Secure Starter Kit Cloud Connect Installation & Setup Guide

Date: March 03, 2021 | Version 1.1

CONTENTS

1 INTRODUCTION

1.1 Purpose of the Document

The Cloud Connect Installation & Setup Guide provides an overview of the AWS services required to run the demo's provided in the Security Starter Quick Start Guides, as well as detailed instructions to setup and configure those required services. Each of these services **MUST** be setup and configured (only once), prior to running the demo's outlined in the Security Starter Quick Start Guides.

1.2 Prerequisites, Background information & AWS Cloud Services Descriptions

1. AWS Account Management Console – the user will need to create their own AWS Account and is used as the basis for the configuration of the other services required to run the demo's provided in the Security Starter Kits. The creation of an account provides the following access and feature;
    - Discover and experiment with over 150 AWS services, many of which you can try for free.
    - Build your cloud-based applications in any AWS data center throughout the world.
    - Manage and monitor users, service usage, health, and monthly billing.
    - Get in-console help from AWS Support.
    - *Link to create AWS Account;* https://portal.aws.amazon.com/billing/signup#/start

2. AWS EC2 Instance Service – https://aws.amazon.com/ec2

    Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment.

    - *The output of the setup and configuration of the EC2 instance will provide the user with URL and Login credentials required to run their own instance of the Security Starter Kit Cloud Connect Tool.*

3. AWS Relational Database Service (Amazon RDS) – https://aws.amazon.com/rds/

    Makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching and backups. It frees you to focus on your applications so you can give them the fast performance, high availability, security and compatibility they need.

    Amazon RDS is available on several database instance types - optimized for memory, performance or I/O - and provides you with six familiar database engines to choose from, including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, and SQL Server. You can use the AWS Database Migration Service to easily migrate or replicate your existing databases to Amazon RDS.

    - *MySQL is employed as the database instance type when configuring Amazon RDS service.*

4. Docker Hub - Cloud-based application registry and development team collaboration services.
   https://www.docker.com/
   https://hub.docker.com/

   Docker Hub is the world's largest repository of **container images** with an array of content sources including container community developers, open source projects and independent software vendors (ISV) building and distributing their code in containers. Users get access to free public repositories for storing and sharing images or can choose subscription plan for private repos.

   - *Docker is the repository service used to store source code for our web-based, open-source "Security Starter Kit Cloud Connect Tool". The user will need to update, configure and build an "Image" from the source code stored on Docker Hub using their specific AWS Account and AWS Service credentials. These instructions are provided below in the document.*

5. AWS OTA Role Access - https://docs.aws.amazon.com/freertos/latest/userguide/create-ota-user-policy.html

   When you create an OTA update, the OTA Update Manager service creates an AWS IoT job to notify your devices that an update is available. The OTA demo application runs on your device and creates a FreeRTOS task that subscribes to notification topics for AWS IoT jobs and listens for update messages. When an update is available, the OTA Agent publishes requests to AWS IoT and receives updates using the HTTP or MQTT protocol, depending on the settings you chose. The OTA Agent checks the digital signature of the downloaded files and, if the files are valid, installs the firmware update. If you don't use the FreeRTOS OTA Update demo application, you must integrate the OTA Agent library into your own application to get the firmware update capability.

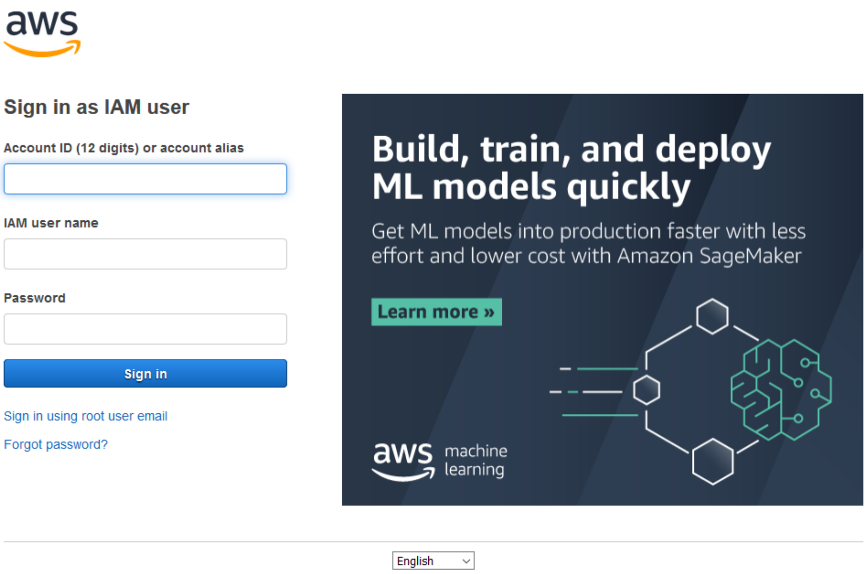
   - *OTA setup and configuration is listed in Section 5 of the SSK Cloud Connect Users Guide.*
   - *These steps do not need to be completed to run the demo outlined in the Quick Start Guide, but will need to be configured in order to perform OTA firmware updates from within AWS Cloud Services.*

2 AWS ACCOUNT CREATION & SETUP EC2 SERVICE

2.1 Login or Create your AWS Account

Note: If the User does not have an AWS Account, you will need to create one and this is used as the basis for the configuration of the other services required to run the demo's provided in the Security Starter Kits.

Login URL: https://aws.amazon.com/console/



Sign in as IAM user

Account ID (12 digits) or account alias

IAM user name

Password

Sign in

Sign in using root user email

Forgot password?

Build, train, and deploy ML models quickly

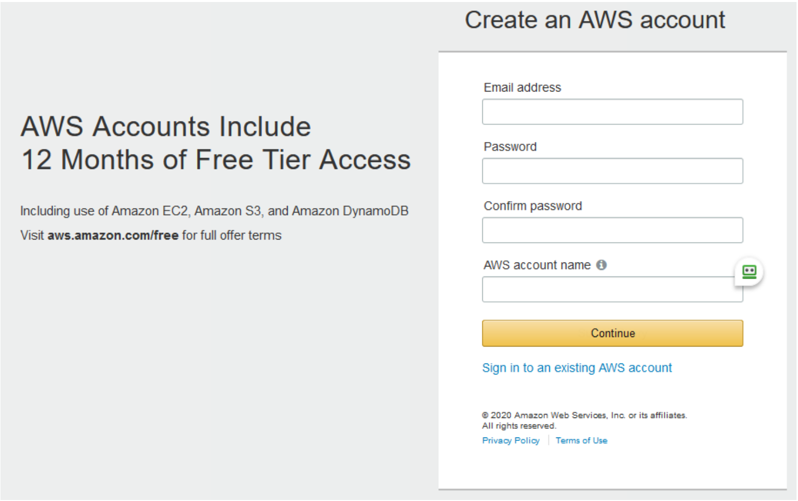
Get ML models into production faster with less effort and lower cost with Amazon SageMaker

Learn more »

aws machine learning

English

Figure 1: Login page



Create an AWS account

AWS Accounts Include 12 Months of Free Tier Access

Including use of Amazon EC2, Amazon S3, and Amazon DynamoDB

Visit **aws.amazon.com/free** for full offer terms

Email address

Password

Confirm password

AWS account name

Continue

Sign in to an existing AWS account

© 2020 Amazon Web Services, Inc. or its affiliates.
All rights reserved.
Privacy Policy | Terms of Use

Figure 2: Create New Account page

2.2 AWS EC2 Instance Service

- Go to AWS Console >> Services >> Select EC2 (Under Compute section).

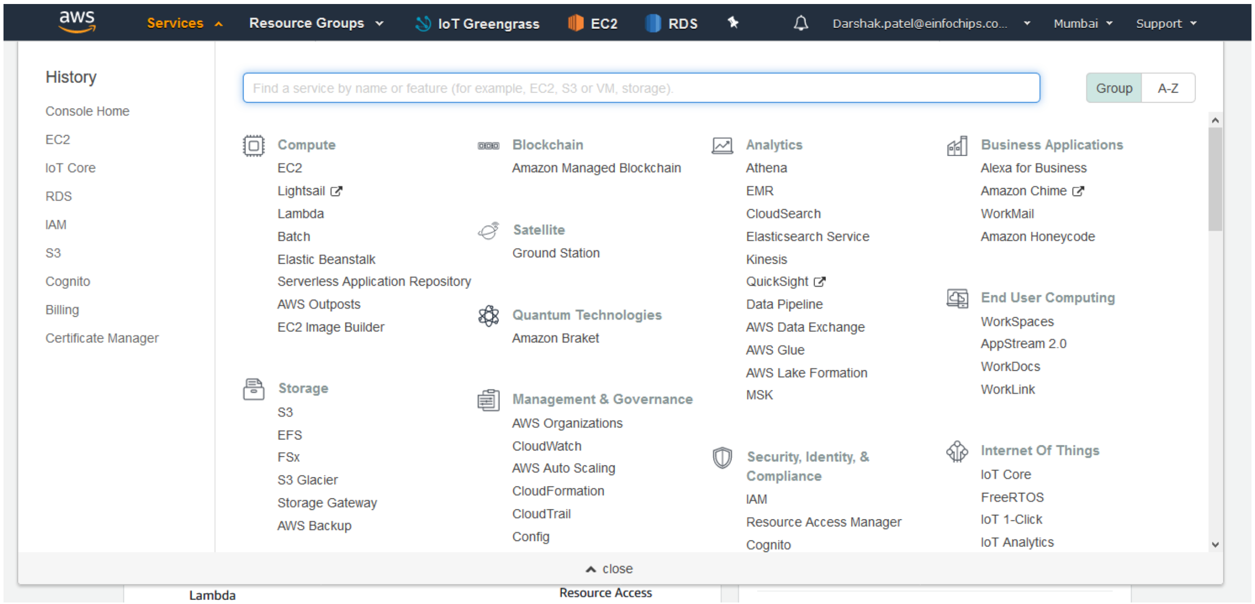
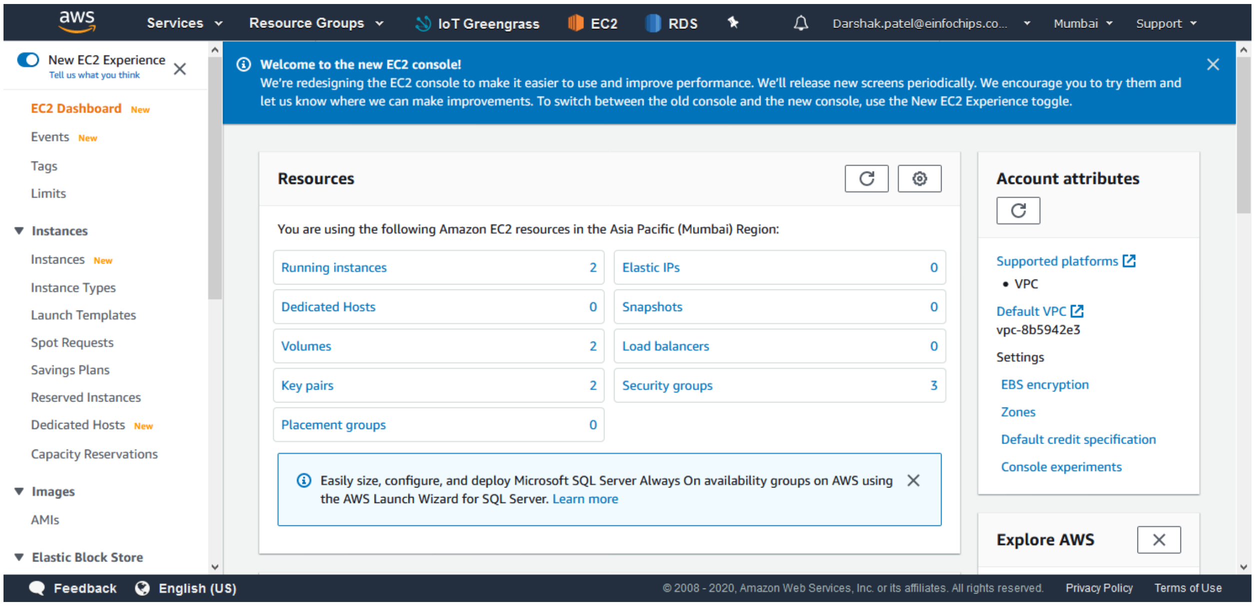


Figure 2: Select EC2 Instance

2.3 EC2 Dashboard

- Go to Instances >> Instances Click on it.



Welcome to the new EC2 console!
We're redesigning the EC2 console to make it easier to use and improve performance. We'll release new screens periodically. We encourage you to try them and let us know where we can make improvements. To switch between the old console and the new console, use the New EC2 Experience toggle.

Resources

You are using the following Amazon EC2 resources in the Asia Pacific (Mumbai) Region:

Running instances	2	Elastic IPs	0
Dedicated Hosts	0	Snapshots	0
Volumes	2	Load balancers	0
Key pairs	2	Security groups	3
Placement groups	0		

Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS using the AWS Launch Wizard for SQL Server. Learn more

Account attributes

Supported platforms
- VPC

Default VPC
vpc-8b5942e3

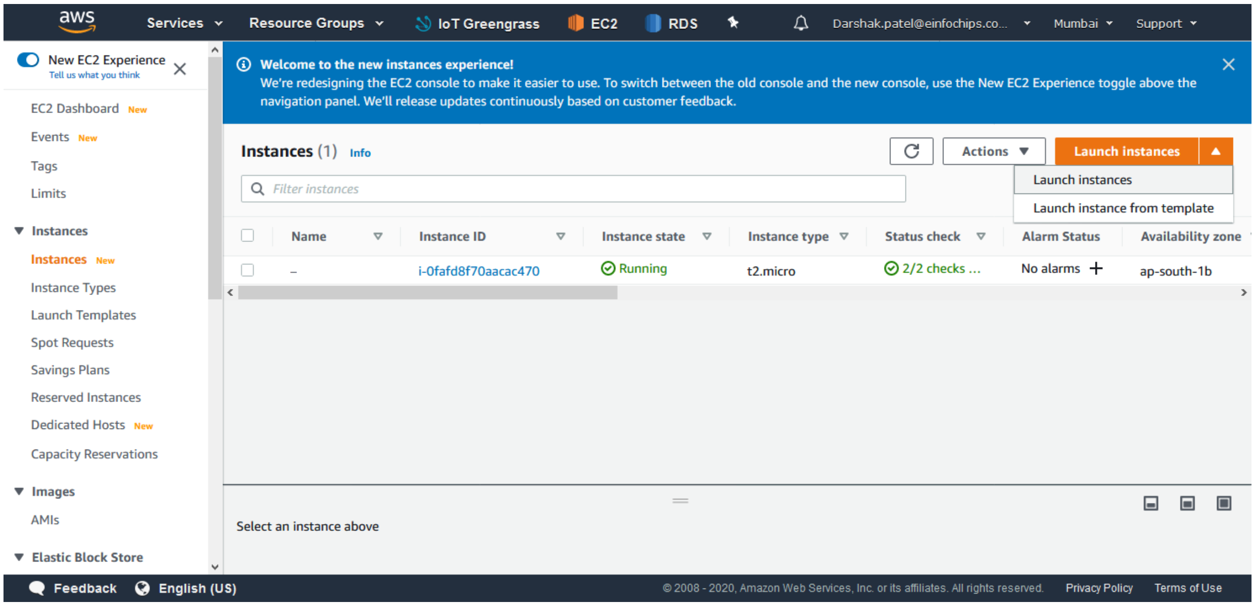
Settings
EBS encryption
Zones
Default credit specification
Console experiments

Explore AWS

Figure 2: List EC2 Dashboard

2.4 Creating an EC2 Instance

Step 1: Click on Launch instances (Top right corner)



Welcome to the new instances experience!
We're redesigning the EC2 console to make it easier to use. To switch between the old console and the new console, use the New EC2 Experience toggle above the navigation panel. We'll release updates continuously based on customer feedback.

