

# Private OpenHAB Cloud and Alexa Skill Tutorial

Updated 8/16/2020

## Introduction

The myopenhab.org service has been fairly reliable and I'm grateful that the OpenHAB Foundation maintains this service. Nonetheless, I've experienced a few certificate and connectivity errors I've felt that I'd like to have more control over this critical piece to my home automation setup, especially if the service were discontinued in the future. OpenHAB provides the code to set up your own OpenHAB Cloud instance and Alexa Skill and, although most of the work is already done for you, setting everything up isn't necessarily easy as a beginner. This tutorial consolidates my experience, resources used, and recommendations for setting this up.

*Disclaimer: My understanding of OpenHAB and Alexa is very superficial, at the end-user level. I struggled through this but eventually got it all working. I'm sure that there are errors and better ways, and I'd appreciate corrections and additions.*

## My environments:

- OpenHAB Cloud on AWS EC2 (t2.micro) instance running Ubuntu 20.04
- OpenHAB server on Raspberry Pi 3B running Openhabian 2.5.7-1

## OpenHAB Cloud Summary

1. Set up Virtual Private Server (I used AWS EC2, t2.micro, Ubuntu 20.04)
  - a. Set up Key Pair for SSH authentication
  - b. Set an elastic (static) IP address
  - c. Set up firewall / security rules to allow SSH (22), HTTP (80), HTTPS (443)
  - d. Register a domain name
  - e. Set up DNS (I used AWS Route 53)
2. Update and install dependencies
3. Install Node.js (ver 7.10.1) and NPM
4. Clone and Configure the OpenHAB-cloud code
5. Configure nginx (basic setup and testing)
6. Install letsencrypt and certbot for https certificate and auto renewal
7. Configure nginx for OpenHAB Cloud, start and test
8. Apply patch to prevent registration by others
9. Set up OpenHAB Cloud to run as a service

Setting up your VPS, firewall, domain, DNS, etc. is well documented and I won't cover those details. This tutorial will start with #2.

## Alexa Skill Summary

1. Create AWS Developer account
2. Create Access Key
3. Create Security Profile
4. Create new Security Role
5. Install Alexa Skills Kit Command Line Interface (ASK-CLI)
6. Configure and Deploy the Skill
7. Set up Account Linking and Login with Amazon

## **OpenHAB Cloud Setup**

Key overall references/resources:

<https://community.openhab.org/t/setup-your-own-openhab-cloud-myopenhab-server-instance/24716>

<https://github.com/openhab/openhab-cloud/blob/master/README.md#quickStart>

### **Set Up Your New Server**

*Instructions assume user is 'ubuntu'*

```
sudo apt update
```

```
sudo apt upgrade
```

```
sudo apt install build-essential redis-server mongodb nginx
```

### **Install Node.js (ver 7.10.1) and NPM**

<https://www.surrealcms.com/blog/how-to-upgrade-or-downgrade-nodejs-using-npm.html>

<https://stackoverflow.com/questions/47008159/how-to-downgrade-node-version>

*Note: OpenHAB Cloud is based on node.js version 7.10.1. When I attempted to install with a newer version of node I encountered errors. Downgrading to 7.10.1 seemed to solve it. There are several ways to do this, I used 'n'.*

```
sudo apt install nodejs
```

```
sudo apt install npm
```

```
node --version
```

```
sudo npm install -g n
```

```
sudo n 7.10.1
```

Restart shell (log out and back in)

```
node --version
```

should see 7.10.1

### **Clone and Configure the OpenHAB Cloud code**

```
cd /home/ubuntu
```

```
git clone https://github.com/openhab/openhab-cloud.git
```

```
cd /home/ubuntu/openhab-cloud
```

```
sudo npm install
```

*Note: I got a lot of deprecation warnings for packages that are outdated. Not knowing what specific versions may be required for OpenHAB-cloud, I left them at the specified versions. I'm hoping nothing breaks in the future.*

Two packages in the configuration (package.json) weren't found and can be installed manually:

```
sudo npm install bcryptjs
```

```
sudo npm Install qunitjs
```

Make a copy of config-production and call it config.json

```
cp config-production.json config.json
```

Change site name in config.json (1st line) to match your domain name. I used openhab.xxx.net where xxx is my domain name.

