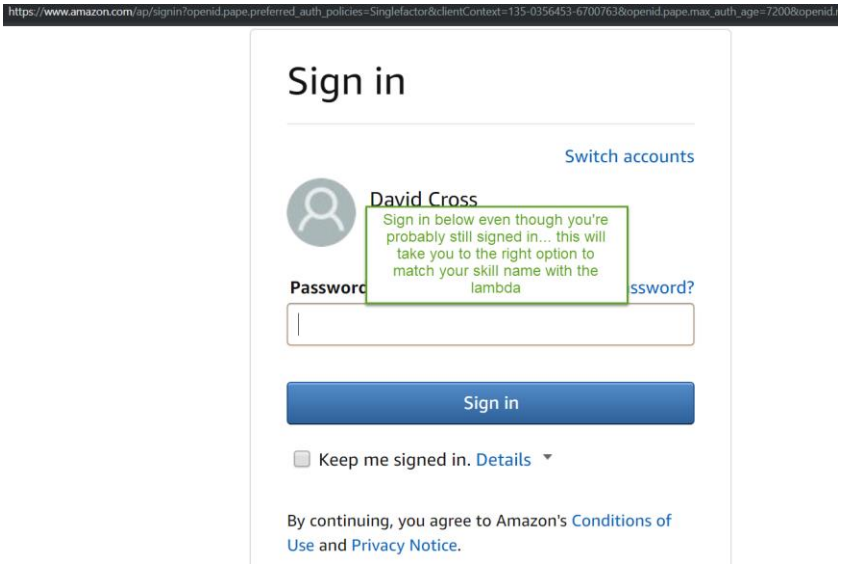


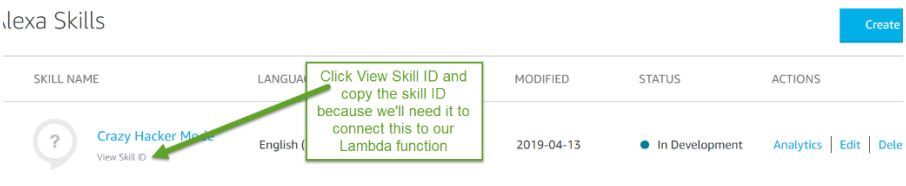


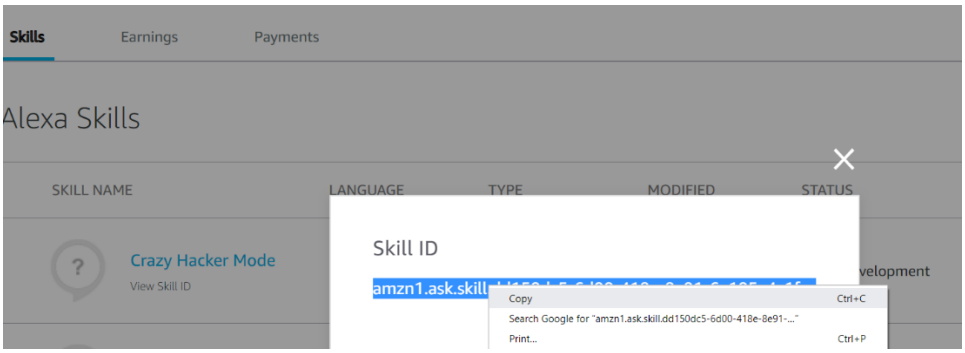



aws
Services
Resource Groups
myHackerMode2
Throttle
Qualifiers
Actions
Select a test event
Add triggers
Choose a trigger from the list below to add it to your function.
API Gateway
AWS IoT
Alexa Skills Kit
Alexa Smart Home
Application Load Balancer
CloudFront
myHackerMode2
Layers
(0)
Alexa Skills Kit
Amazon Clou
Amazon Dyn
Add triggers
Resources that the fu appear here
If you don't already have an Alexa Skills Kit item attached to your function then click and add it from the left side.
Adding it or clicking the blue Alexa Skills Kit box to activate it will drop down configuration information and instructions in the web page just below this visual section.

