

User Manual

**For creating Box Skill using AWS Lambda and Microsoft
Video Indexer**

Table of Contents

Environment Variable Checklist.....	3
Prerequisites.....	5
• Installing Node.js.....	5
• Installing Severless.....	5
Box Set Up.....	6
• Signing up for Box.....	6
• Configuring Box.....	6
• Creating a folder.....	6
• Connects Skill to Folder.....	6
Microsoft Video Indexer Set Up.....	7
• Subscribing to Microsoft Video Indexer.....	7
• Account ID.....	7
• Account Keys.....	7
Set Up Serverless.....	8
• Obtaining the code.....	8
• Use NPM to install required packages.....	9
• Obtain AWS access information.....	10
• Set Up credentials within serverless.....	11
• Deploying Lambda.....	12
• Deploying Lambda.....	13
• Configuring Box.....	14
• Get S3 Bucket Name.....	16
• Set Up Environment Variables.....	18

Environment Variable Checklist

Variables	Description	Location
API GATEWAY:	API Gateway, invocation URL, endpoint between different Box, serverless, and Lambda. Obtain through serverless deployment to AWS Lambda	Deploying to AWS Lambda
FROM EMAIL:	Email receives the metadata, Box email in this case	Box Account
FROM EMAIL PASS:	Password of receiving account	Box Account
TO_EMAIL:	Email that receive the files and perform cognitive skills to the files	Microsoft Video Indexer Account
Video Indexer ACCOUNT ID:	Account ID of video indexer account where subscription of API was authorized.	Microsoft Video Indexer Account ID
Video Indexer AUTH KEY 1:	Authorizing access to the API for that particular account	Microsoft Video Indexer Primary Key
Video Indexer AUTH KEY 2:	Authorizing access to the API for that particular account	Microsoft Video Indexer Secondary Key
S3 BUCKET NAME	Auto generated S3 bucket, need for configuring lambda function	S3 BUCKET

Prerequisites

Following software need to be installed before uploading videos or creating accounts:

Node.js

In order to deploy to Serverless, first need to install Node.js CLI. Follow the link to install the correct version per machine <https://nodejs.org/en/download/>.

Serverless

Install Serverless: <https://www.npmjs.com/package/serverless>

Box Set Up

Signing up for Box

Sponsor should have provide an invitation link to sign up for Box.com

If not, contact them or sign up using following link <http://www.box.com/pricing>

Configuring Box

To set up a Box skill, login to Box developer console using following link <https://developer.box.com/>

Then, follow this link to configure the Box skill

<http://developer.box.com/guides/applications/custom-skills/setup/>

Creating a Folder

The purpose of creating a folder is to instruct what language Microsoft Video Indexer should use and to trigger any video upload to this folder. To create a folder using following link

<http://support.box.com/hc/en-us/articles/360043696394-Create-New-Files-And-Folders>