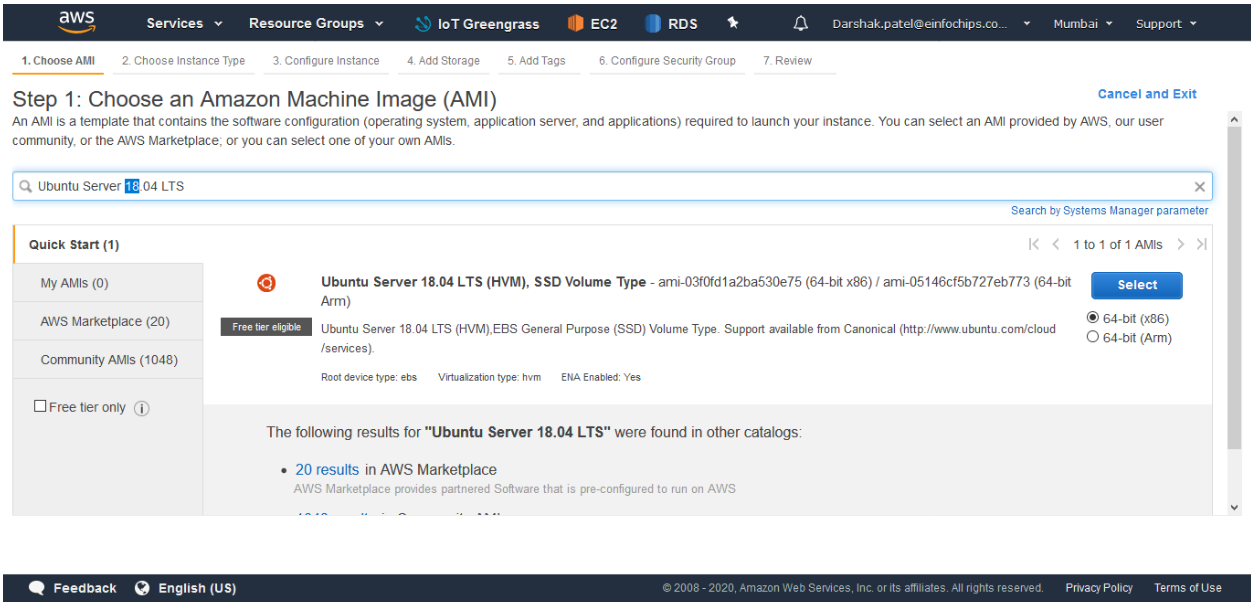
Instances (1) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availability zone
–	i-0fafd8f70aacac470	Running	t2.micro	2/2 checks ...	No alarms	ap-south-1b

Select an instance above

Figure 3: Launch EC2 Instance

- Choose an Amazon Machine Image
  Search "Ubuntu Server 18.04 LTS" in textbox then press select button.



Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Ubuntu Server 18.04 LTS

☐ Free tier only

Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-03f0fd1a2ba530e75 (64-bit x86) / ami-05146cf5b727eb773 (64-bit Arm)

Ubuntu Server 18.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

● 64-bit (x86)
○ 64-bit (Arm)

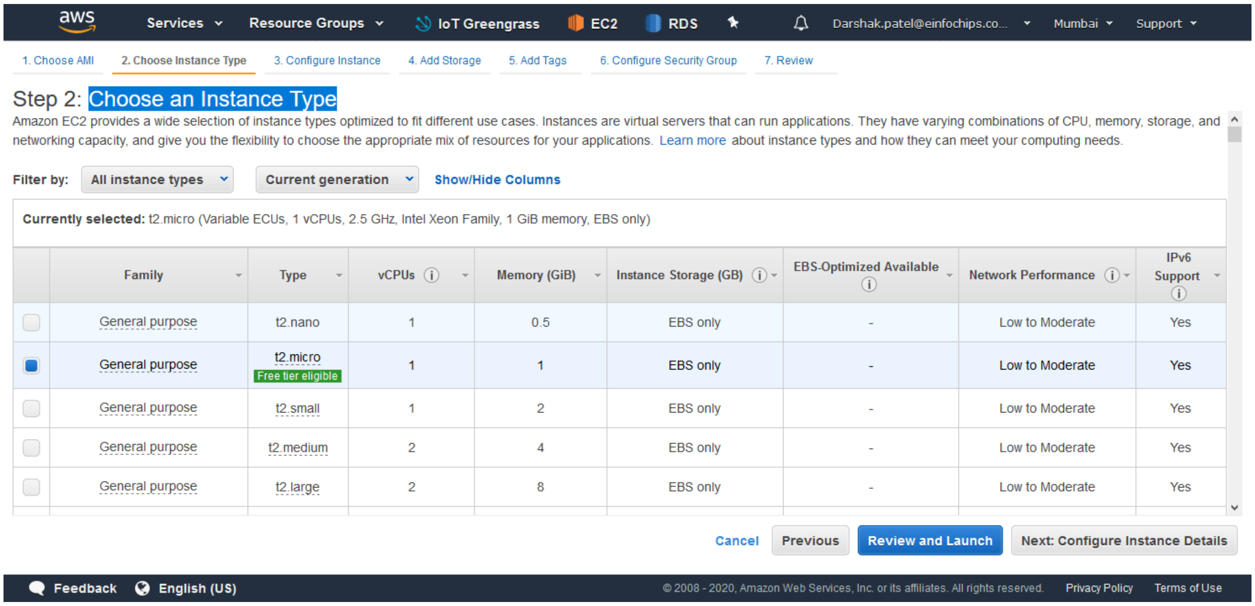
The following results for **"Ubuntu Server 18.04 LTS"** were found in other catalogs:

- 20 results in AWS Marketplace
  AWS Marketplace provides partnered Software that is pre-configured to run on AWS

Figure 4: Ubuntu AMI

Step 2: Choose an Instance Type (Change as per your performance requirement)

- Click on Next: Configure Instance details



Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance types | Current generation | Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

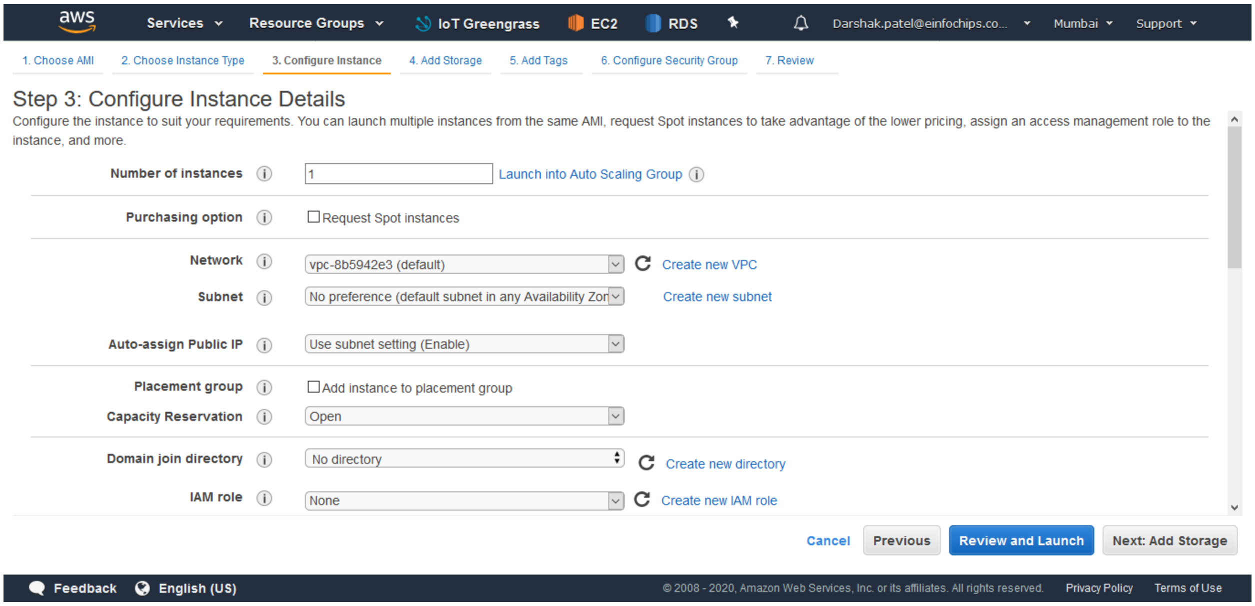
	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
☐	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
☑	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes

Cancel
Previous
Review and Launch
Next: Configure Instance Details
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Figure 5: Configure EC2 Instance Type

Step 3: Configure Instance Details (Don't alter anything if don't needed)

- Click on Next: Add Storage



Step 3: Configure Instance Details

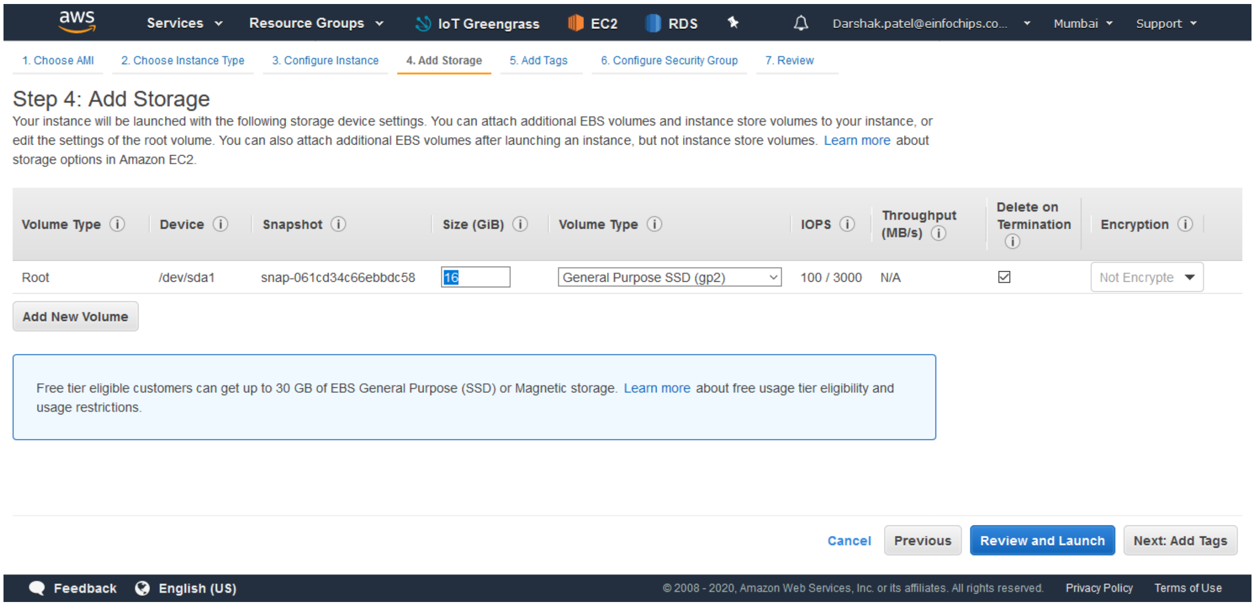
Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	1 Launch into Auto Scaling Group
Purchasing option	☐ Request Spot instances
Network	vpc-8b5942e3 (default) Create new VPC
Subnet	No preference (default subnet in any Availability Zor Create new subnet
Auto-assign Public IP	Use subnet setting (Enable)
Placement group	☐ Add instance to placement group
Capacity Reservation	Open
Domain join directory	No directory Create new directory
IAM role	None Create new IAM role

Figure 6: Configure Instance details

Step 4: Add Storage

- Change size to 16 GB (Default 8 GB) then press Next: Add Tags



Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-061cd34c66ebbdc58	16	General Purpose SSD (gp2)	100 / 3000	N/A	☑	Not Encrypte

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.

Figure 7: Add Storage

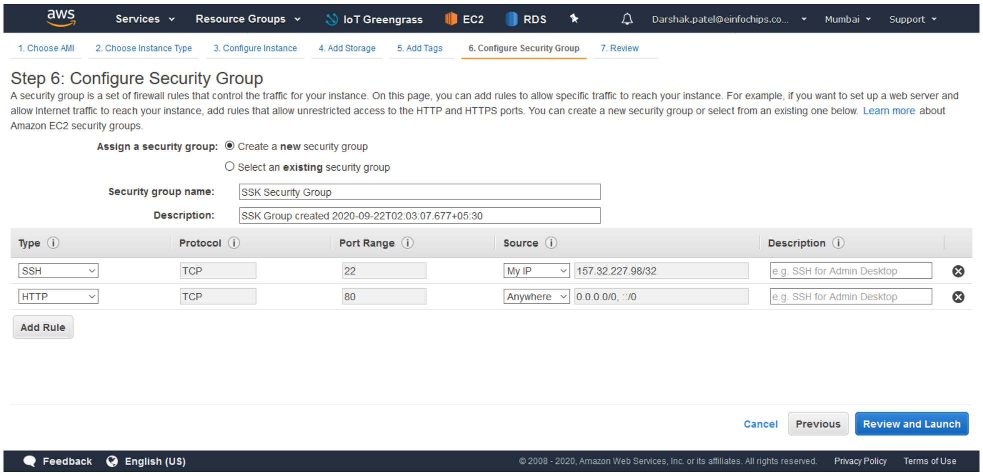
Step 5: Add Tags (Don’t do anything)

- Press Next: Configure Security Group

Step 6: Configure Security Group

- Fill up Security group Name: SSK Security Group (Also add description)
- Then press “Review and Launch”

Note: Ensure that both SSH and HTTP are listed as “type” below, otherwise click “Add Rule” to enable those services.



Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ● Create a **new** security group
○ Select an **existing** security group

Security group name: SSK Security Group

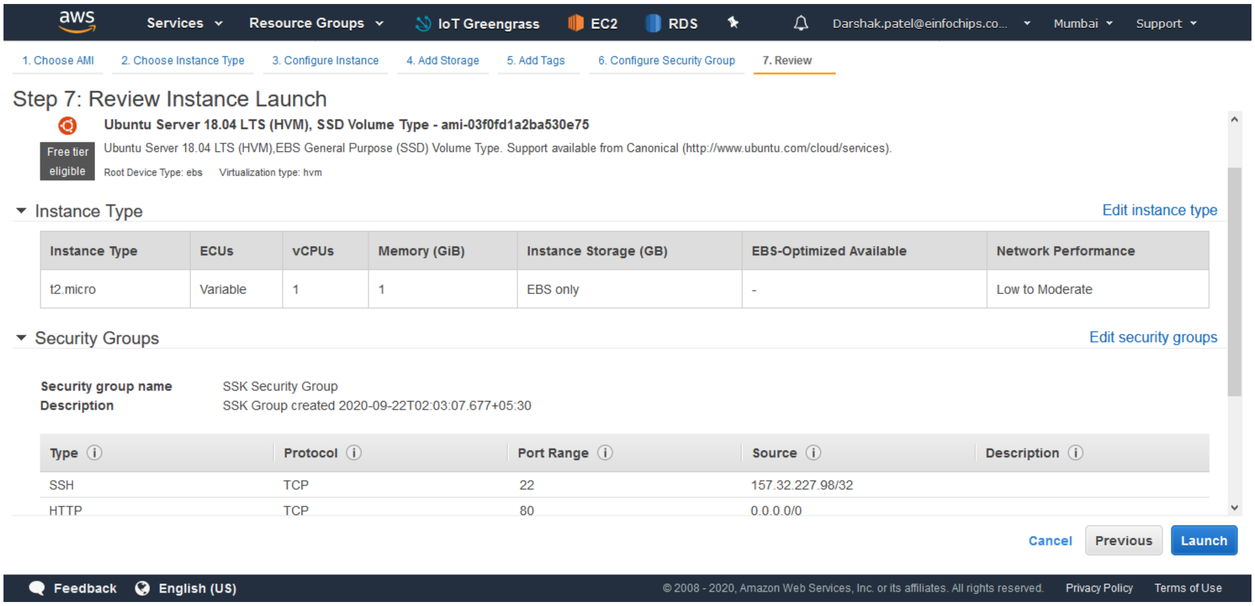
Description: SSK Group created 2020-09-22T02:03:07.677+05:30

Type	Protocol	Port Range	Source		Description
SSH	TCP	22	My IP	157.32.227.98/32	e.g. SSH for Admin Desktop
HTTP	TCP	80	Anywhere	0.0.0.0/0, ::/0	e.g. SSH for Admin Desktop

Figure 8: Configure Security group

Step 7: Review Instance Launch

- Press Launch



Step 7: Review Instance Launch

Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-03f0fd1a2ba530e75

Free tier eligible

Ubuntu Server 18.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Root Device Type: ebs Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups

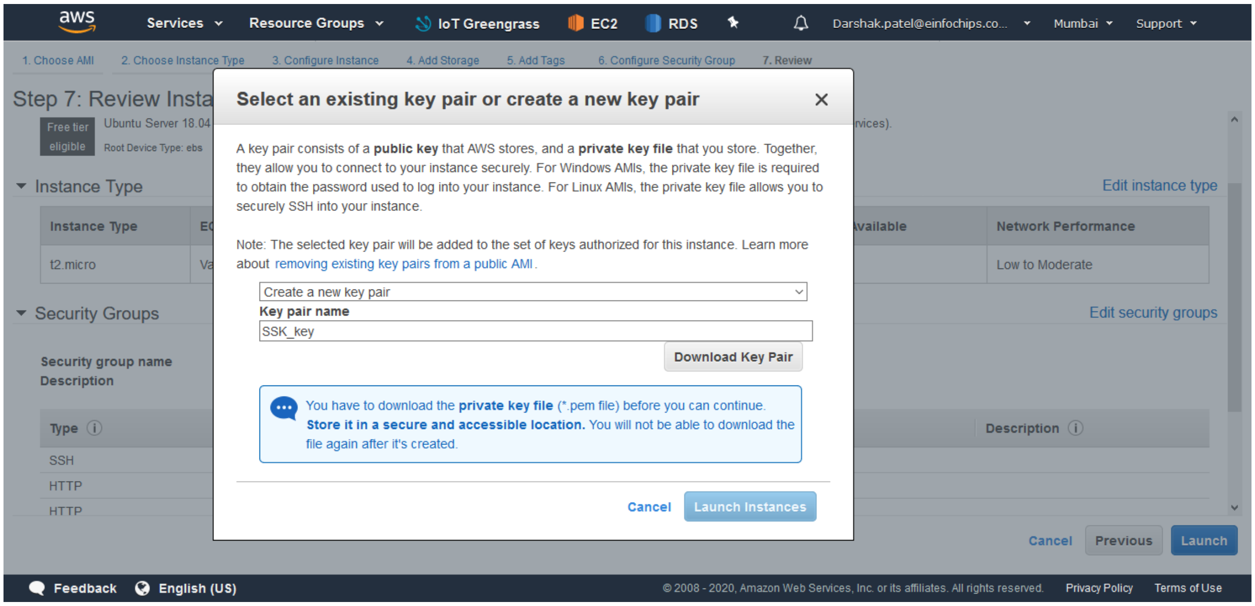
Security group name SSK Security Group
Description SSK Group created 2020-09-22T02:03:07.677+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	157.32.227.98/32	
HTTP	TCP	80	0.0.0.0/0	