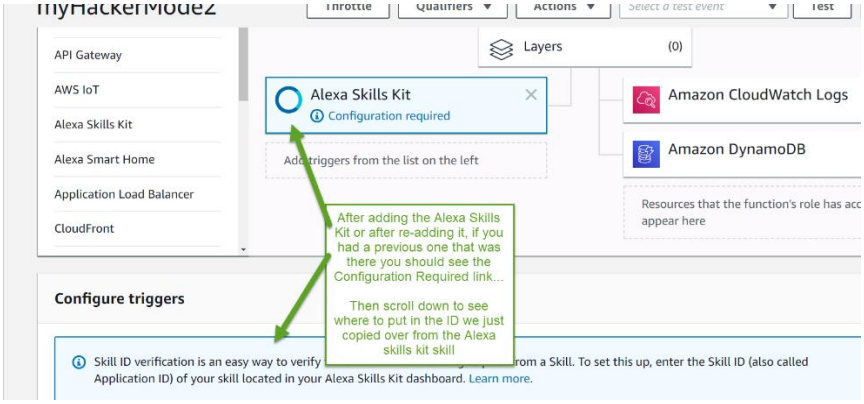

aws
Services
Resource Groups
David Cross
N. Virginia
myHackerMode2
Throttle
Qualifiers
Actions
Select a test event
Test
Save
Alexa Skills Kit
Alexa Smart Home
Application Load Balancer
CloudFront
Add triggers from the list on the left
Amazon DynamoDB
Resources that the function's role has access to appear here
Alexa Skills Kit
Click on Alexa Developer Portal
73df67d9-a0de-4b50-9584-5fadba8f200d
amzn1.ask.skill.73df67d9-a0de-4b50-9584-5fadba8f200d
Enabled
Delete
To configure your Alexa skill, go to the Alexa Developer Portal.

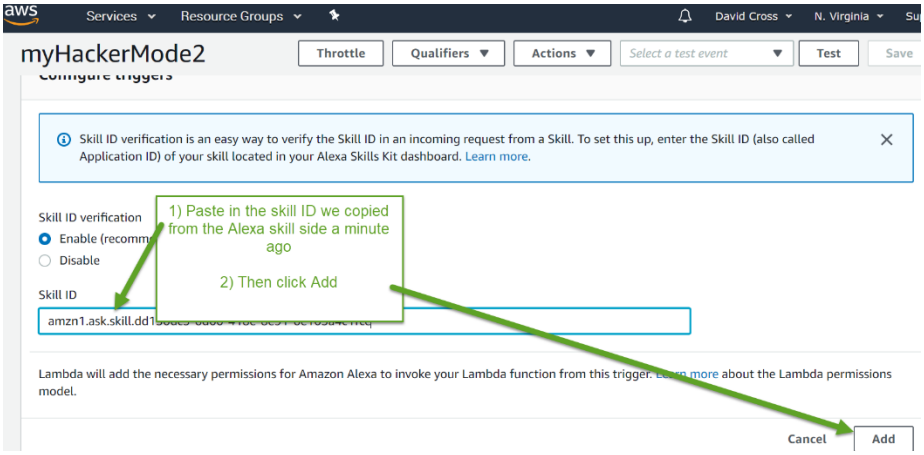

https://www.amazon.com/ap/signin?openid.pape.preferred_auth_policies=SingleFactor&clientContext=135-0356453-6700763&openid.pape.max_auth_age=7200&openid
Sign in
Switch accounts
David Cross
Sign in below even though you're probably still signed in... this will take you to the right option to match your skill name with the lambda
Password
ssword?
Sign in
Keep me signed in. Details
By continuing, you agree to Amazon's Conditions of Use and Privacy Notice.