## Configure Nginx (basic setup and testing)

```
cd /etc/nginx/sites-available
cp default openhab.xxx.net
sudo nano openhab.xxx.net
edit server_name line to:
    server_name openhab.xxx.net;
```

Activate the site by making symbolic link from sites-enabled to sites-available:

```
sudo ln -s /etc/nginx/sites-available/openhab.xxx.net /etc/nginx/sites-enabled/openhab.xxx.net
```

Test the site by browsing to <http://openhab.xxx.net>, you should see the default nginx page

*Note: Later we will add the OpenHAB configuration to this file. You could start directly with the configuration text provided here at [github](https://github.com/openhab/openhab-cloud/blob/master/README.md#quickStart), but I wanted to first test that the basic nginx server was working without and complications from node.js*

<https://github.com/openhab/openhab-cloud/blob/master/README.md#quickStart>

## Install letsencrypt and certbot for https certificate auto renewal

Install Let's Encrypt:

```
sudo apt install letsencrypt
```

Install Certbot:

*The Certbot website is outstanding for tutorials, depending on your specific combination of OS and web server*

<https://certbot.eff.org/lets-encrypt/ubuntu-focal-nginx>

```
sudo apt update
sudo apt install software-properties-common
sudo add-apt-repository universe
sudo apt update
sudo apt install certbot python3-certbot-nginx
sudo certbot --nginx
```

Follow the prompts, select your openhab.xxx.net site and certbot will set up a certificate

Test your certificate and https at: <https://www.ssllabs.com/ssltest/analyze.html?d=openhab.xxx.net>

Test https by browsing to <https://openhab.xxx.net>

### **Configure nginx for OpenHAB Cloud, Start and Test**

```
cd /etc/nginx/sites-available  
nano openhab.xxx.net
```

You should now see the new lines that certbot added for port 443 and the ssl certificates in the server block. Check that the following lines exist, are uncommented, and point to the right place where your openhab-cloud files are located:

```
root /home/ubuntu/openhab-cloud  
index index.html index.htm  
server_name openhab.xxx.net
```

<https://github.com/openhab/openhab-cloud/blob/master/README.md#quickStart>

Add the following lines (from the openhab-cloud github) within the server block:

```
charset utf-8;
access_log /var/log/nginx/openhabcloud.org-access.log;
error_log /var/log/nginx/openhabcloud-error.log;
client_max_body_size 300m;

location /css {
    alias /home/ubuntu/openhab-cloud/public/css;
}
location /js {
    alias /home/ubuntu/openhab-cloud/public/js;
}
location /img {
    alias /home/ubuntu/openhab-cloud/public/img;
}
location /bootstrap {
    alias /home/ubuntu/openhab-cloud/public/bootstrap;
}
location /font-icons {
    alias /home/ubuntu/openhab-cloud/public/font-icons;
}
location /fonts {
    alias /home/ubuntu/openhab-cloud/public/fonts;
}
location /js-plugin {
    alias /home/ubuntu/openhab-cloud/public/js-plugin;
}
location /staff/js-plugin {
    alias /home/ubuntu/openhab-cloud/public/js-plugin;
}
location /downloads {
    alias /home/ubuntu/openhab-cloud/public/downloads;
}
location / {
    proxy_pass http://localhost:3000;
    proxy_redirect off;
    proxy_http_version 1.1;
    proxy_set_header Host $host;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto https;
}
error_page 500 502 503 504 /50x.html;
location = /50x.html {
    root html;
}
```

Restart nginx  
sudo systemctl restart nginx

Start the javascript app:  
cd /home/ubuntu/openhab-cloud  
node app.js

Browse to <https://openhab.xxx.net>, you should see the OpenHAB login/registration page  
Register with your UUID and secret

On your OpenHAB server, configure the Cloud Connector to use new URL. Go to Paper UI, Configuration, Services, IO, OpenHAB Cloud, Configure. Change the address to your new cloud server. Test connection and functionality.