Figure 9: Review Instance

Step 8: Create New key pair

- Select “Create a new key pair” then name “SSK_Key”



Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Create a new key pair

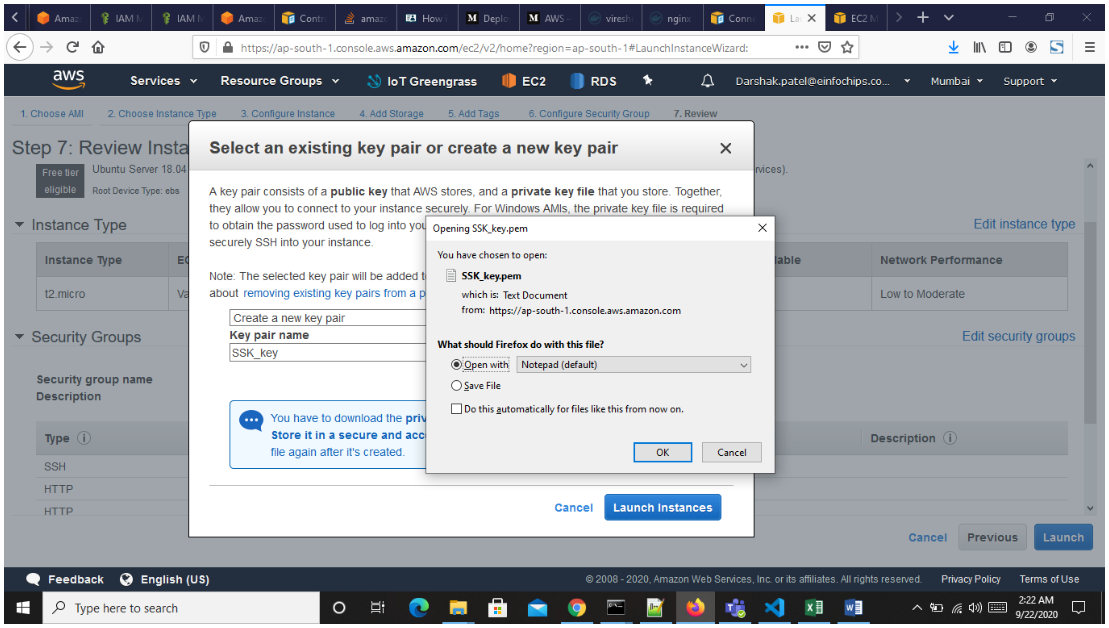
Key pair name
SSK_key

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Figure 10: Configure key

Step 9: Download key pair (To Connect EC2 Instance)

- Keep Certificate key file at secure place which will be used to connect EC2 instance.
- Then press “launch Instance”.



Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into you securely SSH into your instance.

Note: The selected key pair will be added t about removing existing key pairs from a p

Create a new key pair

Key pair name

SSK_key

You have to download the **priv** **Store it in a secure and acc** file again after it's created.

Opening SSK_key.pem

You have chosen to open:

SSK_key.pem

which is: Text Document

from: https://ap-south-1.console.aws.amazon.com

What should Firefox do with this file?

● Open with Notepad (default)

○ Save File

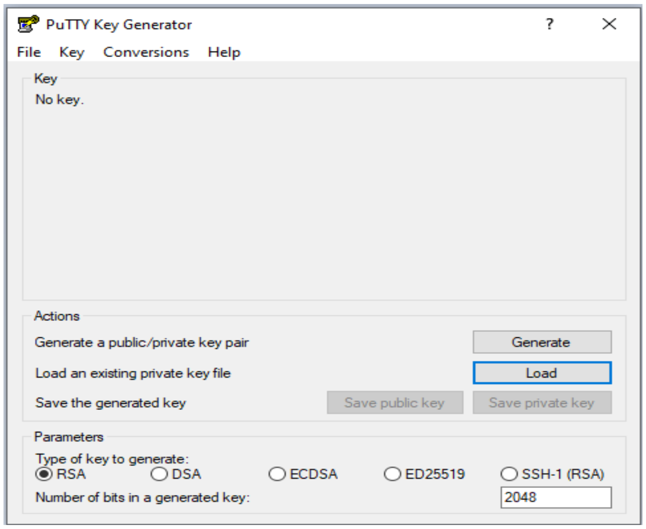
□ Do this automatically for files like this from now on.

Figure 11: Download key

2.5 Convert key to Putty Format

Step 1: Convert SSK_key.pem file to SSK_key.ppk (Using Putty)

Open PuTTYgen (From Windows) press Load button.



PuTTY Key Generator

Key

No key.

Actions

Generate a public/private key pair — Generate

Load an existing private key file — Load

Save the generated key — Save public key, Save private key

Parameters

Type of key to generate:

● RSA ○ DSA ○ ECDSA ○ ED25519 ○ SSH-1 (RSA)

Number of bits in a generated key: 2048

Figure 1: Convert PPK file

- It will ask for file to choose, here you'll need to provide SSK_key.pem file (select all file format)
- After successfull loading of key it will popup the successfully loaded key
- Press "Save private key". (Ignore passphrase warning)
- Name the file " SSK_key" and Save file along with ppk file

Ref Link:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html#putty-private-key

2.6 Configure Putty

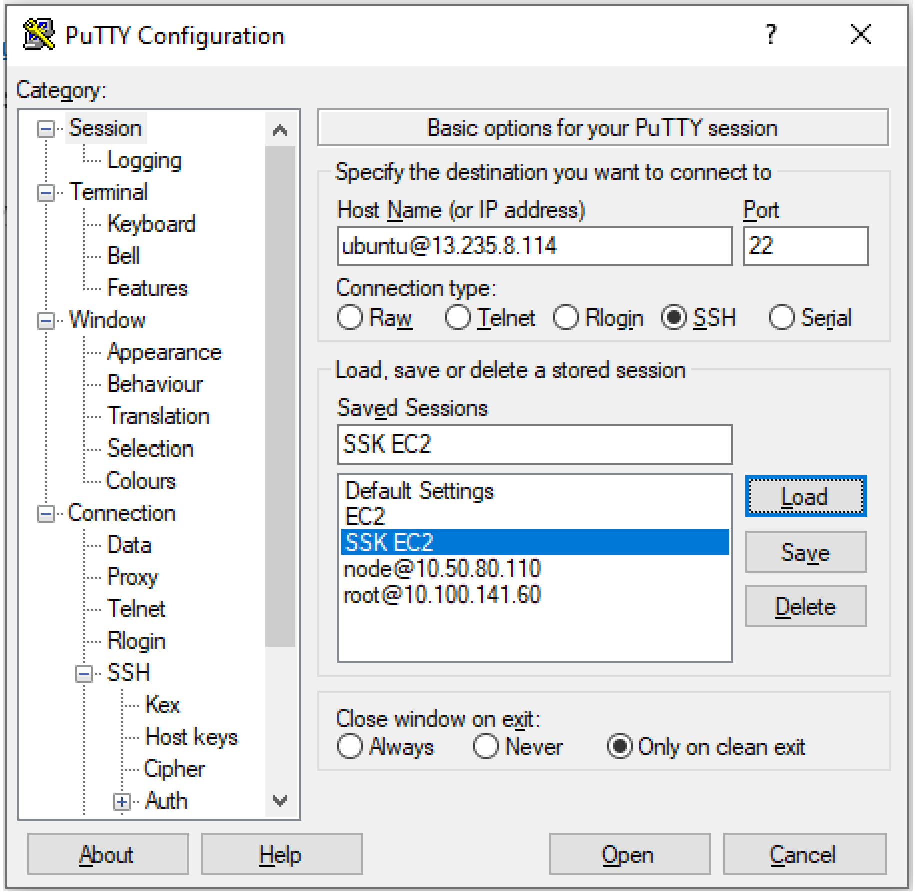
- Open Putty and save the session with following details

  Host Name: ubuntu@<host ip address> (Host Ip address can be obtained from EC2 instance)

  i.e.

  Host Name: ubuntu@13.235.8.114

  Session Name: SSK EC2



PuTTY Configuration

Category:

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): ubuntu@13.235.8.114

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions: SSK EC2

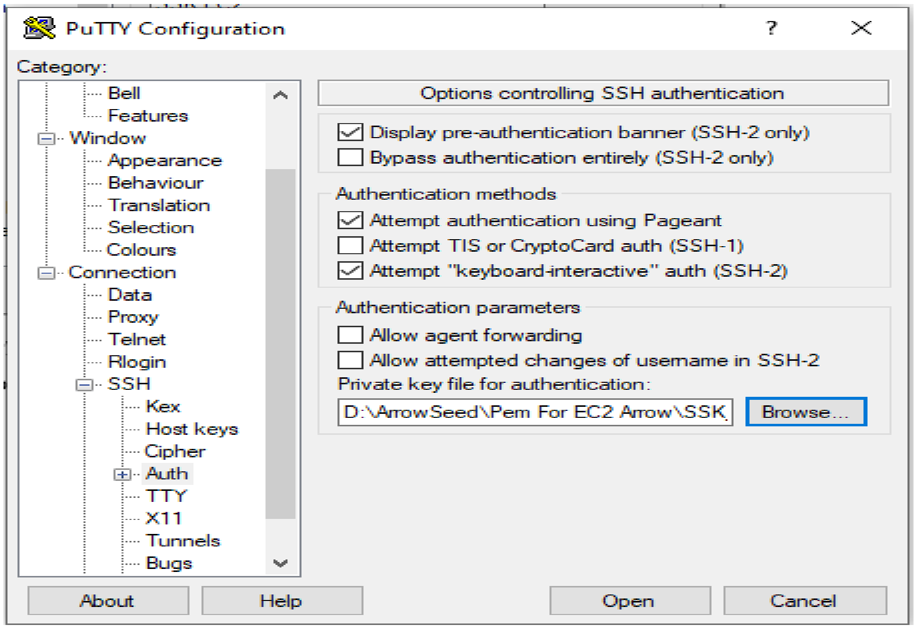
Default Settings
EC2
SSK EC2
node@10.50.80.110
root@10.100.141.60

Close window on exit: ○ Always ○ Never ● Only on clean exit

Figure 1: Configure putty

To configure key

- Go to Connection >> SSH >> Auth >> Select private key
- Then again save it and press open button.



Options controlling SSH authentication

☑ Display pre-authentication banner (SSH-2 only)
□ Bypass authentication entirely (SSH-2 only)

Authentication methods

☑ Attempt authentication using Pageant
□ Attempt TIS or CryptoCard auth (SSH-1)
☑ Attempt "keyboard-interactive" auth (SSH-2)

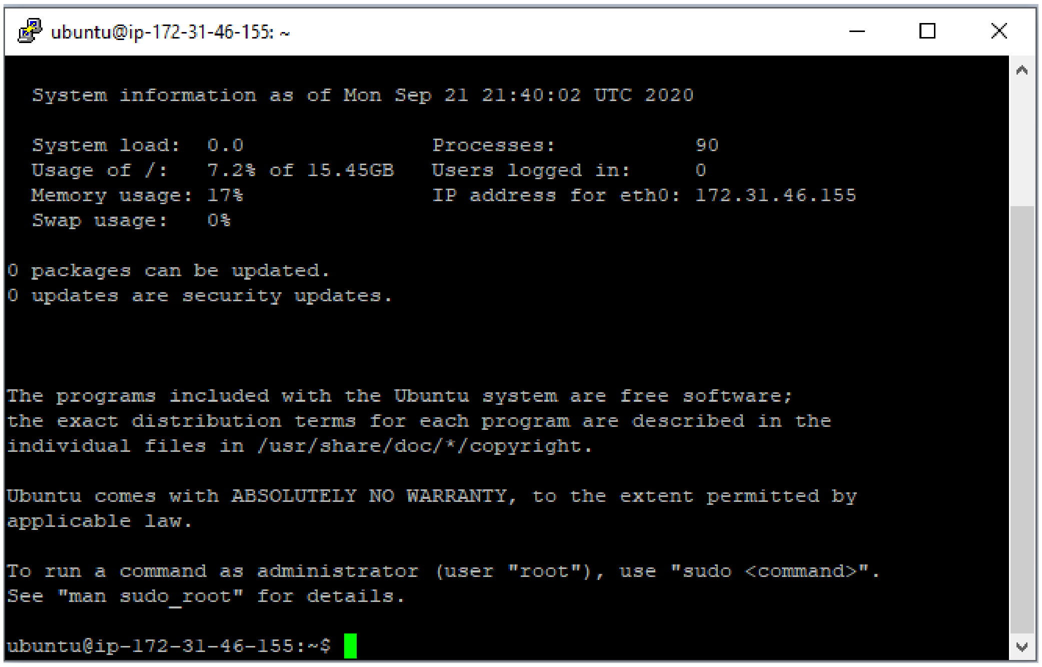
Authentication parameters

□ Allow agent forwarding
□ Allow attempted changes of username in SSH-2

Private key file for authentication:
D:\ArrowSeed\Pem For EC2 Arrow\SSK

Figure 2: add key to putty

- Here you can now connect to the AWS EC2 Instance.



```
  System information as of Mon Sep 21 21:40:02 UTC 2020

  System load:  0.0                Processes:           90
  Usage of /:   7.2% of 15.45GB    Users logged in:     0
  Memory usage: 17%                IP address for eth0: 172.31.46.155
  Swap usage:   0%

0 packages can be updated.
0 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-46-155:~$
```

Figure 3: Connected AWS

3 INSTALLING DOCKER ON EC2

3.1 Execute below command

Step 1: Update your existing list of packages

$ sudo apt-get update

Step 2: Next, install a few prerequisite packages which will let apt use packages over HTTPS:

$ sudo apt-get install apt-transport-https ca-certificates curl gnupg-agent software-properties-common

Step 3: Add Docker’s official GPG key:

$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

Step 4: Add the Docker repository to APT sources

$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"

Step 5: Update the package database with the Docker packages

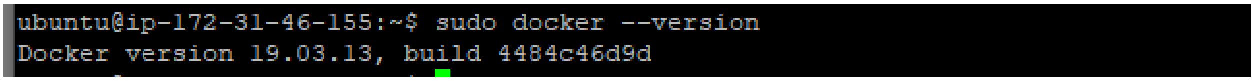
$ sudo apt-get update

Step 6: Install Docker

$ sudo apt-get install docker-ce docker-ce-cli containerd.io

Step 7: To verify installation

$sudo docker –version



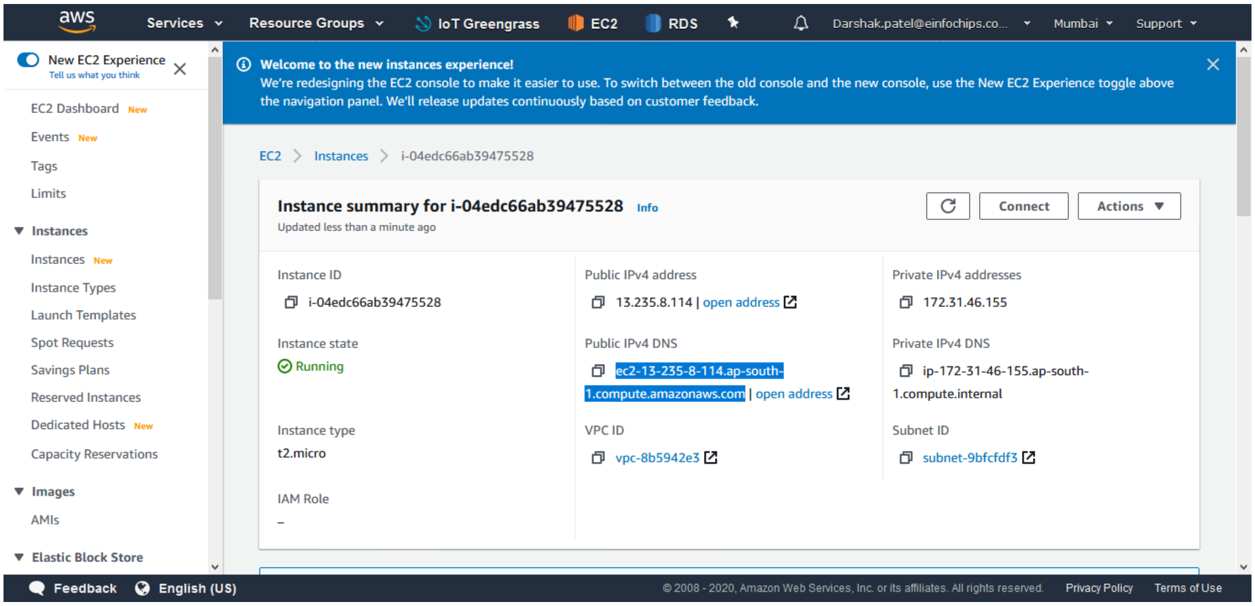
```
ubuntu@ip-172-31-46-155:~$ sudo docker --version
Docker version 19.03.13, build 4484c46d9d
```

Figure 1: Docker version

4 CONFIGURATION OF EC2 INSTANCE, RDS SERVICE AND SQL DATABASE

4.1 Application access

After executing docker run command to access application using following page:



Welcome to the new instances experience!
We're redesigning the EC2 console to make it easier to use. To switch between the old console and the new console, use the New EC2 Experience toggle above the navigation panel. We'll release updates continuously based on customer feedback.