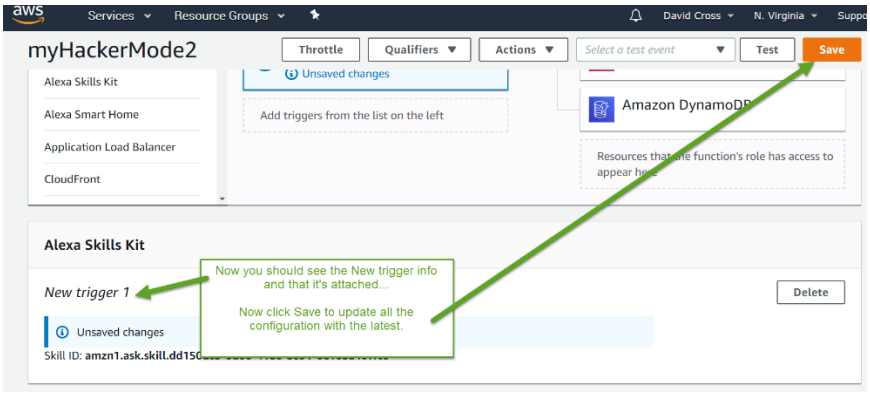

lexa Skills
Create
SKILL NAME
LANGUA
MODIFIED
STATUS
ACTIONS
Click View Skill ID and copy the skill ID because we'll need it to connect this to our Lambda function
Crazy Hacker Mode
View Skill ID
English (
2019-04-13
In Development
Analytics
Edit
Dele



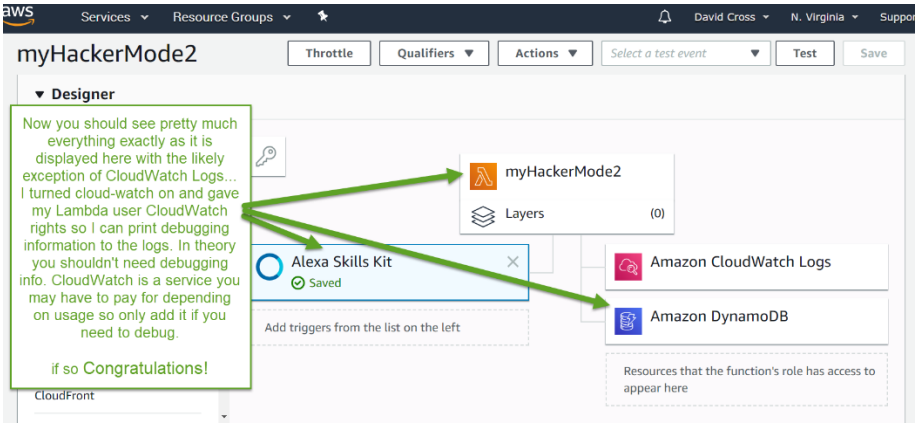
Step Five Phase 3 (Add a Lambda trigger for your Alexa Skill)





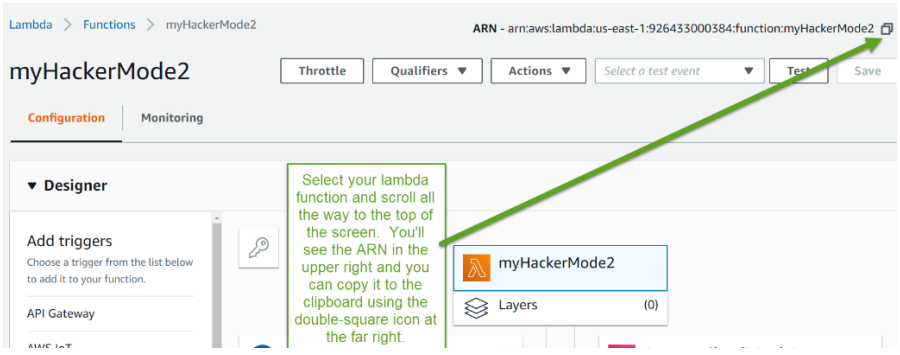


You should see things pretty much as below.