Connect Skill to Folder

To approve and connects skill to folder using following link

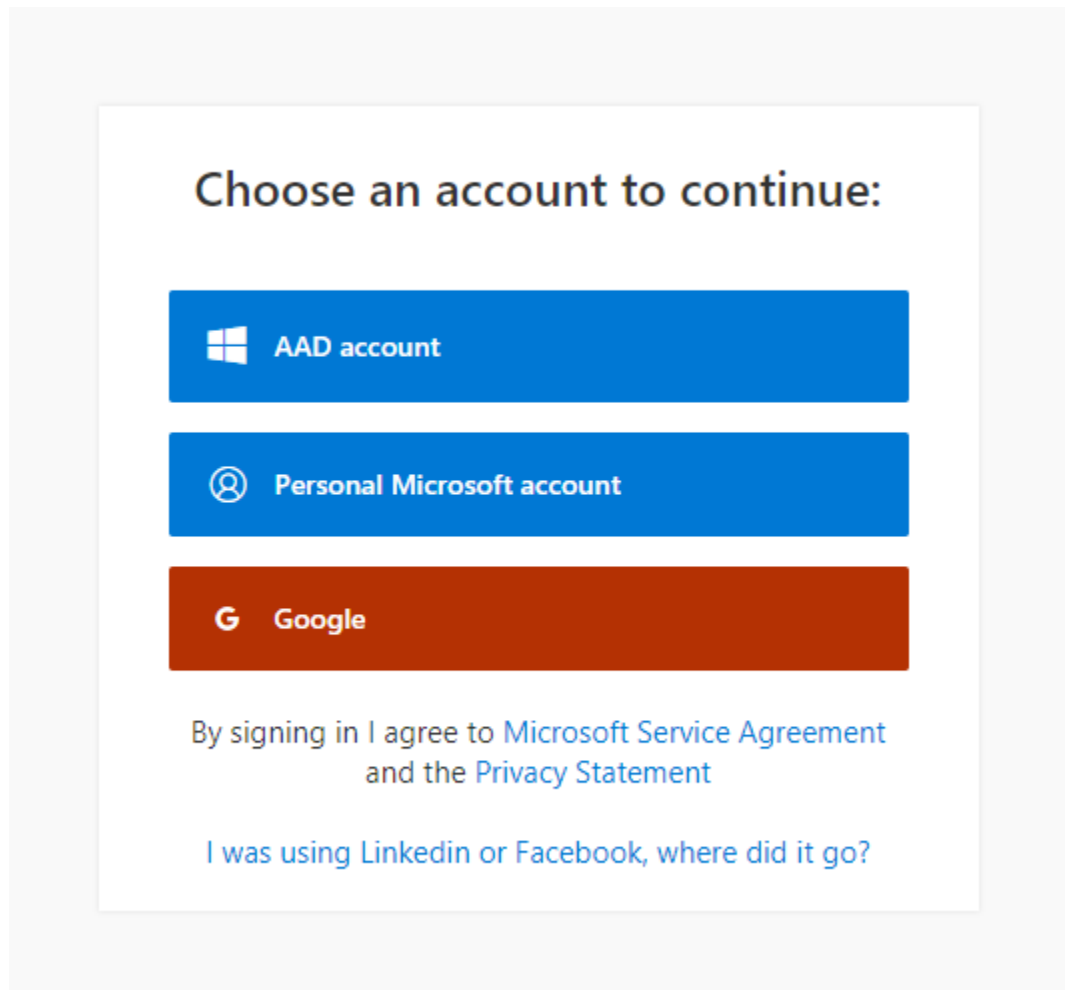
<https://developer.box.com/guides/authorization/custom-skill-approval/>

Microsoft Video Indexer Set Up

Sign into the video indexer using

<https://vi.microsoft.com/en-us>

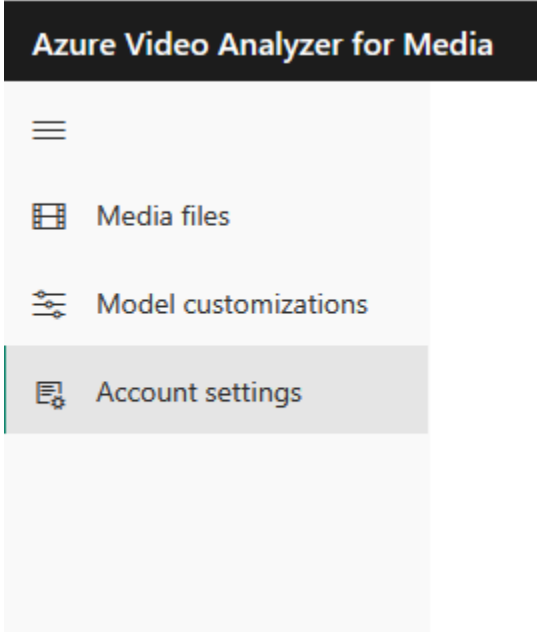
You should be able to see following screen after click on Sign in icon on upper right corner.



We need to stay consistent with the account when signing in. Google account will be a trail account, which is good for testing purpose.

Video Indexer Account ID

To retrieve the account ID, locate the **account settings** from the upper left toolbar



Video Indexer Account Keys

To retrieve the account keys, sign in developer portal using following link

<https://api-portal.videoindexer.ai/>

Go to profile after signed in and following interface should appear

The screenshot shows the 'Profile' page of the Azure Video Analyzer for Media developer portal. The page has a top navigation bar with 'Home', 'APIs', 'Developer Support', 'Profile', and 'Sign out'. The main content area is titled 'User profile' and is divided into two sections: 'Account details' and 'Subscriptions'. The 'Account details' section lists the user's email (huangchuang0102@gmail.com), first name (chuang), last name (huang), and registration date (10/21/2021). The 'Subscriptions' section contains a table with one subscription entry. The table has columns for 'Subscription details', 'Product', 'State', and 'Action'. The subscription entry shows 'Product Authorization subscription' with a 'Rename' link. Below the subscription name, there are fields for 'Name', 'Started on' (10/22/2021), 'Primary key', and 'Secondary key', each with a 'Show | Regenerate' link. The primary and secondary keys are masked with 'x's. At the bottom of the page, it says 'Powered by Azure API Management'.