## Apply Registration Patch

Apply this patch to prevent other users from registering on your private server.

<https://github.com/openhab/openhab-cloud/pull/57/commits/6ced6d326ee90002848df6fa6f9bb90260cbe2a7>

*Note: It seems that the latest version of the openhab-cloud code already contained most of the patch, except for the edits to app.js*

```
cd /home/ubuntu/openhab-cloud
```

```
nano app.js
```

Add the lines in green shown on the github page

## Set up openhab-cloud to Run as a Service

<https://github.com/openhab/openhab-cloud/blob/master/etc/openhabcloud.service>

```
nano /home/ubuntu/openhab-cloud/etc/openhabcloud.service
```

Several lines need to be changed to match your configuration:

```
User=ubuntu
```

```
Group=ubuntu
```

```
WorkingDirectory=/home/ubuntu/openhab-cloud
```

```
ExecStart=/usr/local/bin/node /home/ubuntu/openhab-cloud/app.js
```

*Note: I'm not sure why, but node appears in two locations on my server. **/usr/bin/node** and **/usr/local/bin/node**. I found that the service only runs when using **/usr/local/bin/node**. If you have issues with the service not starting, this is something to check.*

Test the service

```
sudo systemctl start /home/ubuntu/openhab-cloud/etc/openhabcloud.service
```

Enable the service on system startup

```
sudo systemctl enable /home/ubuntu/openhab-cloud/etc/openhabcloud.service
```

## **Alexa Skill Configuration**

Helpful links, although several outdated:

<https://community.openhab.org/t/setup-your-own-openhab-cloud-myopenhab-server-instance/24716/135>

<https://community.openhab.org/t/setup-your-own-openhab-cloud-myopenhab-server-instance/24716/59?u=dvzkul>

<https://developer.amazon.com/de-DE/docs/alexa/smarthome/smart-home-skill-migration-guide.html>

<https://heathpaddock.com/2017/02/02/making-amazon-echo-alexa-sing-to-openhab2/>

<https://github.com/auchter/haaska/issues/102>

<https://community.openhab.org/t/private-openhab-cloud-and-alexa/66226/7>

<https://github.com/openhab/openhab-alexa/blob/master/README.md>

<https://community.openhab.org/t/alexa-smart-home-skill-v3-is-now-live/79848>

<https://developer.amazon.com/blogs/post/Tx3CX1ETRZZ2NPC/Alexa-Account-Linking-5-Steps-to-Seamlessly-Link-Your-Alexa-Skill-with-Login-wit>

<https://community.openhab.org/t/oauth2-using-just-oh-rules-and-myopenhab-org/46680>

### **Create Access Key**

Go to AWS Management Console ([console.aws.amazon.com](https://console.aws.amazon.com)), at the top right select your Account Name, drop down to My Security Credentials

Under Your Security Credentials, open the Access Keys window, select Create New Access Key

Download the Key File and save the Access Key ID and Secret Access Key for later

### **Create Security Profile**

Go to AWS Developer Console ([developer.amazon.com](https://developer.amazon.com))

Sign in, go to Settings, Security Profiles.

Select Create a New Security Profile, give it a name like OpenHAB-Alexa

Save the Security Profile ID

*Note: I did not initially provide a Consent Privacy Notice URL but was later required to before linking the Alexa skill. I set up an alternate URL (i.e. [openhab.xxx.net/privacy.html](https://openhab.xxx.net/privacy.html)) on my nginx server with a barebones privacy statement and provided it here.*

Go to the Web Settings tab and copy the Client ID and Secret for later

### **Create a New Security Role**

*Note: This may not be necessary to do manually, it appears that the Skill deployment may do this automatically.*

Go to AWS Identify and Access Management (IAM) console ([console.aws.amazon.com/iam](https://console.aws.amazon.com/iam))