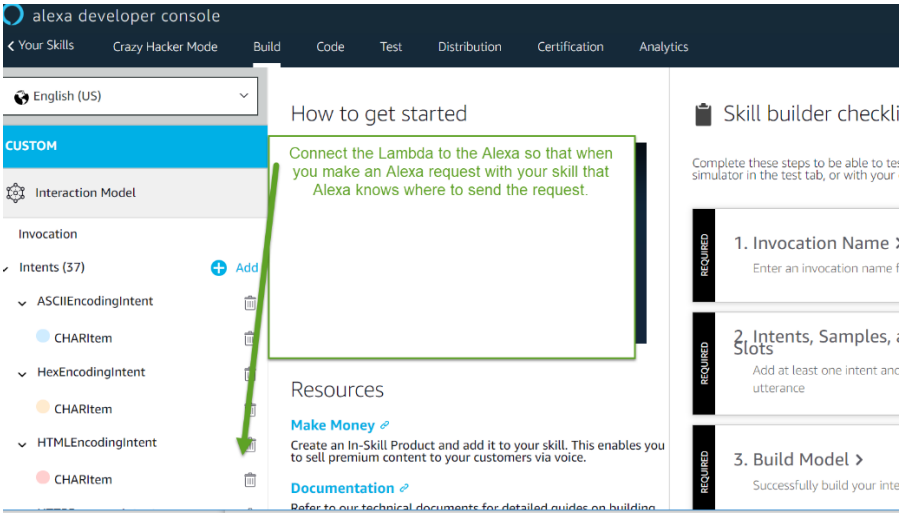


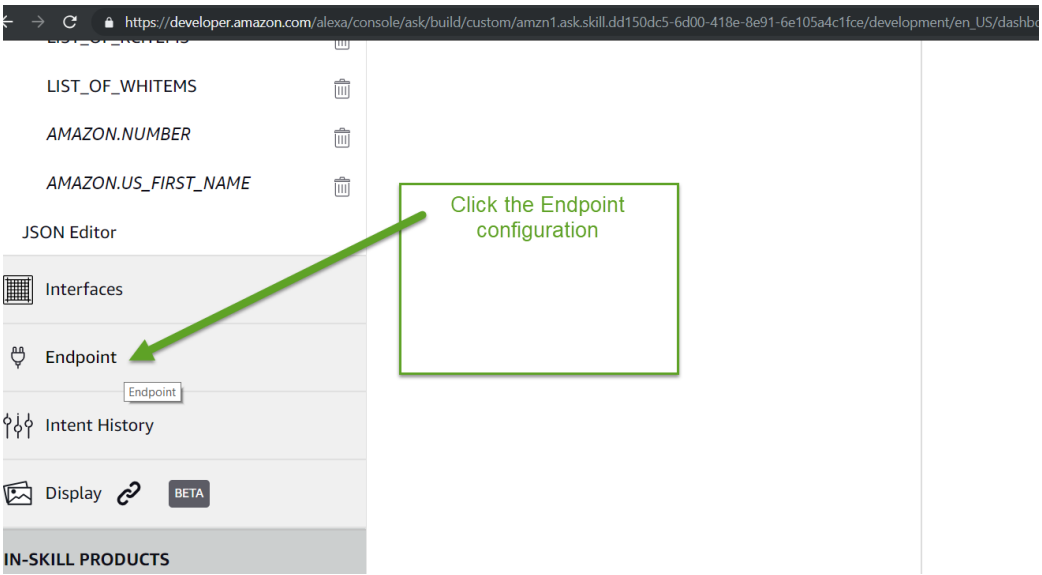
Step Five Phase 4 (Connecting the ARN)