Azure Video Analyzer for Media

Home APIs Developer Support Profile Sign out

User profile

Account details

Email	huangchuang0102@gmail.com
First name	chuang
Last name	huang
Registration date	10/21/2021

Subscriptions

Subscription details	Product	State	Action
Name	Product Authorization subscription	Authorization	Active
Started on			Cancel
10/22/2021			
Primary key	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	Show Regenerate	
Secondary key	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	Show Regenerate	

Powered by Azure API Management

Click on show to see the primary and secondary key

Set Up Serverless

Obtaining the code

There are two ways to get the code.

Download it from: <https://github.com/cyee10/box-videoindexer.git>

Then place it in your chosen directory; this will bypass the need to download git.

or

Download and install git: <https://git-scm.com/downloads>

Then open up the command console.

Navigate to the place where the code will reside using the command 'cd.'

Example: cd Desktop, recommend doing so for simplicity

To obtain the code from GitHub use command

- git clone <https://github.com/cyee10/box-videoindexer.git>

there should now be a folder called box-videoindexer at the chosen directory.

```
Microsoft Windows [Version 10.0.19043.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Chuang H>cd desktop

C:\Users\Chuang H\Desktop>git clone https://github.com/cyee10/box-videoindexer.git
Cloning into 'box-videoindexer'...
remote: Enumerating objects: 70, done.
remote: Counting objects: 100% (70/70), done.
remote: Compressing objects: 100% (65/65), done.
remote: Total 70 (delta 36), reused 23 (delta 5), pack-reused 0R
Receiving objects: 100% (70/70), 264.70 KiB | 1.24 MiB/s, done.
Resolving deltas: 100% (36/36), done.

C:\Users\Chuang H\Desktop>
```

Install required packages using NPM

To install all the packages we need, we need to change the directory to the box-videoindexer folder and do the installation inside the folder using following commands:

- **cd box-videoindexer**
- **npm install**

```
Microsoft Windows [Version 10.0.19043.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Chuang H>cd desktop
C:\Users\Chuang H\Desktop>git clone https://github.com/cyee10/box-videoindexer.git
Cloning into 'box-videoindexer'...
remote: Enumerating objects: 70, done.
remote: Counting objects: 100% (70/70), done.
remote: Compressing objects: 100% (65/65), done.
remote: Total 70 (delta 36), reused 23 (delta 5), pack-reused 0R
Receiving objects: 100% (70/70), 264.70 KiB | 1.24 MiB/s, done.
Resolving deltas: 100% (36/36), done.

C:\Users\Chuang H\Desktop>cd box-videoindexer
C:\Users\Chuang H\Desktop\box-videoindexer>npm install

> core-js@3.5.0 postinstall C:\Users\Chuang H\Desktop\box-videoindexer\node_modules\core-js
> node -e "try{require('./postinstall')}catch(e){}"

Thank you for using core-js ( https://github.com/zloirock/core-js ) for polyfilling JavaScript standard library!

The project needs your help! Please consider supporting of core-js on Open Collective or Patreon:
> https://opencollective.com/core-js
> https://www.patreon.com/zloirock

Also, the author of core-js ( https://github.com/zloirock ) is looking for a good job -)

> serverless@1.66.0 postinstall C:\Users\Chuang H\Desktop\box-videoindexer\node_modules\serverless
> node ./scripts/postinstall.js

  Serverless Framework successfully installed!
  To start your first project run 'serverless'.

npm WARN samuel-moon-box@1.0.0 No description
npm WARN samuel-moon-box@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.11 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.11: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.2 (node_modules\chokidar\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

added 1320 packages from 709 contributors and audited 1394 packages in 105.435s

21 packages are looking for funding
  run npm fund for details

found 274 vulnerabilities (5 low, 101 moderate, 165 high, 3 critical)
  run npm audit fix to fix them, or npm audit for details
```

Set up AWS credentials within serverless

To run the next command, you need to input your AWS KEY and SECRET without the ‘’.