Instance summary for i-04edc66ab39475528 Info
Updated less than a minute ago

Instance ID: i-04edc66ab39475528
Public IPv4 address: 13.235.8.114 | open address
Private IPv4 addresses: 172.31.46.155
Instance state: Running
Public IPv4 DNS: ec2-13-235-8-114.ap-south-1.compute.amazonaws.com | open address
Private IPv4 DNS: ip-172-31-46-155.ap-south-1.compute.internal
Instance type: t2.micro
VPC ID: vpc-8b5942e3
Subnet ID: subnet-9bfcfdf3
IAM Role: –

Figure 1: Application access

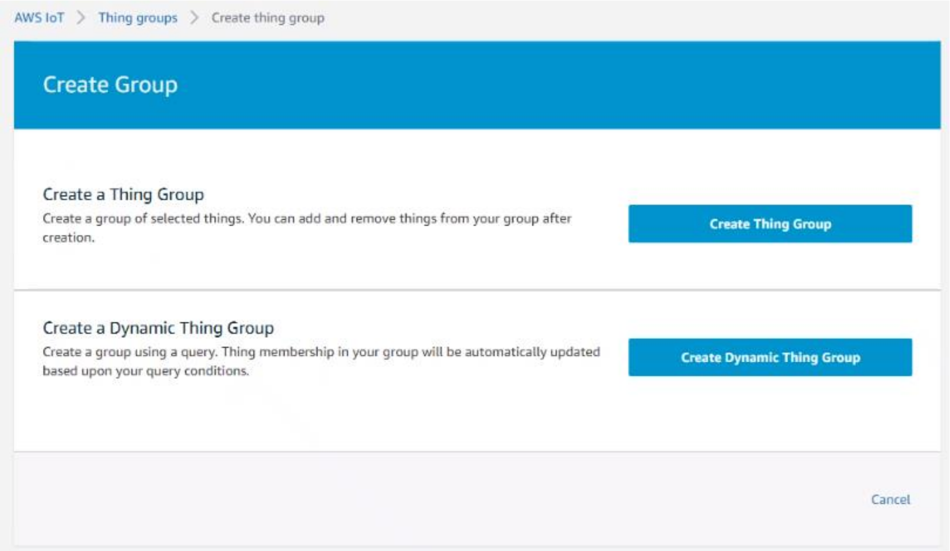
Make note of the Public IPv4 DNS address provided and it needs to be in the following format, with a leading HTTP:// as shown below;

Public Access URL: http://ec2-13-235-8-114.ap-south-1.compute.amazonaws.com/

4.2 Application – “Allow” or “Deny” listing

1. Create Thing group “BlacklistGroup” and create one default policy “blacklist-policy” then attach policy to thing group.

    - To create ‘Thing group’ : navigate to IoT Core → Manage → Thing groups → Create



AWS IoT > Thing groups > Create thing group
Create Group
Create Thing Group
Create Dynamic Thing Group
Cancel

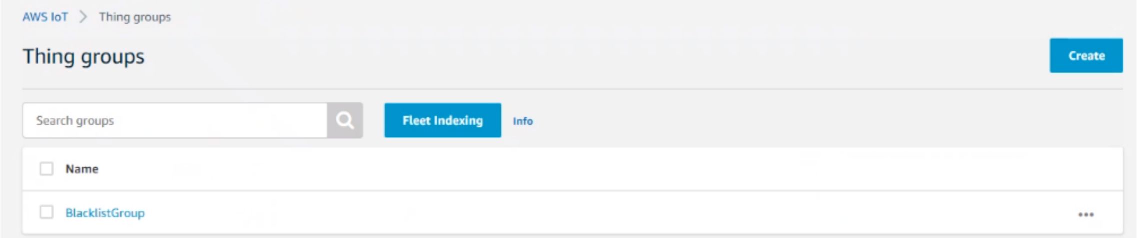
Create a Thing Group

Create a group of selected things. You can add and remove things from your group after creation.

Create a Dynamic Thing Group

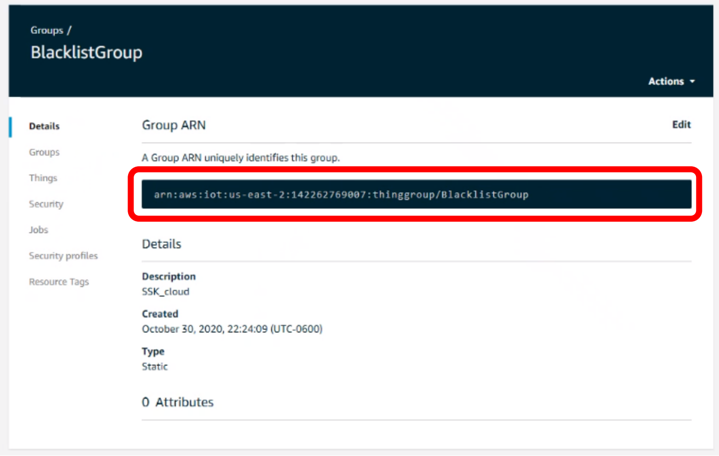
Create a group using a query. Thing membership in your group will be automatically updated based upon your query conditions.

    - Provide a name to your Thing Group, like what is shown below.



Name
BlacklistGroup

    - Click on “BlacklistGroup” (or whatever name you gave it) and make note of your Group ARN listed below;



Group ARN

A Group ARN uniquely identifies this group.

```
arn:aws:iot:us-east-2:142262769007:thinggroup/BlacklistGroup
```

Details

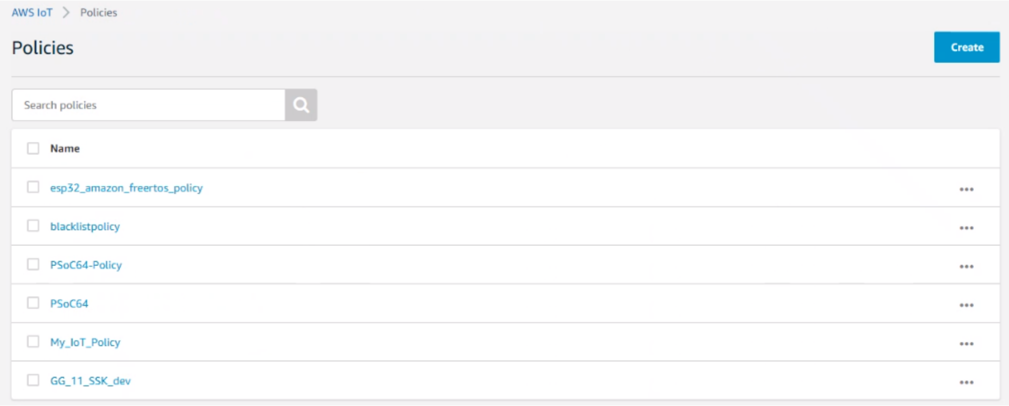
Description
SSK_cloud

Created
October 30, 2020, 22:24:09 (UTC-0600)

Type
Static

0 Attributes

2. To create default policy: Navigate to: IoT Core → Secure → Policies



□ Name
□ esp32_amazon_freertos_policy
□ blacklistpolicy
□ PSoC64-Policy
□ PSoC64
□ My_IoT_Policy
□ GG_11_SSK_dev

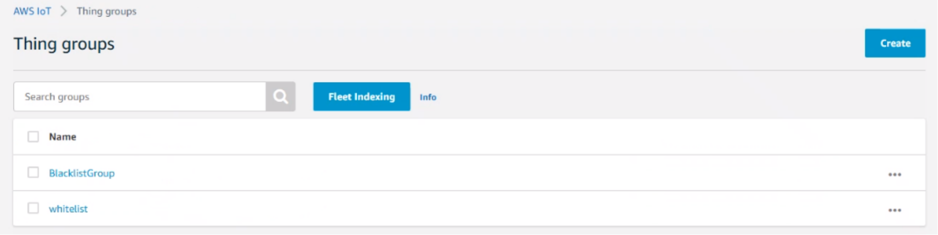
- Click create and enter the name of your policy. Under Add Statements click Advanced mode. JSON statement will be seen and then edit with the information as shown below.

  <u>Note:</u> You will be entering your specific ARN Group provided in the previous step next to “Resource”;

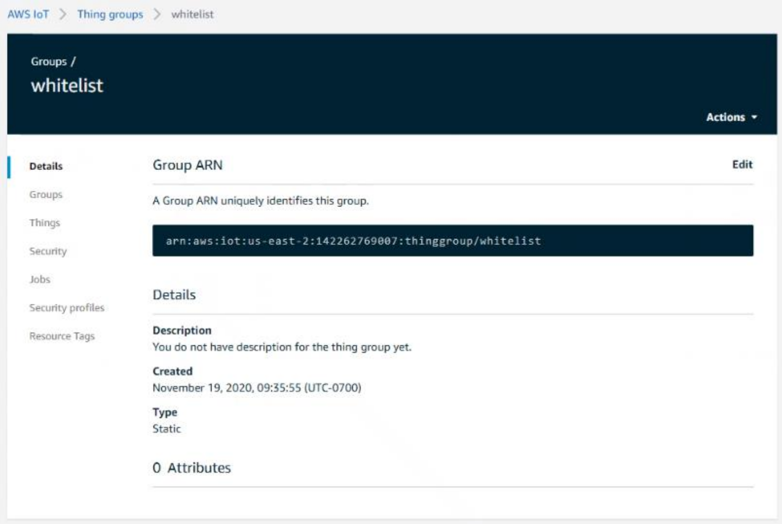
```
{
            “Version”: “2012-10-17”,
            “Statement”: [
            {
              “Effect”: “Deny”,
              “Action”: “iot:*”,
              “Resource”:  “Group ARN noted from the step above:topic/replaceWithATopic”
            }
          ]
}
```

3. Next, you need to attach the “Policy” to the “Thing Group”;

- Navigate to: IoT Core → Manage → Thing Groups and click on the Group you just created;



□ Name
□ BlacklistGroup
□ whitelist



Group ARN

A Group ARN uniquely identifies this group.

```
arn:aws:iot:us-east-2:142262769007:thinggroup/whitelist
```

Details

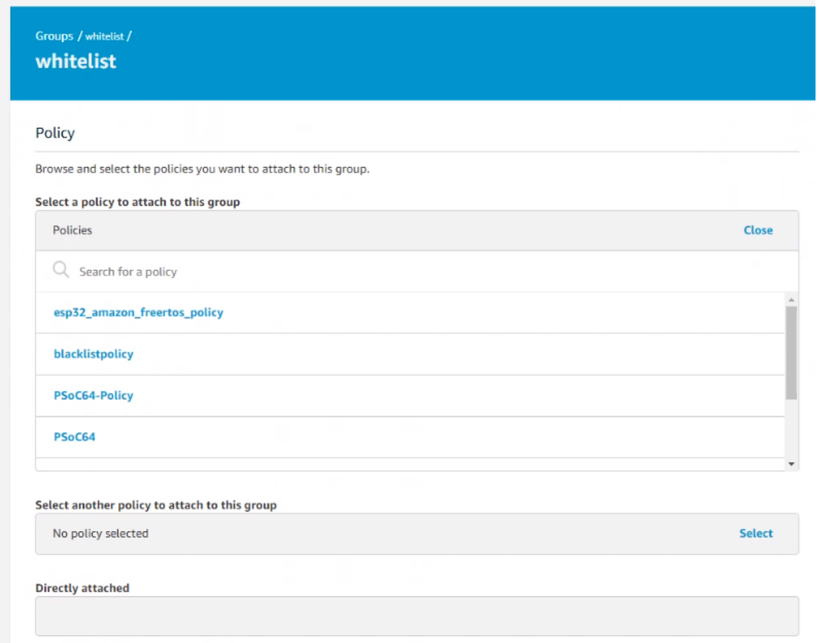
Description
You do not have description for the thing group yet.

Created
November 19, 2020, 09:35:55 (UTC-0700)

Type
Static

0 Attributes

- Click “Security” on the left and then “Edit”, Select the “Policy” you recently created;



Policy

Browse and select the policies you want to attach to this group.

Select a policy to attach to this group

Policies

esp32_amazon_freertos_policy

blacklistpolicy

PSoC64-Policy

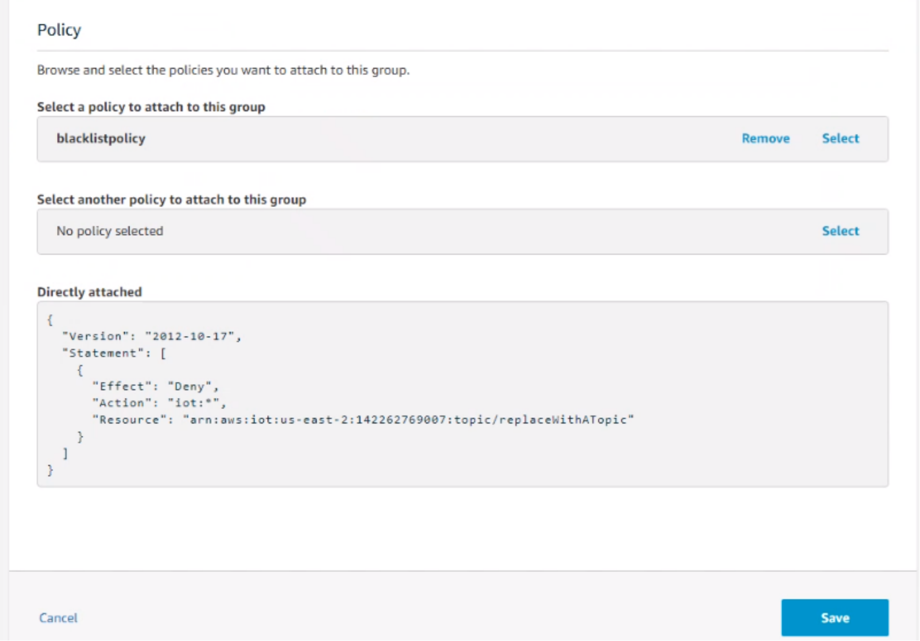
PSoC64

Select another policy to attach to this group

No policy selected

Directly attached

- You should see the policy statements you had edited from the previous steps, then click "Save"



Policy

Browse and select the policies you want to attach to this group.