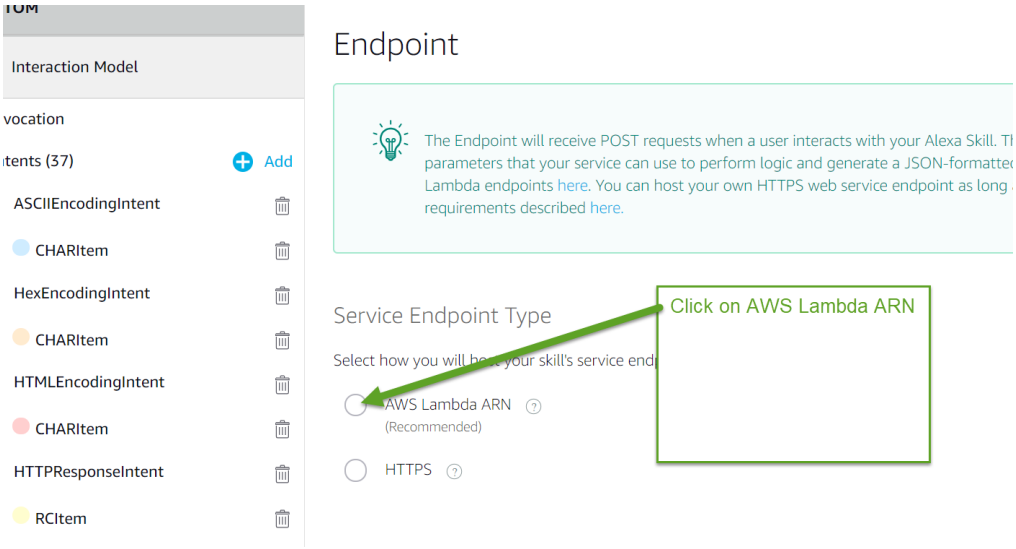
Connecting the Lambda function and the Skill together.



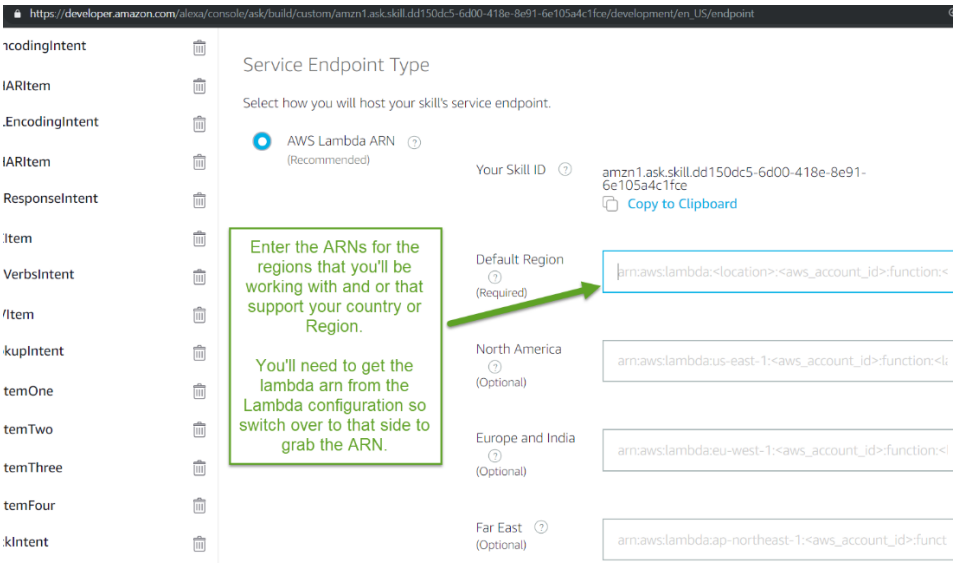
Now go back to the Alexa Skills site and prepare to connect the Lambda and the Skill together.