- `serverless config credentials -o --provider aws --key 'YOUR_AWS_KEY' --secret 'YOUR_AWS_SECRET'`

To get AWS KEY and SECRET follow instruction from their page bellow:

<https://aws.amazon.com/premiumsupport/knowledge-center/create-access-key/>

```
Microsoft Windows [Version 10.0.19043.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Chuang H>cd desktop/box

C:\Users\Chuang H\Desktop\box>serverless config credentials -o --provider aws --key AKIAY3D32RYHGM5BEP5L --secret PJVMTe
dAIWQBb51tT1GSqHsRsjsJt2mfC5qcac+9
Serverless: Setting up AWS...

C:\Users\Chuang H\Desktop\box>
```

Since I already deploy the box-videoindexer, I rename the folder to box. However, first time deployment should stay with box-videoindexer

Deploying to AWS Lambda

Run following command

1. sls deploy

```
C:\Users\Chuang H\Desktop\box>sls deploy
Serverless: Packaging service...
Serverless: Excluding development dependencies...
Serverless: Uploading CloudFormation file to S3...
Serverless: Uploading artifacts...
Serverless: Uploading service box-video-skill.zip file to S3 (286.09 kB)...
Serverless: Validating template...
Serverless: Updating Stack...
Serverless: Checking Stack update progress...
.....
Serverless: Stack update finished..
Service Information
service: box-video-skill
stage: dev
region: us-east-1
stack: box-video-skill-dev
resources: 10
api keys:
  None
endpoints:
  ANY - https://n5xlth89n4.execute-api.us-east-1.amazonaws.com/dev/
functions:
  hello: box-video-skill-dev-hello
layers:
  None
Serverless: Deprecation warning: Resolution of lambda version hashes was improved with better algorithm, which will be used in next major release.
Switch to it now by setting "provider.lambdaHashingVersion" to "20201221"
More Info: https://www.serverless.com/framework/docs/deprecations/#LAMBDA_HASHING_VERSION_V2

Monitor Express APIs by route with the Serverless Dashboard: run "serverless"

C:\Users\Chuang H\Desktop\box>
```

The highlighted endpoint in red, is the environment variable `API_GATEWAY` in our AWS lambda function, it is also the Invocation URL for Box.com. Please copy it starting from the https and save it because it is important and you only get it once per deployment.


Configuring Box

Log in to Box developer console

Find desired app from the My Apps

Navigate to Configuration page and paste the endpoint obtain from lambda deployment to Invocation URL

Save the changes

Demo ? 

General Settings **Configuration** Security Keys App Diagnostics

Manage authentication methods and app permissions Save Changes

Client ID Client ID

Client ID for all transactional actions. [COPY](#)

Invocation URL Invocation URL

The location where your app payload will be sent.

Test Skills Event Select File

If a valid invocation URL is provided, testing your skills event will send a test event to the specified invocation URL for the file you pick.

If the invocation URL field is left blank or the specified invocation URL is invalid, testing the skills event will only return the body of the test event on this page.

File Extensions

Enter the file extensions that will be processed by this Skill. [Learn more](#). To process all file types, leave this field blank.

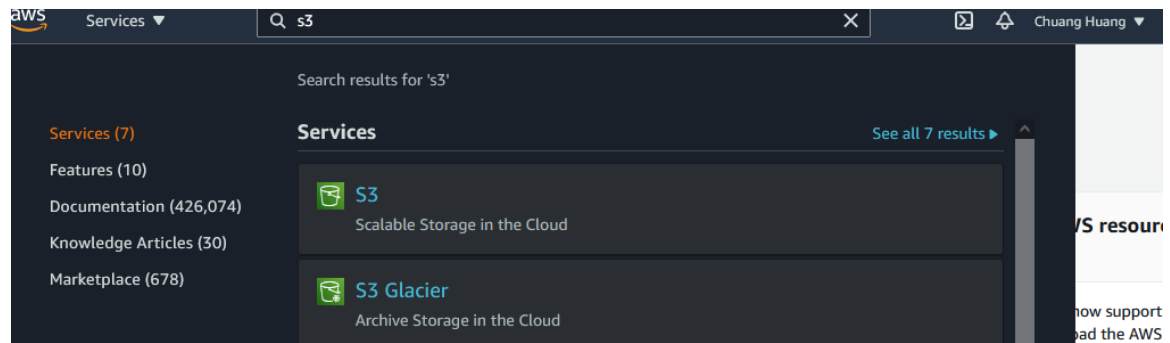
Get S3 Bucket Name

To avoid any trouble with duplicated names, S3 bucket name is auto generated through serverless