Select a policy to attach to this group

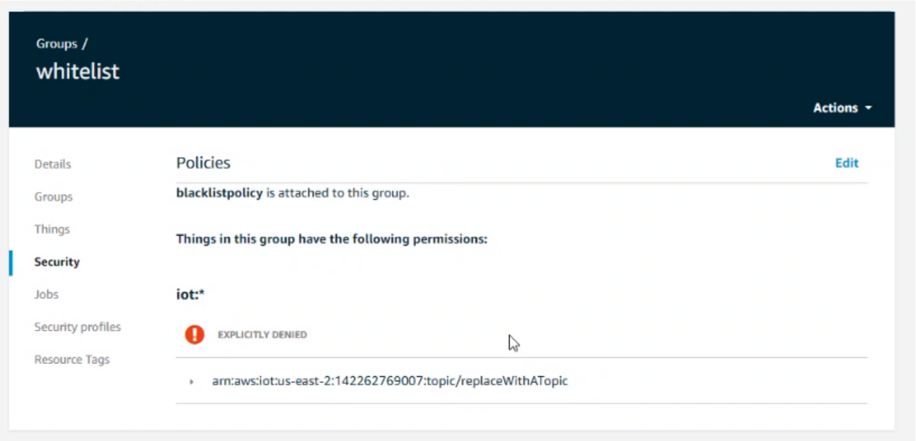
blacklistpolicy

Select another policy to attach to this group

No policy selected

Directly attached

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "iot:*",
      "Resource": "arn:aws:iot:us-east-2:142262769007:topic/replaceWithATopic"
    }
  ]
}
```



Groups /

whitelist

Policies

blacklistpolicy is attached to this group.

Things in this group have the following permissions:

iot:*

EXPLICITLY DENIED

arn:aws:iot:us-east-2:142262769007:topic/replaceWithATopic

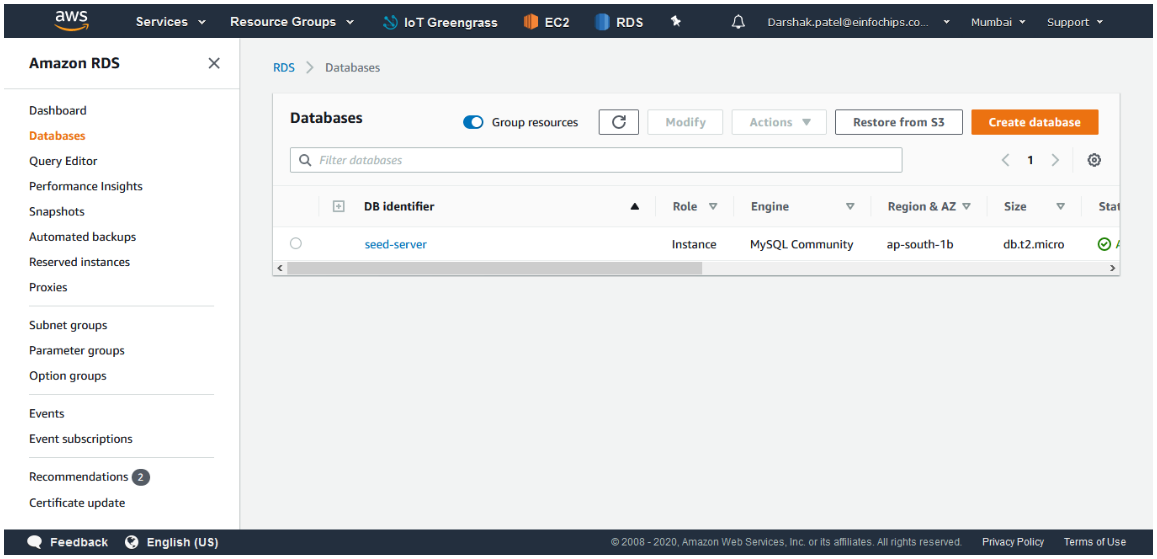
4. In order to perform OTA updates, the user will need to Create an OTA Role and the link to the instructions within AWS is provided below. You will also need to create an OTA Job, which is part of the SSK Cloud Connect Tool and outlined in Section 5 of the SSK Cloud Connect Users Guide;

   To create OTA update role follow below URL:

   URL: https://docs.aws.amazon.com/freertos/latest/userguide/create-service-role.html

4.3 AWS RDS Service – Database Setup and Configuration

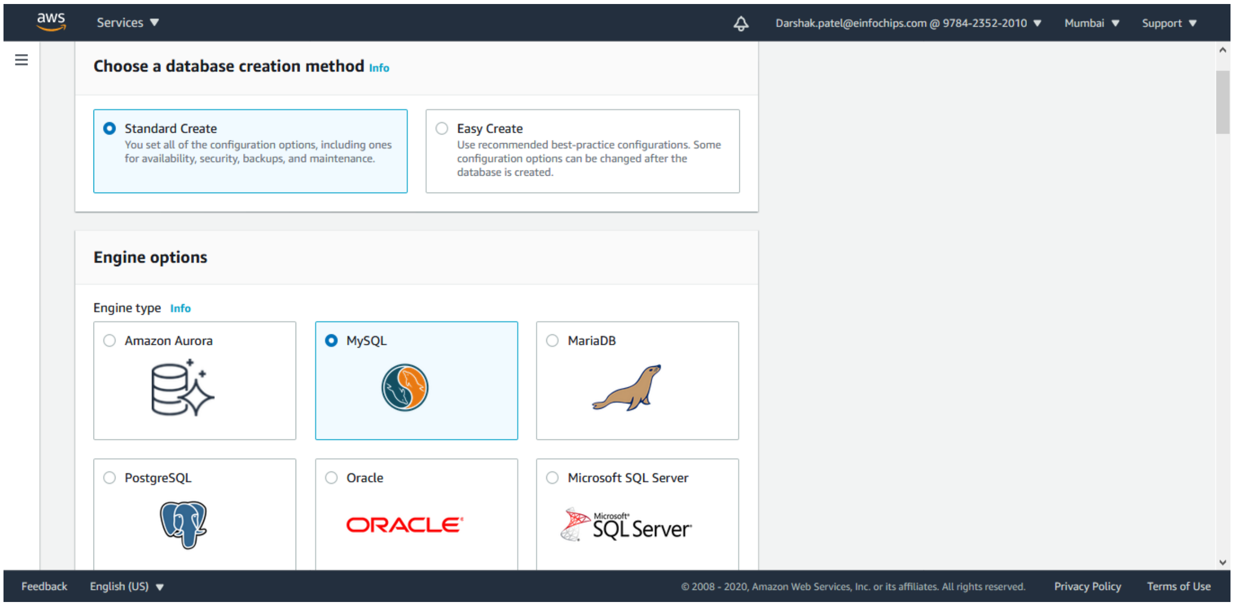
- Go to Services >> Database >> RDS(Select)
- Click Left side navigation “Databases” will show following page.



DB identifier	Role	Engine	Region & AZ	Size	Stat
seed-server	Instance	MySQL Community	ap-south-1b	db.t2.micro	✓ A

4.4 Creating a Database

- Select “Standard Create”


aws
Services
Darshak.patel@einfochips.com @ 9784-2352-2010
Mumbai
Support
Info
Info
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
ORACLE
Microsoft SQL Server

Choose a database creation method

● Standard Create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

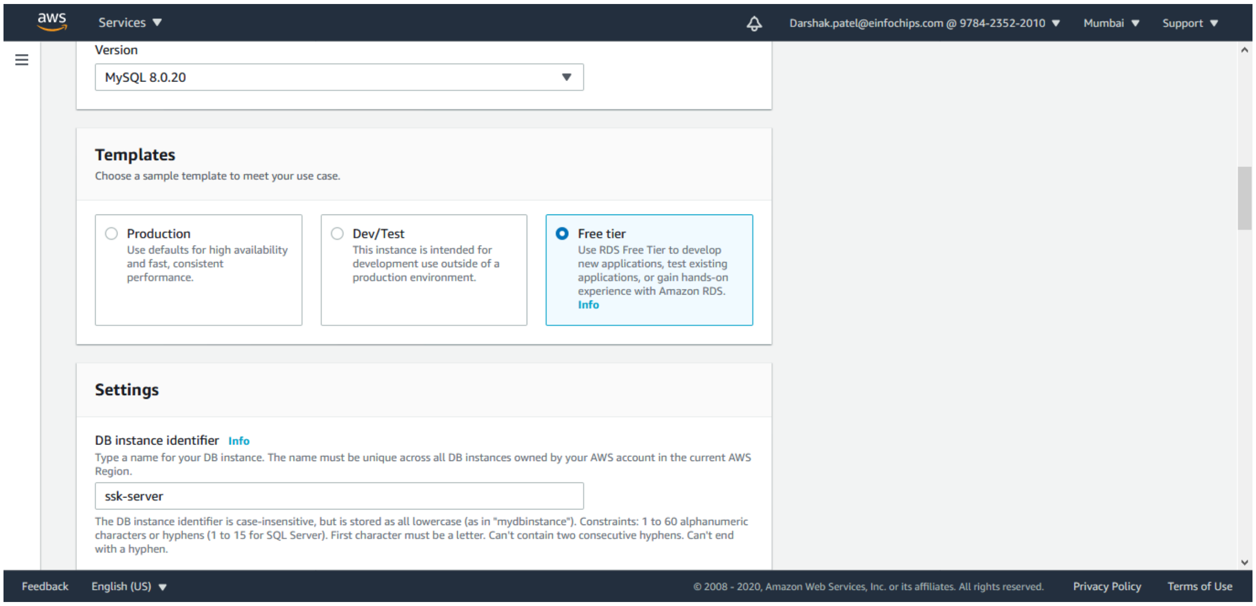
○ Easy Create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type

○ Amazon Aurora
● MySQL
○ MariaDB
○ PostgreSQL
○ Oracle
○ Microsoft SQL Server

- Select “Free Tier”



Version

MySQL 8.0.20

Templates

Choose a sample template to meet your use case.

○ Production
Use defaults for high availability and fast, consistent performance.

○ Dev/Test
This instance is intended for development use outside of a production environment.

● Free tier
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.
Info

Settings

DB instance identifier Info

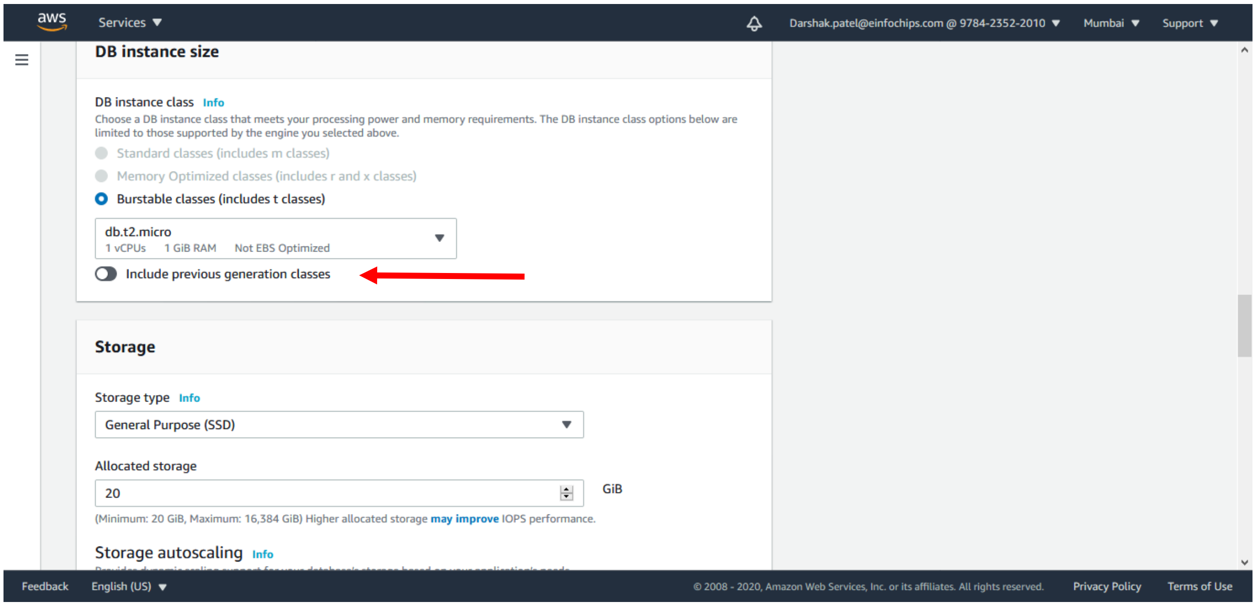
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

ssk-server

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

aws
Services ▼
Darshak.patel@einfochips.com @ 9784-2352-2010 ▼
Mumbai ▼
Support ▼
Feedback
English (US) ▼
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

- Enable “Include previous generation classes”



DB instance size

DB instance class Info

Choose a DB instance class that meets your processing power and memory requirements. The DB instance class options below are limited to those supported by the engine you selected above.

○ Standard classes (includes m classes)

○ Memory Optimized classes (includes r and x classes)

● Burstable classes (includes t classes)

db.t2.micro
1 vCPUs 1 GiB RAM Not EBS Optimized

Include previous generation classes

Storage

Storage type Info

General Purpose (SSD)

Allocated storage

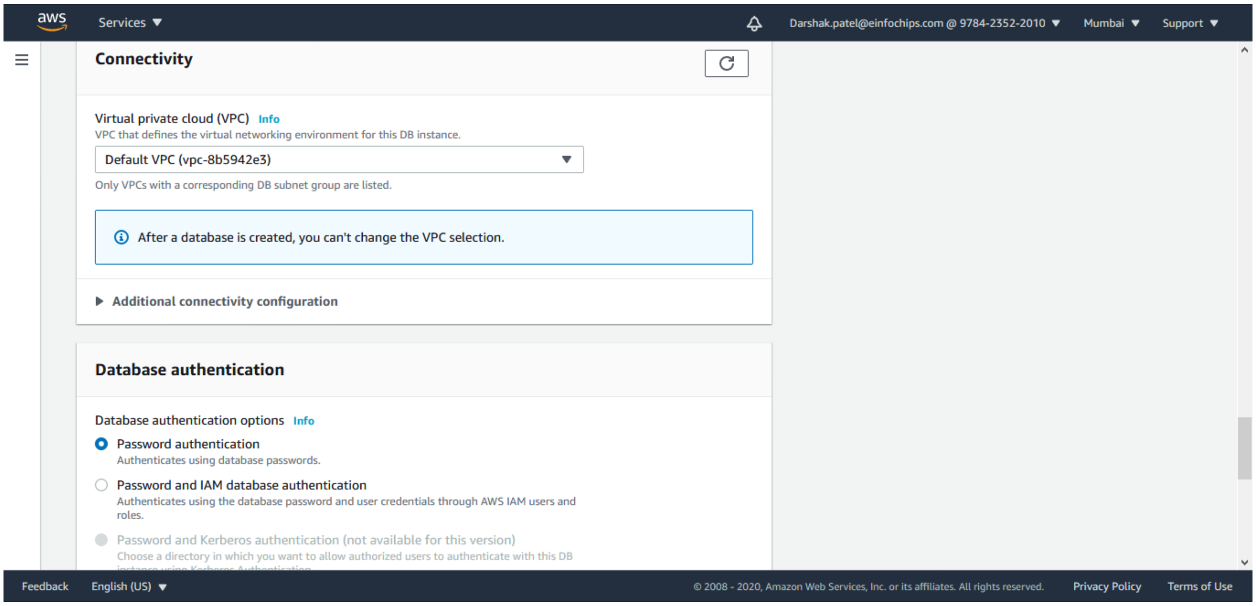
20 GiB

(Minimum: 20 GiB, Maximum: 16,384 GiB) Higher allocated storage **may improve** IOPS performance.

Storage autoscaling Info

aws
Services ▼
Darshak.patel@einfochips.com @ 9784-2352-2010 ▼
Mumbai ▼
Support ▼
Feedback
English (US) ▼
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

- Select “Default VPC” for the Virtual Private Cloud and “Password Authentication”
- Make note of the database password entered


aws
Services ▼