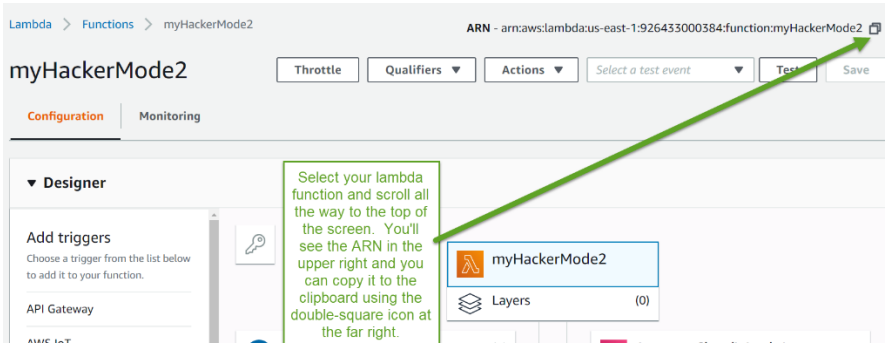




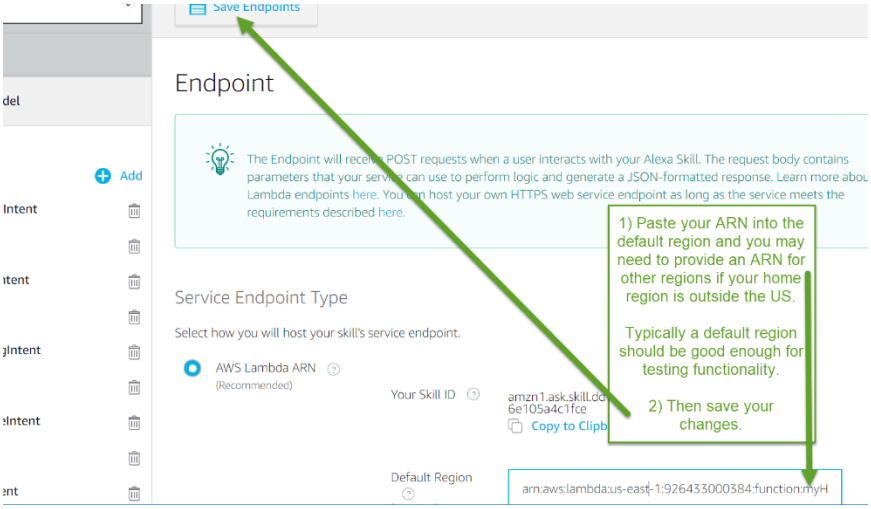
Amazon needs a bit more information about where the Lambda is running. This will normally be the North VA area if you are in the USA as that region has both Dynamo DB support and Alexa Lambda support.



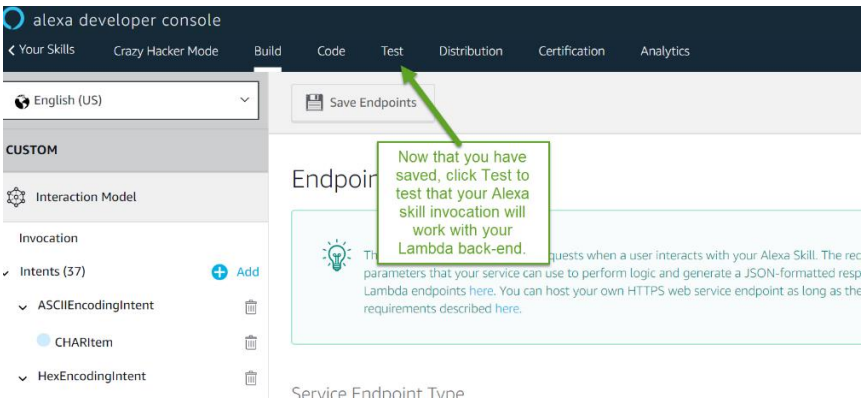
Previously we copied the ARN from the AWS Lambda information but here is the place to copy it from is you missed that step.



Paste that ARN from the Lambda side into the Skill account information screen as seen below:



Now that we have them connected we can test the connection…



Install Step Six (Install Kali Side of Hacker Mode 2.0)

Prerequisite (Your Kali will need both Python 3 and 2.7! Why? Because of install dependencies for 3rd party python libraries that will break otherwise) Default Kali should have 2.7 and 3 pre-installed, but if you are building a container you will need to include it.)