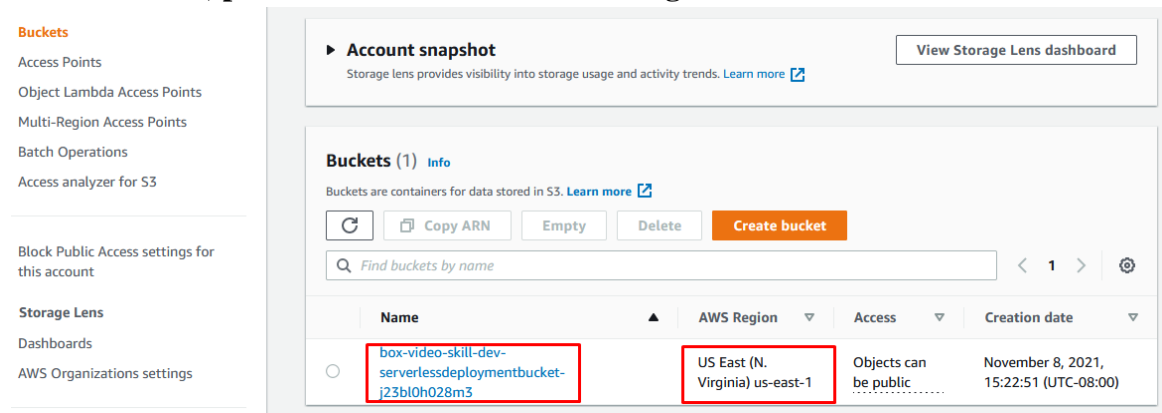
1. Log in to AWS website, you can either sign up a free trial account or ask sponsor for the account

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

2. Search for S3



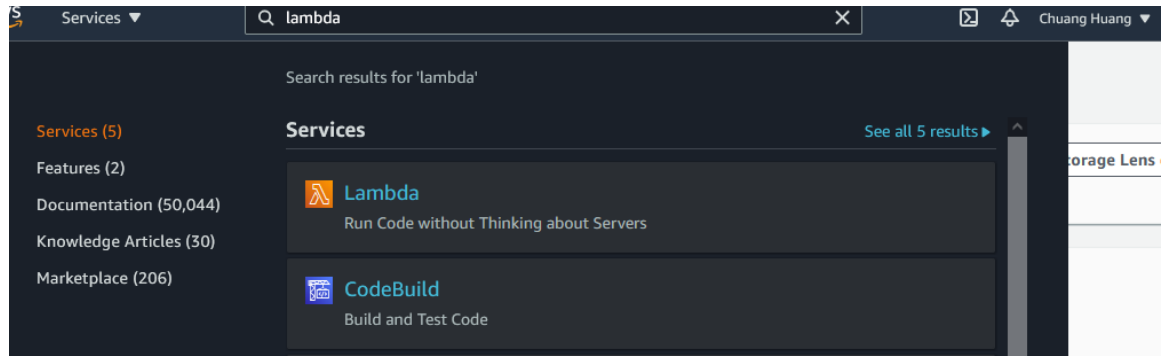
3. S3 bucket name, copy the name. S3 bucket itself is global but lambda function is not, please take a note of the AWS region



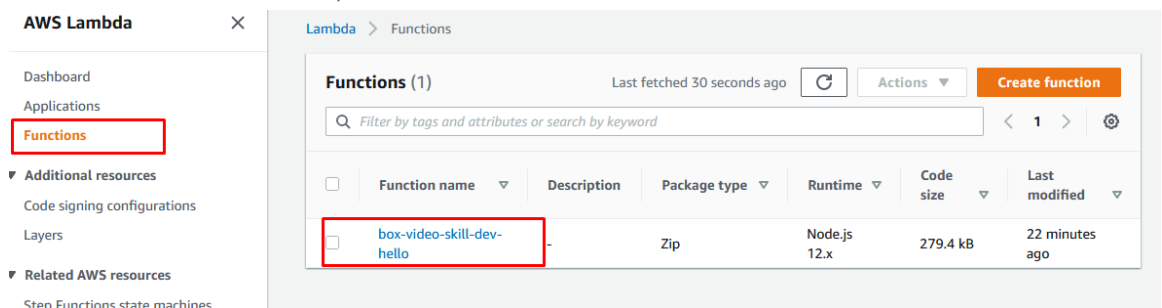
Set Up Environment Variables

This is the final step of this project, setting up all the environment variables we obtain to lambda function

1. Search for Lambda



2. Click functions in the left, find the function and click the function to see detail



3. Click on the configuration, then environment variables. For the environment variables, please refer back to environment variable checklist