Under Access Management, select Roles, Create new Role

## Install Alexa Skills Kit Command Line Interface (ASK-CLI)

*Note: The Alexa Skills Kit (ASK) can be done on any computer, it is done to take the OpenHAB-cloud files and deploy it to an Alexa Skill in the cloud. I don't believe that ASK is necessary after this. I installed ASK on my local OpenHAB server (Raspberry PI 3B), not on my private OpenHAB cloud server, but I believe it could be done on either.*

ASK requires Node.js and NPM, which I did not have previously installed. Instructions:

```
Sudo apt install nodejs
```

```
Sudo apt install npm
```

Check Node version

```
Node --version
```

*Note: When installing nodejs in this manner (from the repository) v. 10.21.0 was installed. Later, when I tried installing the Alexa ASK-CLI I got error messages stating that NPM does not support this version of Node.js. You could install a newer version of Node from their repository, but I did it with 'n', which lets you easily manage what nodejs version you are working with.*

Install n and switch to newer version

```
sudo npm install -g n
```

```
sudo n 12.18.3
```

Go to home directory: `cd ~`

Clone the OpenHAB Alexa repository from Github:

```
git clone https://github.com/openhab/openhab-alexa.git
```

```
cd /home/openhabian/openhab-alexa
```

```
sudo npm install -g ask-cli@^1.0.0
```

```
sudo ask configure --no-browser
```

Paste the URL into your browser, log in with your AWS credentials, get the authorization code, and paste/type it back into ASK-CLI.

Select yes to link your AWS account to host your Alexa skills

Copy and paste the link into your browser, it should take you to the same Access Key ID and Secret Access Key that you created earlier. Enter these into the terminal.

## Configure and Deploy the skill

```
cd /home/openhabian/openhab-alexa/lambda/smarthome
cp config_sample.js config.js
nano config.js
```

In the `module.exports = {` block, uncomment the `baseUrl`, `user`, and `pass` lines and fill in with your credentials for your private OpenHAB Cloud.

```
baseUrl: 'https://openhab.xxx.net/rest'
```

Navigate to the directory where `skill.json` is

```
cd /home/openhabian/openhab-alexa
ask deploy
```

This creates the skill on Alexa. Copy the skill ID and Lambda ARN.

*Note: When running 'ask deploy' the first time I received an error message indicating that the /home/openhabian/.ask/cli\_config did not exist. It may have been since the first time I installed ask-cli that I omitted the @^1.0.0. My notes are slim on whether this fixed it or how I resolved this.*

## Set up Account Linking and Login with Amazon

*Note: The github page recommends doing this from the command line interface; I did it from the Alexa console instead and skipped step 3 of the Deployment Steps*

Go to the Alexa Developer Console <https://developer.amazon.com/alexa/console/ask>

You should see a new skill that was created by `ask deploy`

Go into the skill, check that `payload version = v3` and the default endpoint is the ARN that you copied earlier.

Go to Account Linking

Under Security Provider Information,

Auth Code Grant should be the only option

Your Web Authorization URI:

```
https://www.amazon.com/ap/oa?redirect_uri=https://pitangui.amazon.com/api/skill/link/...
```

Where the `redirect uri` is the one from below

Access Token URI: `https://api.amazon.com/auth/o2/token`

Your Client ID and Secret: From the Security Profile you set up earlier

Your Authentication Scheme: HTTP Basic

Click Add Scope, type 'profile'

Click Save, then navigate to 'Distribution' in the menu bar

Select English (US) or whatever your language is.

I changed the Public Name to `openHAB-Private` so I could distinguish it from the public OpenHAB Alexa skill.



## Final Steps

Go into your Alexa App on your phone, find the new skill, enable it, log in to Amazon

(Note, you need to use your AWS developer login, not your Amazon shopping account)

Discover new devices

Remove the previous OpenHAB skill

All seems to be working except for one bug I'm not able to resolve: My Alexa app only shows the new devices found from my new skill, however when setting up a Routine there are duplicates of each device. Not sure how to clear this cache or purge the old devices.