Phase 1 Install nmap library for Python

Install nmap for python 2.7:

- **pip install python-nmap**

(http://xael.org/norman/python/python-nmap/)

Phase 2 Install fixed MSFRpcd Python library

- **python msfrpcd.py install**

Phase 3 Install Pymetasploit from original source

See the following steps:

2. **pip install pymetasploit**

Phase 4 Re-install Pymetasploit from xssninja Github

The original pymetasploit has a flaw that is fixed by the Mikaayenson version of the PyMetasploit repo. I forked it because I want to add more custom features to it, and I want it to be available in the event someone patches the others with unexpected “fixes”, but you can install the corrected code easily from the fork on my Github as shown below:

```
root@kali:~/Downloads# pip install git+https://github.com/xssninja/pymetasploit
Collecting git+https://github.com/xssninja/pymetasploit
  Cloning https://github.com/xssninja/pymetasploit to /tmp/pip-req-build-xeb8_O
Collecting msgpack-python>=0.1.12 (from pymetasploit==1.1)
  Downloading https://files.pythonhosted.org/packages/8a/20/6eca772d1a5830336f84aca1d
8198e5a3f4715cd1c7fc36d3cc7f7185091/msgpack-python-0.5.6.tar.gz (138kB)
    100% |████████████████████████████████| 143kB 1.9MB/s
Building wheels for collected packages: pymetasploit, msgpack-python
  Running setup.py bdist_wheel for pymetasploit ... done
  Stored in directory: /tmp/pip-ephem-wheel-cache-sTlWFz/wheels/1f/6b/99/c9ad9822e15b
21f72e7b4cae0916b50e393a6b736e45cc87d5
  Running setup.py bdist_wheel for msgpack-python ... done
  Stored in directory: /root/.cache/pip/wheels/d5/de/86/7fa56fda12511be47ea0808f3502b
c879df4e63ab168ec0406
Successfully built pymetasploit msgpack-python
Installing collected packages: msgpack-python, pymetasploit
Successfully installed msgpack-python-0.5.6 pymetasploit-1.1
root@kali:~/Downloads#
```

Phase 5 Install AlexaPwn.py file

- Grab AlexaPwn.py from Github and put it where you want.

Phase 6 Install and configure AlexaPwn6cfg.py file

- Grab AlexaPwncfg.py from Github and put it in the same directory as alexapwn.py.
- Configure AlexaPwn6cfg.py with the paths to your Alexa-bound PHP queue path and the subnet prefix of your target subnet.
- Complie Alexapwn6cfg.py to Alexapwn6cfg.pyc or it won't be importable into alexapwn.py
- Edit the alexapwn6cfg.py file to point to the hosted php files Kali2Alexa.php and Kali_Read.php
- ```
  root@kali:~/Documents/alexapwn# gedit alexapwn6cfg.py
  ```

  Open ▾ | alexapwn6cfg.py ~/Documents/alexapwn

  ```
  #!/usr/bin/env python
  kalicfg = {'Kali2Alexa': 'https://www.[redacted].ninja/Kali2Ale[redacted].php',
          'Kali_Read': 'https://www.[redacted]ninja/Kali_Rea[redacted].php',
          'default_lport': '4000',
                  'default_msfrpc_port': 55553}
  ```
- Compile it with Python 2.7 like so:
- ```
  root@kali:~/Documents/alexapwn# python -m py_compile alexapwn6cfg.py
  ```
- Ignoring the extra crap in this directory that you don't have in yours… note that the file in .pyc form should exist and be importable into Alexapwn.py
- ```
  root@kali:~/Documents/alexapwn# ls
  2to3-2.7                          alexapwn3.py          alexapwn6.py
  activate-global-python-argcomplete alexapwn4.py         alexapwn-min.py
  alexapwn2cfg.py                   alexapwn5a.py         alexapwn_orig.py
  alexapwn2.py                      alexapwn5b.py         alexapwn.py
  alexapwn2-with_new_tags.py        alexapwn5.py          originaldev
  alexapwn3cfg.py                   alexapwn6cfg.py
  alexapwn3cfg.pyc                  alexapwn6cfg.pyc
  ```

Phase 7 Run MSFRPC daemon

AlexaPwn.py depents on MSFRPCD to run. Start it before launching the skill.