Darshak.patel@einfochips.com @ 9784-2352-2010 ▼
Mumbai ▼
Support ▼
Feedback
English (US) ▼
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Connectivity

Virtual private cloud (VPC) Info
VPC that defines the virtual networking environment for this DB instance.
Default VPC (vpc-8b5942e3) ▼
Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change the VPC selection.

▶ Additional connectivity configuration

Database authentication

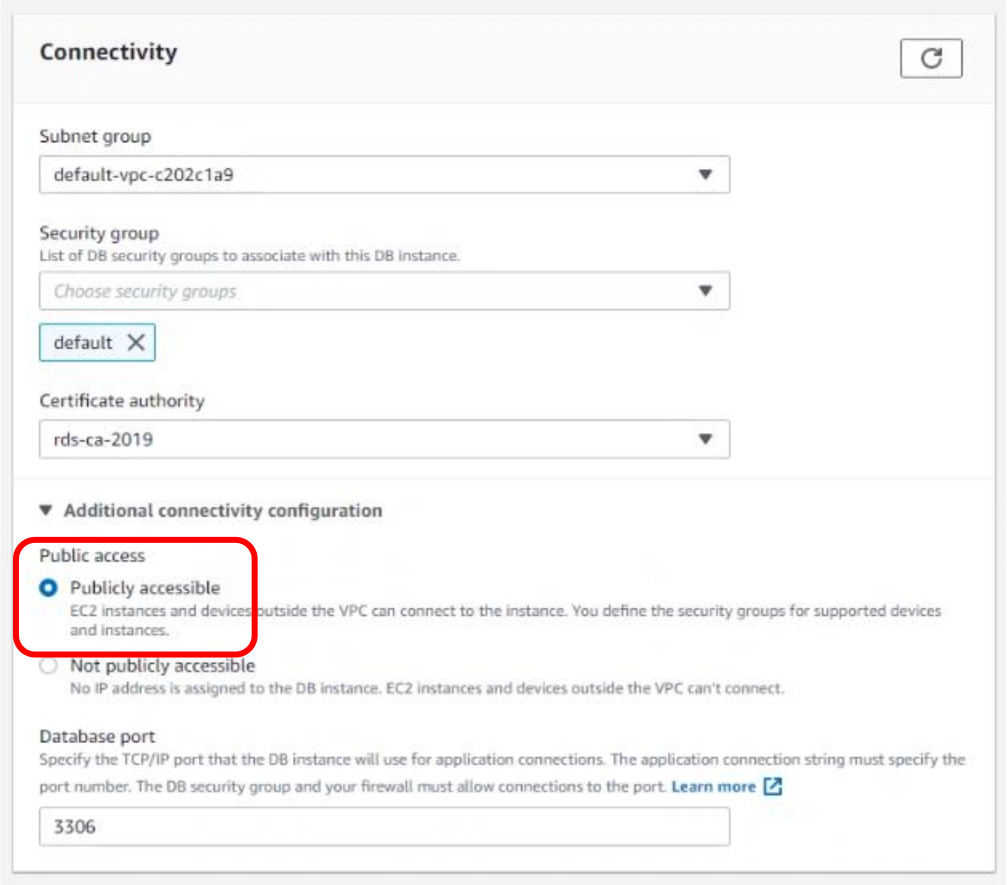
Database authentication options Info

● Password authentication
Authenticates using database passwords.

○ Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.

○ Password and Kerberos authentication (not available for this version)
Choose a directory in which you want to allow authorized users to authenticate with this DB

- Select “Publicly accessible” under Additional Connectivity Configuration



Connectivity

Subnet group
default-vpc-c202c1a9 ▼

Security group
List of DB security groups to associate with this DB instance.
Choose security groups ▼
default ✕

Certificate authority
rds-ca-2019 ▼

▼ Additional connectivity configuration

Public access

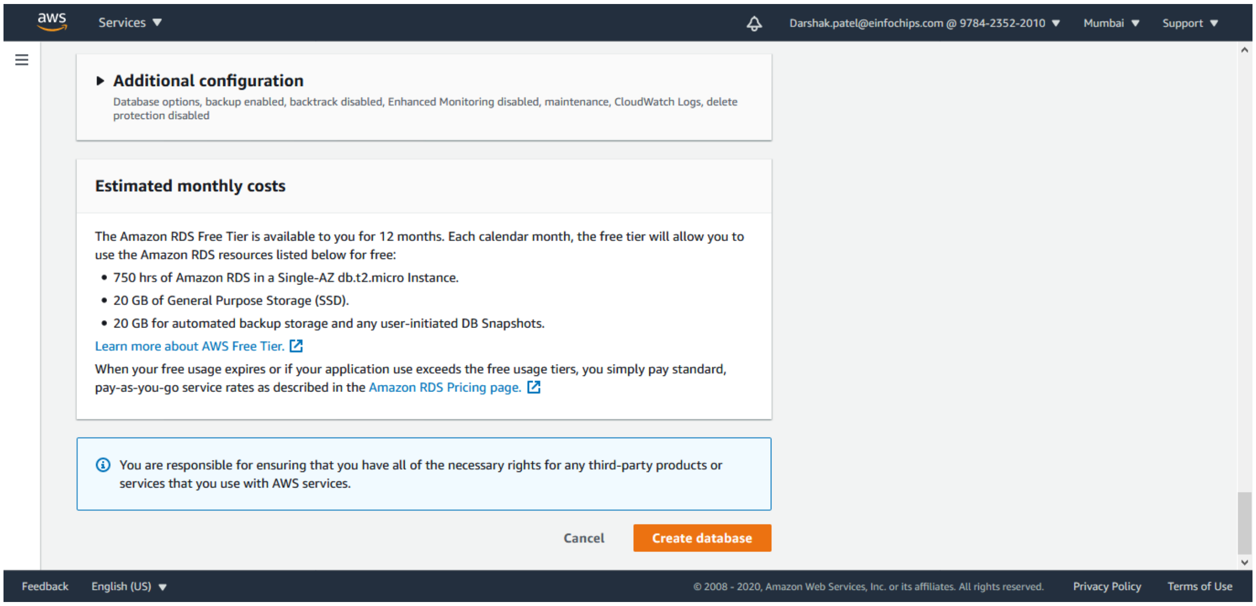
● Publicly accessible
EC2 instances and devices outside the VPC can connect to the instance. You define the security groups for supported devices and instances.

○ Not publicly accessible
No IP address is assigned to the DB instance. EC2 instances and devices outside the VPC can't connect.

Database port
Specify the TCP/IP port that the DB instance will use for application connections. The application connection string must specify the port number. The DB security group and your firewall must allow connections to the port. Learn more

3306

- Click “Create database”



Additional configuration

Database options, backup enabled, backtrack disabled, Enhanced Monitoring disabled, maintenance, CloudWatch Logs, delete protection disabled

Estimated monthly costs

The Amazon RDS Free Tier is available to you for 12 months. Each calendar month, the free tier will allow you to use the Amazon RDS resources listed below for free:

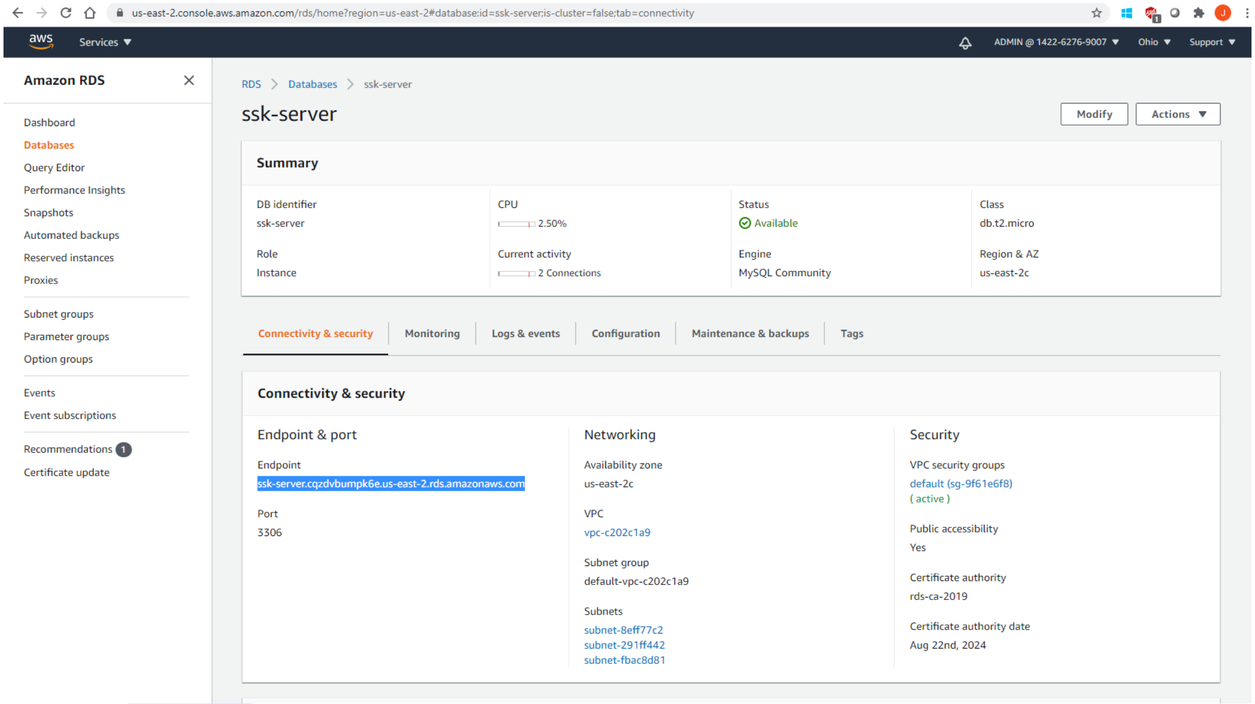
- 750 hrs of Amazon RDS in a Single-AZ db.t2.micro Instance.
- 20 GB of General Purpose Storage (SSD).
- 20 GB for automated backup storage and any user-initiated DB Snapshots.

Learn more about AWS Free Tier.

When your free usage expires or if your application use exceeds the free usage tiers, you simply pay standard, pay-as-you-go service rates as described in the Amazon RDS Pricing page.

You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



ssk-server

Summary

DB identifier	CPU	Status	Class
ssk-server	2.50%	Available	db.t2.micro
Role	Current activity	Engine	Region & AZ
Instance	2 Connections	MySQL Community	us-east-2c

Connectivity & security

Endpoint & port

Endpoint
ssk-server.cqzdvbumpk6e.us-east-2.rds.amazonaws.com

Port
3306

Networking

Availability zone
us-east-2c

VPC
vpc-c202c1a9

Subnet group
default-vpc-c202c1a9

Subnets
subnet-8eff77c2
subnet-291ff442
subnet-fbac8d81

Security

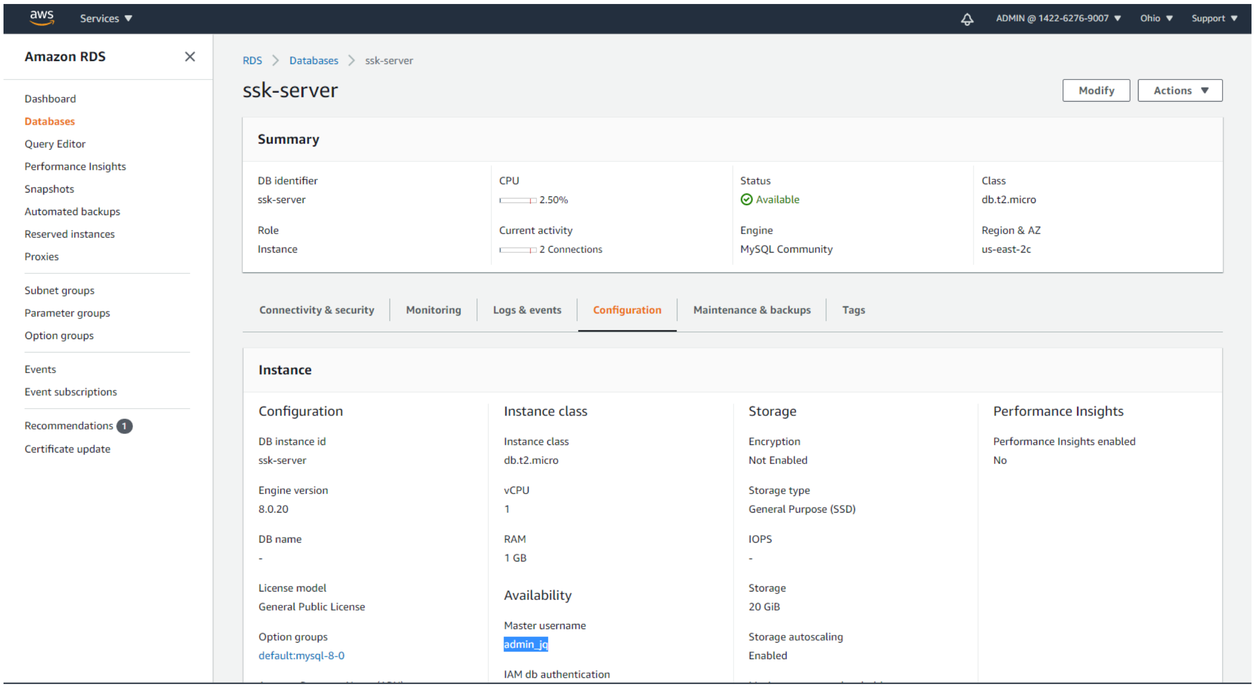
VPC security groups
default (sg-9f61e6f8)
(active)

Public accessibility
Yes

Certificate authority
rds-ca-2019

Certificate authority date
Aug 22nd, 2024

- Make note of the RDS URL that is created and highlighted above.



ssk-server

Summary

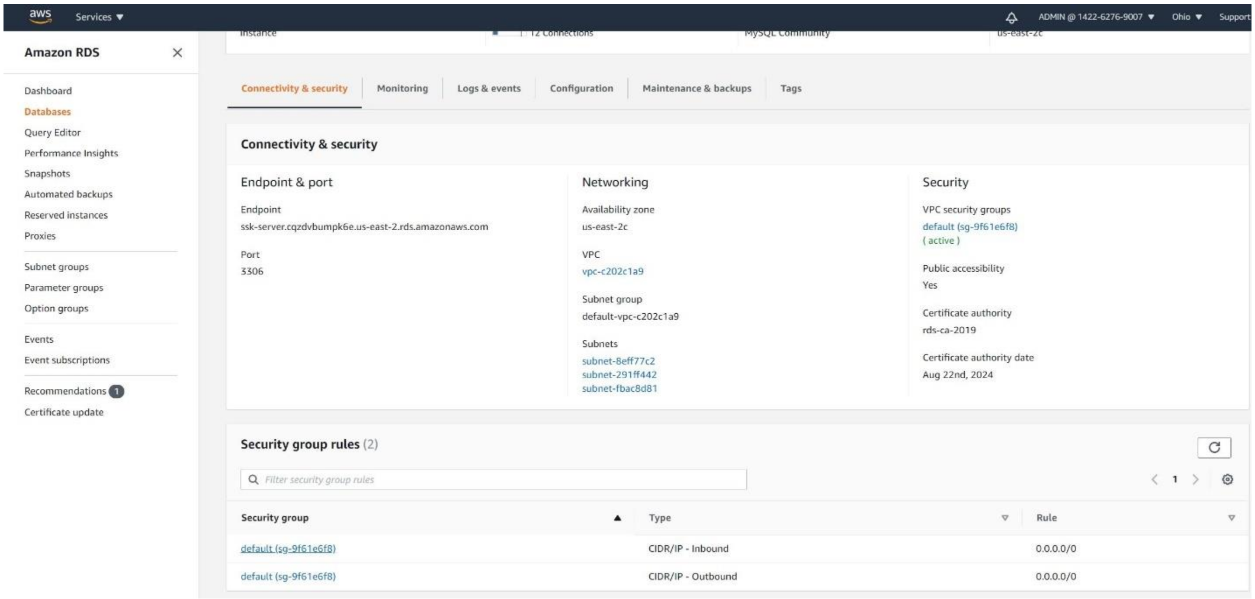
DB identifier ssk-server	CPU 2.50%	Status Available	Class db.t2.micro
Role Instance	Current activity 2 Connections	Engine MySQL Community	Region & AZ us-east-2c

Instance

Configuration	Instance class	Storage	Performance Insights
DB instance id ssk-server	Instance class db.t2.micro	Encryption Not Enabled	Performance Insights enabled No
Engine version 8.0.20	vCPU 1	Storage type General Purpose (SSD)	
DB name -	RAM 1 GB	IOPS -	
License model General Public License	**Availability**	Storage 20 GiB	
Option groups default:mysql-8-0	Master username admin_jq	Storage autoscaling Enabled	
	IAM db authentication		

- Make note of the User Name highlighted above, under the configurations tab.

- Once MySQL Database is created, ensure the below Security group rules are set by clicking on the default security group under “Security group rules” in Amazon RDS



Connectivity & security

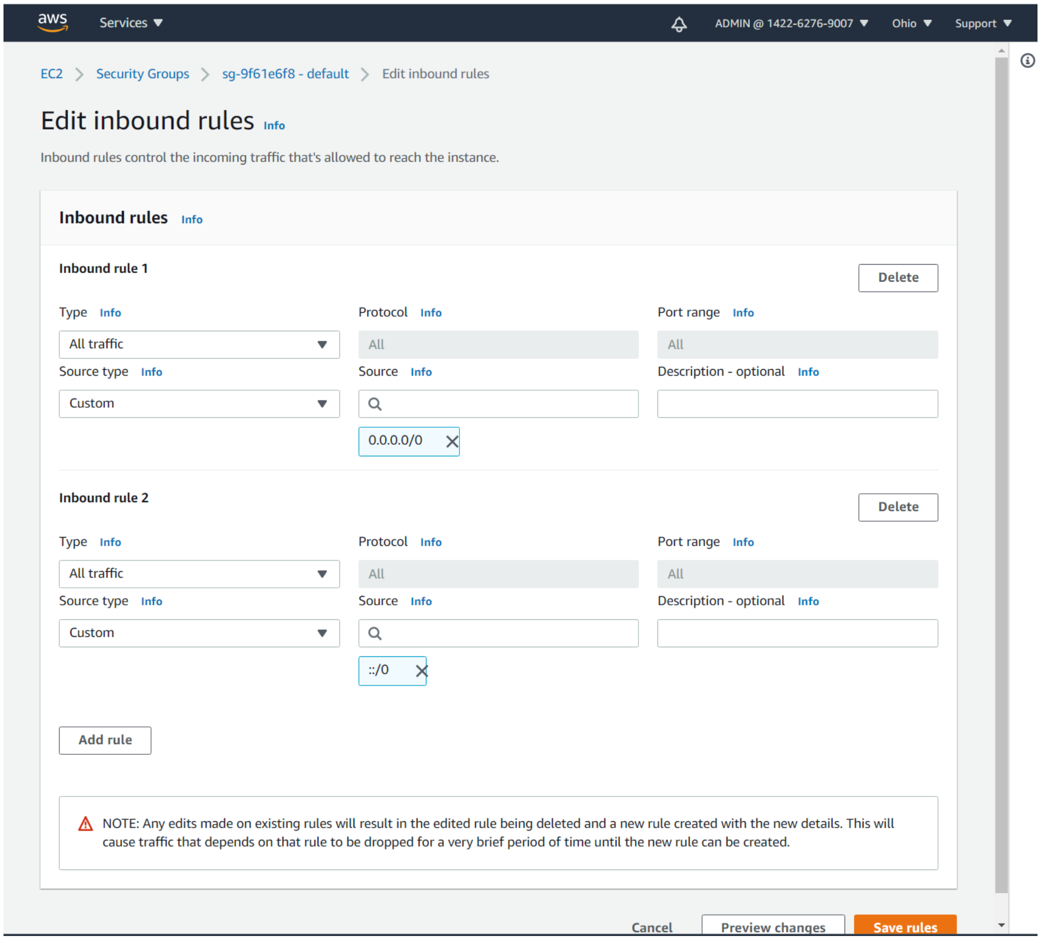
Endpoint & port	Networking	Security
Endpoint ssk-server.cqzdvbumpk6e.us-east-2.rds.amazonaws.com	Availability zone us-east-2c	VPC security groups default (sg-9f61e6f8) (active)
Port 3306	VPC vpc-c202c1a9	Public accessibility Yes
	Subnet group default-vpc-c202c1a9	Certificate authority rds-ca-2019
	Subnets subnet-8eff77c2 subnet-291ff442 subnet-fbac8d81	Certificate authority date Aug 22nd, 2024

Security group rules (2)

Filter security group rules

Security group	Type	Rule
default (sg-9f61e6f8)	CIDR/IP - Inbound	0.0.0.0/0
default (sg-9f61e6f8)	CIDR/IP - Outbound	0.0.0.0/0

Inbound Rules:



EC2 > Security Groups > sg-9f61e6f8 - default > Edit inbound rules

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Inbound rule 1

Type Info: All traffic

Protocol Info: All

Port range Info: All

Source type Info: Custom

Source Info: 0.0.0.0/0

Description - optional Info:

Inbound rule 2

Type Info: All traffic

Protocol Info: All

Port range Info: All

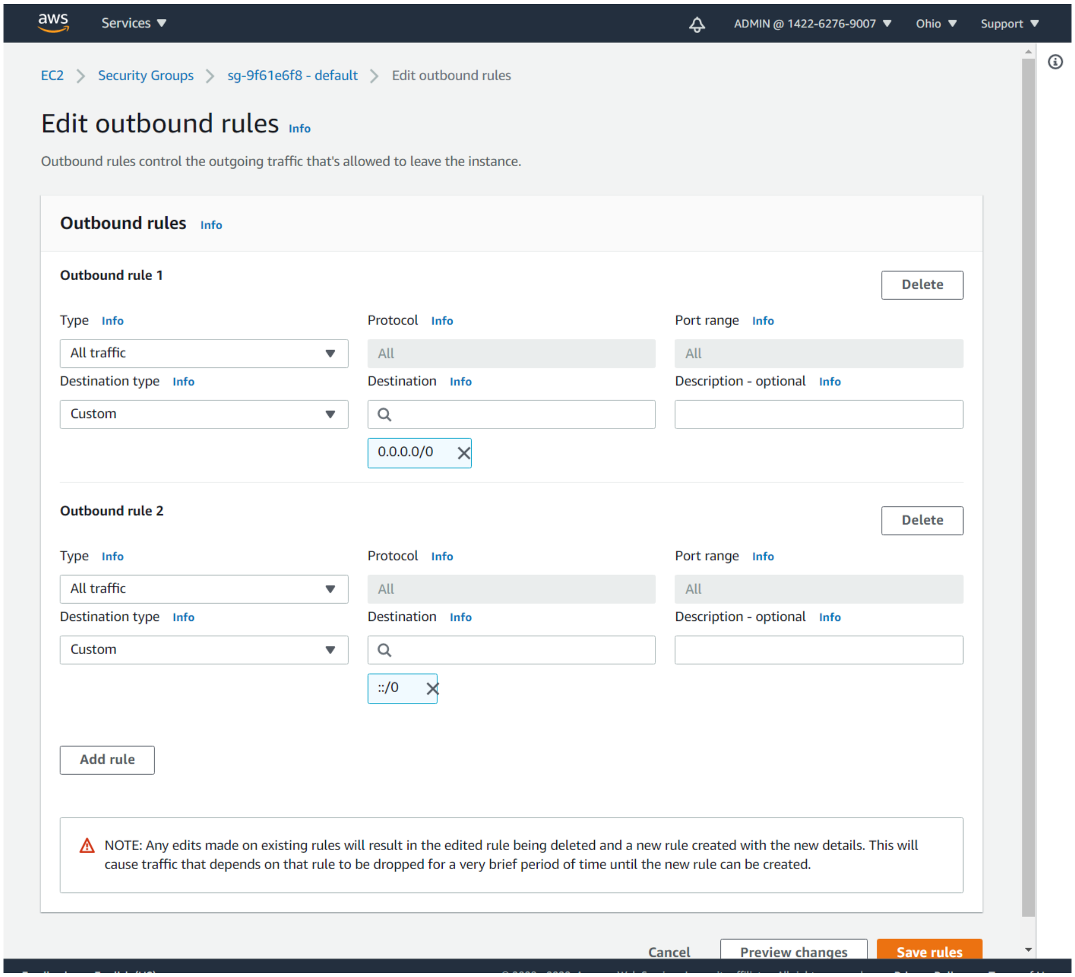
Source type Info: Custom

Source Info: ::/0

Description - optional Info:

⚠ NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Outbound Rules:



Edit outbound rules Info

Outbound rules control the outgoing traffic that's allowed to leave the instance.

Outbound rules Info

Outbound rule 1

Type: All traffic	Protocol: All	Port range: All
Destination type: Custom	Destination: 0.0.0.0/0	Description - optional:

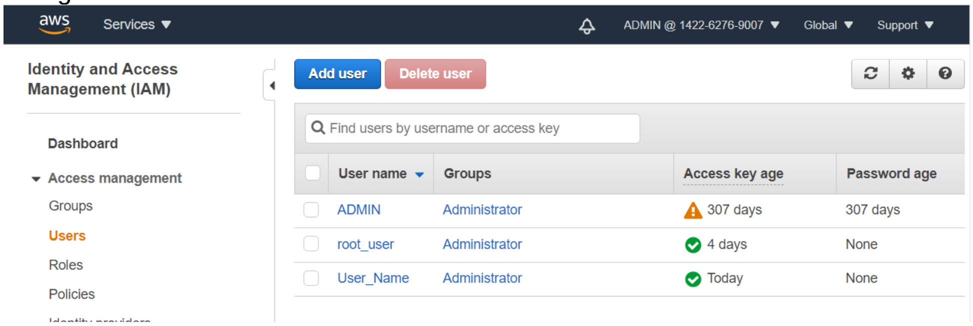
Outbound rule 2

Type: All traffic	Protocol: All	Port range: All
Destination type: Custom	Destination: ::/0	Description - optional:

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

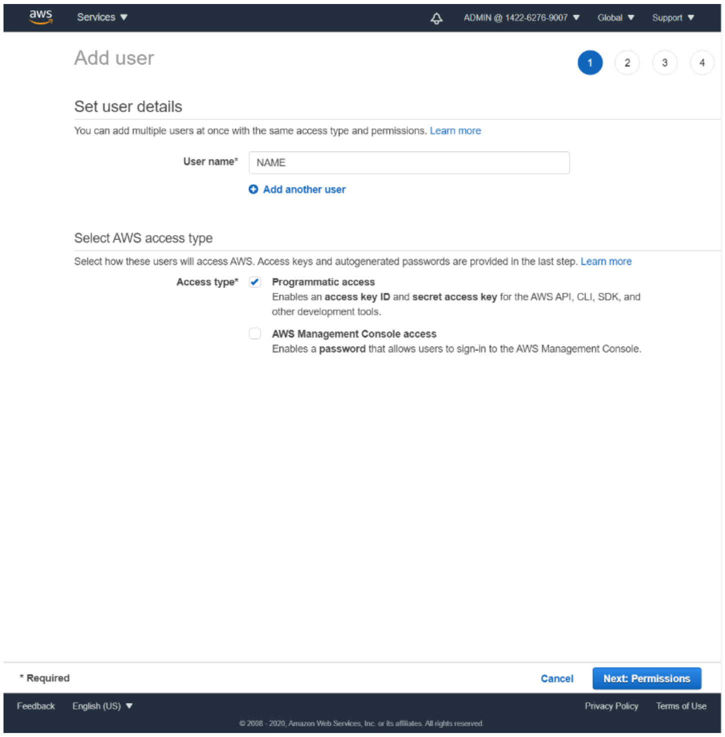
4.5 Creating an IAM User

- In AWS go to Services >> IAM >> Users and select 'Add User'



User name	Groups	Access key age	Password age
ADMIN	Administrator	⚠ 307 days	307 days
root_user	Administrator	✓ 4 days	None
User_Name	Administrator	✓ Today	None

- Pick a username and give it Programmatic access. Note this will be the username you will use to log into SSK Cloud Connect



Add user

Set user details

You can add multiple users at once with the same access type and permissions. Learn more

User name* NAME

Add another user

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. Learn more

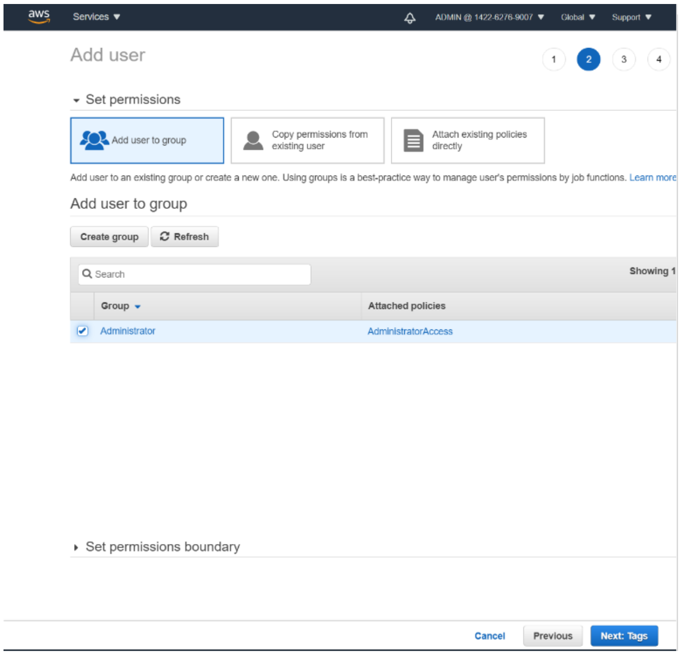
Access type* ☑ Programmatic access
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

□ AWS Management Console access
Enables a **password** that allows users to sign-in to the AWS Management Console.

* Required

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

- Choose ‘Next: Permissions’ and add your IAM user to the Administrator group with the Administrator Access policy. If you don’t have an Administrator group then choose “Attach existing policies directly, search for the ‘Administrator Access’ policy and attach it



Add user

Set permissions

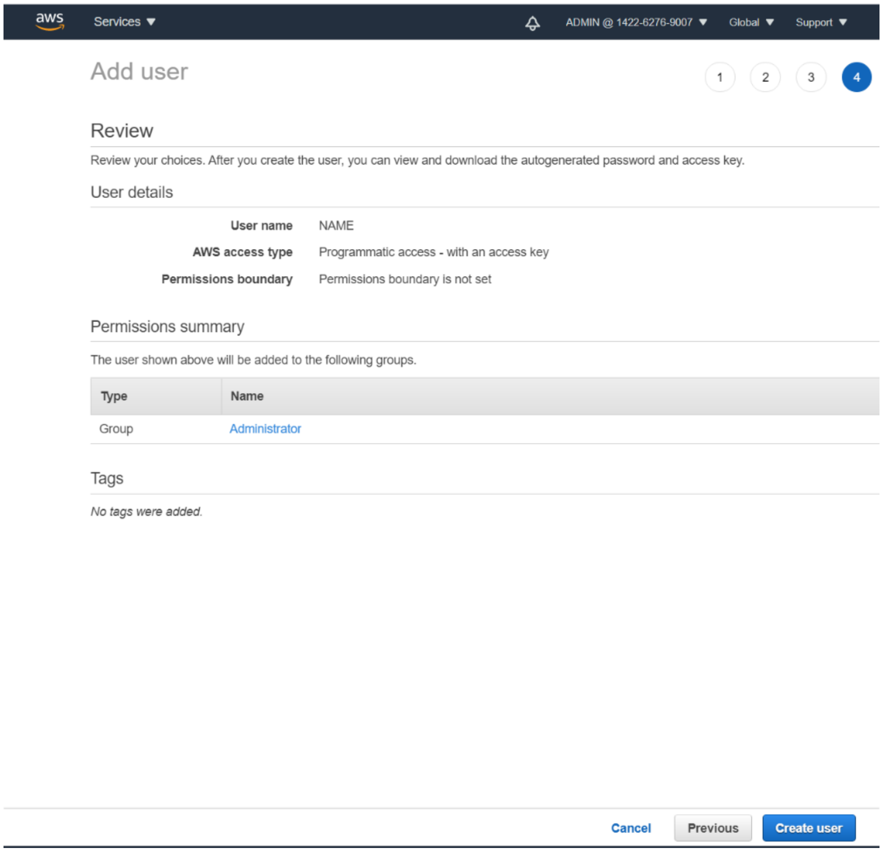
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more