- **msfrpcd -S -f -U msf -P test**

If you want to save this script and chmod it to executable, you can drop it on your desktop or otherwise script it to start to launch MSFRPCD for you.
  ```

Phase 8 Start AlexaPwn.py

Run the python side of HackerMode on Kali with Python 2.7. If you have 2.7 as your default Python it should be as easy as:

- **Python alexapwn.py**

Docker Alternative for Kali Install

Install the Docker Container created by James Blackburn for the Kali side of things and avoid having to do any of the Kali config/install stuff.

Here is the URL to the repo: https://hub.docker.com/r/eli23/kali_pwn

Running Hacker Mode2 soup-to nuts

1. Start your Kali instance and make sure you have internet and are in the subnet range you configured in your Lambda. (ifconfig and ping are helpful obviously)
2. Start Msfrpcd in the terminal in Kali if it's not already running: **msfrpcd -S -f -U msf -P test**
3. Wait a second and start: **python alexapwn.py** in the terminal and watch for errors:
    1. If the config file won't load, make sure it's been updated and compiled so Python can load it.
    2. If it warns of MSFrpcd not being ready, make sure you wait a few seconds for the daemon to fully load before running alexapwn.py
    3. If it warns of failures reaching the queue PHP page online, check your path in the config (you may need to recompile the config if you had to make a change). Also, hit the queue page from a browser to be sure your site is up and the path is OK.
4. Ensure you have permission to scan stuff in your Kali subnet. If you don't you are going to pwn your life!
5. Ensure your victim machine is online and your subnet is correct for targeting it. Ping is your friend.
   IT WILL HAVE TO BE IN THE SAME SUBNET AS YOUR KALI! That is unless you customize the project to target external systems. Beware however, that if you say the wrong IP, or Alexa misunderstands you and you just start pwning a box that's not yours… you'll be in a world of hurt. Don't be that person! You know the song: "Don't go chasin' waterfalls, stick to the rivers and the lakes that you're used to…" You probably look better in colors other than orange if you know what I mean!
6. Make sure Alexa is on, connected, and your are not muted… Invoke the skill by saying "**Alexa, start Hacker Mode**"
7. If the skill responds you can start immediately with hacking a box using an instruction like: "hack IP address 137.121" NOTE THAT THE FIRST TWO IP NUMBERS NOT PRESENT ARE ONES YOU CONFIGURE IN THE ALEXA SKILL CONFIGURATION PORTION. THEY ARE SET IN CONFIG AND NOT EASILY CHANGEABLE BY DESIGN.

8. If the Alexa skill immediately quits it's probably because the configuration for the QUEUE URL is wrong. Check it and fix it in your Lambda runtime variables config which is located below the code window in the lambda code editor window of your AWS account.
9. You should see a command to Nmap a box appear in your Kali and Alexa should play music. If you don't hear music you messed up the lambda config for where to load the MP3 file from. (You can make your own MP3s for it with Audacity, LAME and or some help from a mp3 to mp3 converting website online as Alexa likes a weird variety of MP3 that is special.)
10. You can ask Alexa questions about things like "what runs on port 1433?" "what is the hex encoding for double-quote" "what is HTTP response code 403?" "Rick Roll David" "what is the URL encoding for space" "what runs on port 443?" "what does the header x-xss protection do?" and a million other things and infinite variations on those. If the skill times out then just wake it back up with "Alexa start Hacker Mode" and resume from there.
11. As the hack progresses, and since Alexa a) can't stay open for more than 8 seconds, and b) doesn't know everything that's happening moment to moment during the hack you will need to ask it to check by saying "**how's the hack going?**"
12. Alexa should read back open ports or tell you it doesn't have information yet. From this point, Alexa will give Kali the next instructions which will be a list of exploits to try that appear to match the open ports and service names found during the scan portion of the interaction.
13. You'll notice the Kali side will start chugging though exploits. If you start to see interesting things on the screen it's probably gotten a session.
14. By having the skill active and asking "**How's the hack going?**" Alexa will read the progress of the hack back to you including success or failure in getting hashes and list out the users if it was able to escalate privileges successfully.
15. THAT's IT! Repeat at your leisure and explore the code because there is a wealth of knowledge you can ask Alexa about on things that are pretty relevant when you're web app hacking at the very least.