Add user to group

	Group	Attached policies
☑	Administrator	AdministratorAccess

Set permissions boundary

- Click Next: Tags >> Next: Review >> Create user



Add user

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name	NAME
AWS access type	Programmatic access - with an access key
Permissions boundary	Permissions boundary is not set

Permissions summary

The user shown above will be added to the following groups.

Type	Name
Group	Administrator

Tags

No tags were added.

- Download the ‘new_user_credentials.csv’ and save it in a safe location

4.6 MySQL Setup and Configuration

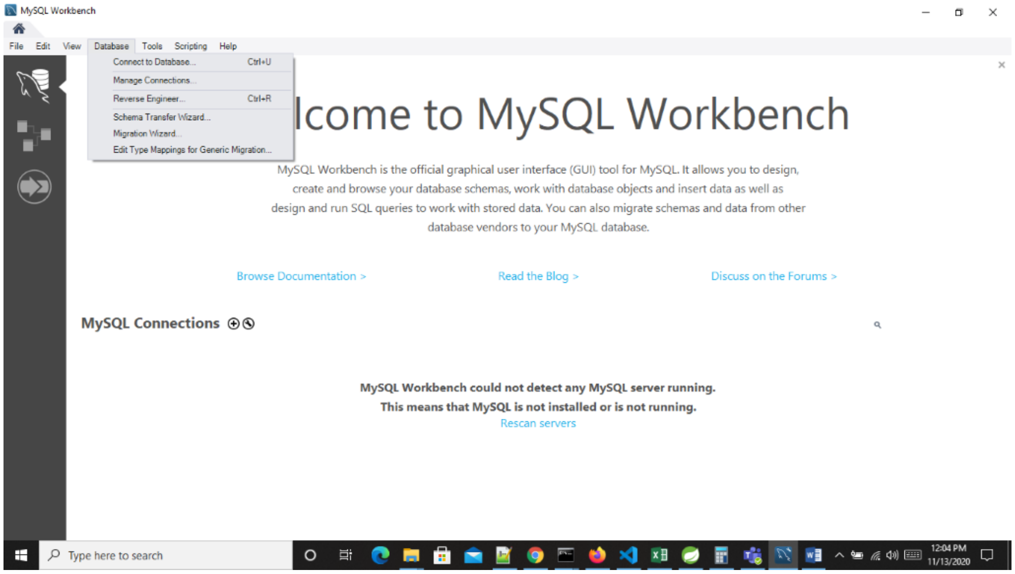
1. Install MySQL Workbench (link provided below), then Open MySQL workbench.

   https://dev.mysql.com/downloads/workbench/

2. Install Postman

   https://www.postman.com/

3. On Tab Database & select Manage connections.(Database -> Manage Connections)



lcome to MySQL Workbench

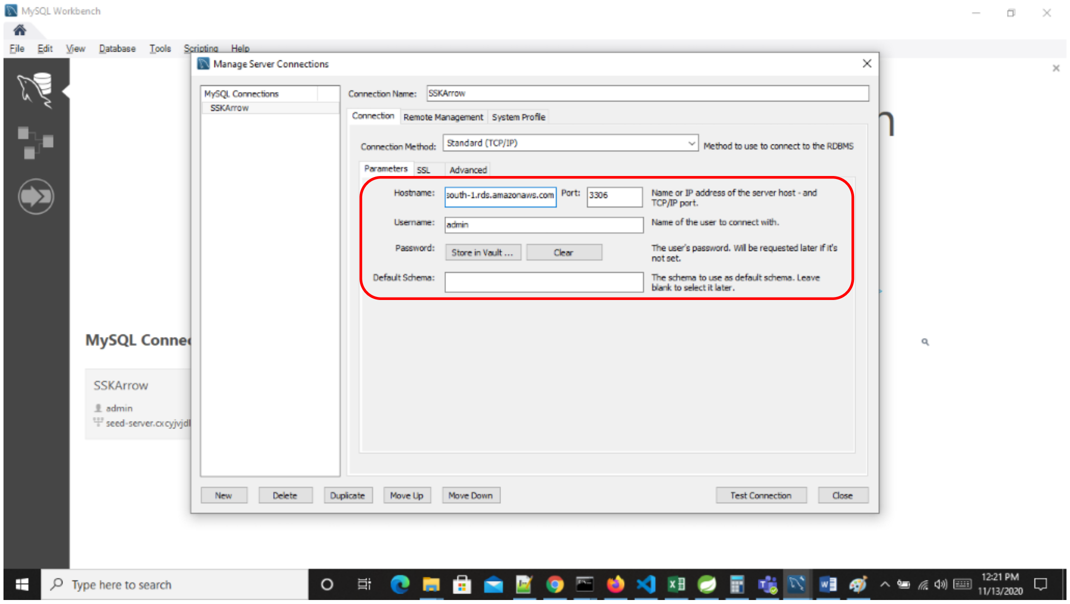
MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

MySQL Connections

MySQL Workbench could not detect any MySQL server running.
This means that MySQL is not installed or is not running.

4. It will open pop up model. Press New Button then fill up details as per below:

   Connection Name: <Name of connection>
   Host Name: <AWS RDS HOST URL>
   Port: 3306 (Default value)
   User Name: < AWS RDS User name>
   Password: <AWS RDS user password> (Store in Vault if needed)



Connection Name: SSKArrow

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

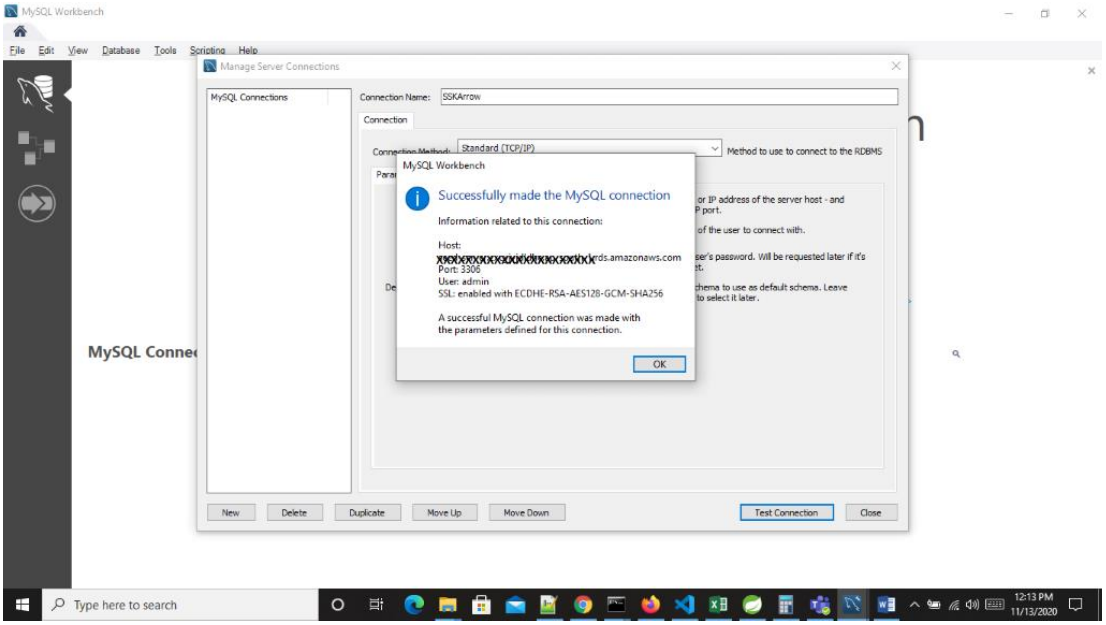
Hostname: south-1.rds.amazonaws.com Port: 3306 — Name or IP address of the server host - and TCP/IP port.

Username: admin — Name of the user to connect with.

Password: — The user's password. Will be requested later if it's not set.

Default Schema: — The schema to use as default schema. Leave blank to select it later.

5. Press on Test Connection. It will pop up successful connection message. Then click on close button



Connection Name: SSKArrow

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

MySQL Workbench

Successfully made the MySQL connection

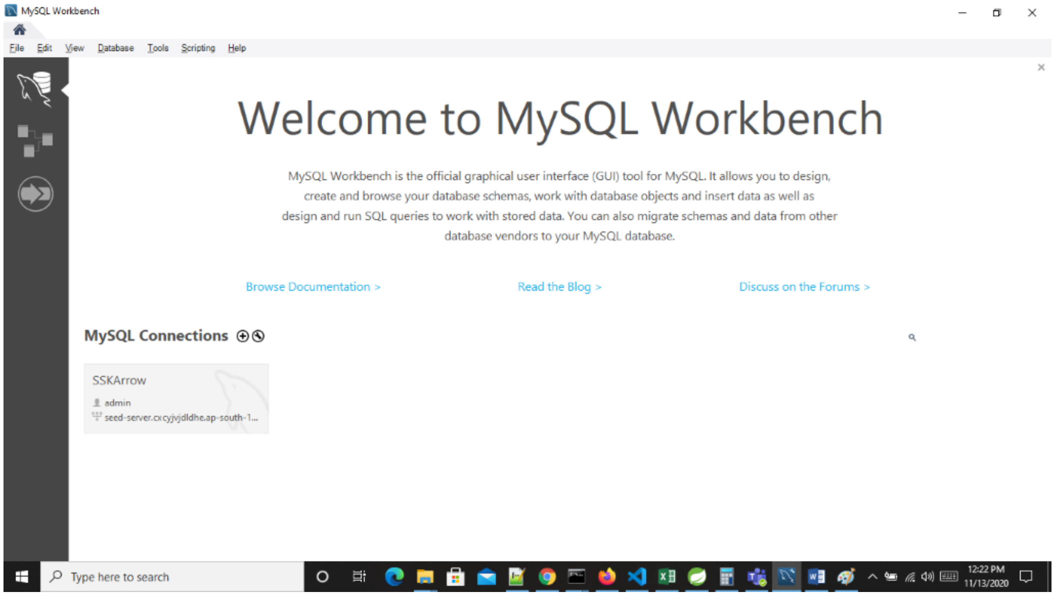
Information related to this connection:

Host:
[redacted]rds.amazonaws.com
Port: 3306
User: admin
SSL: enabled with ECDHE-RSA-AES128-GCM-SHA256

A successful MySQL connection was made with
the parameters defined for this connection.

OK

6. Further it will show following details. Click on created connection button.

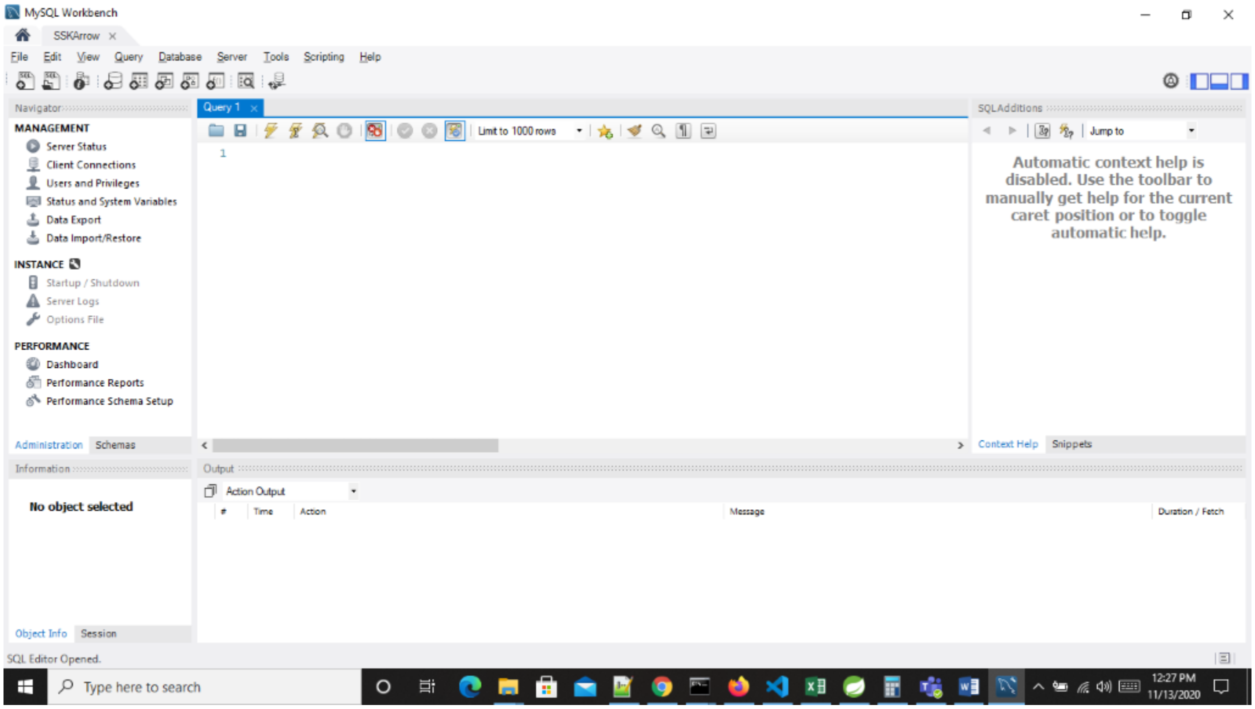


Welcome to MySQL Workbench

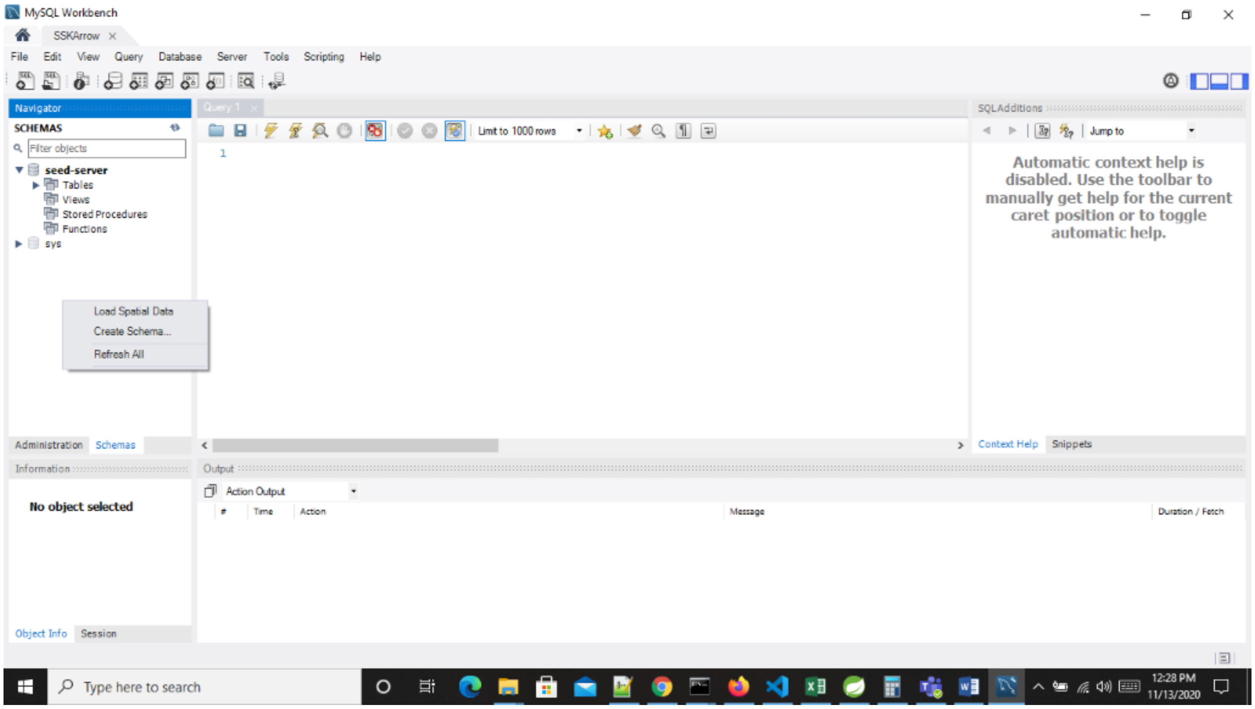
MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

SSKArrow
admin
seed-server.cxcyjvjdldhe.ap-south-1...

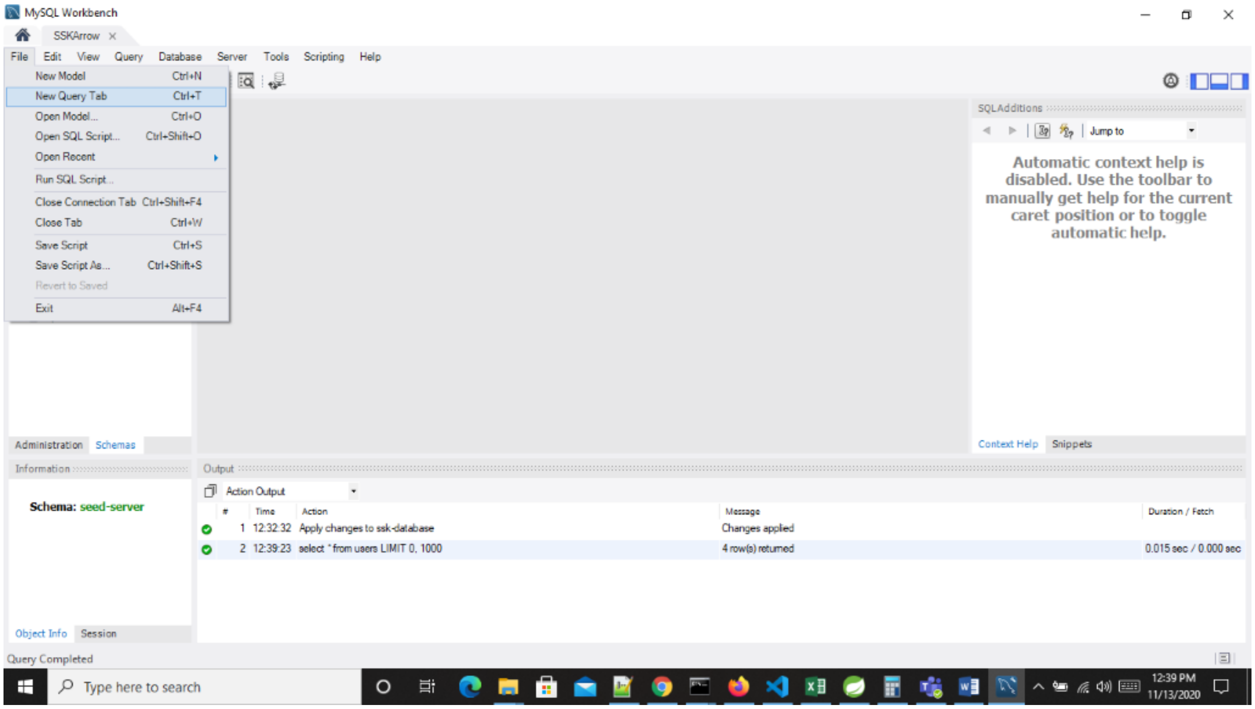
7. It will open Administration tab.



8. Select Schemas Tab & right click on mouse.



9. Click on File Tab then select “New Query Tab”



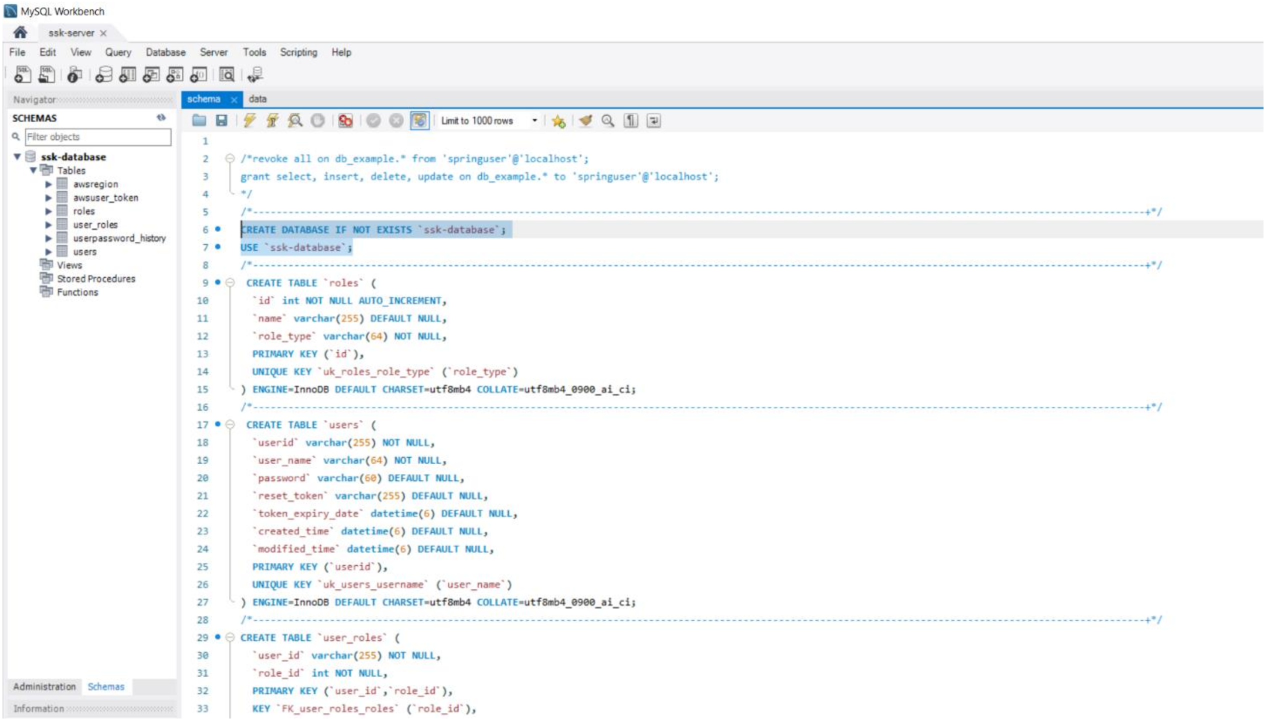
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Schema: seed-server

```
#  Time      Action                                  Message            Duration / Fetch
1  12:32:32  Apply changes to ssk-database           Changes applied
2  12:39:23  select * from users LIMIT 0, 1000       4 row(s) returned  0.015 sec / 0.000 sec
```

10. Edit and Execute schema.sql

- Go to File ->Open SQL Script…, navigate to the location of ‘schema.sql’ on your PC, and select it
- Modify lines 6 and 7 and choose a unique name for the schema database, for example `ssk-database`
- Make note of this name
- Highlight lines 6 and 7 as shown below and click the yellow bolt one time to execute the selected lines in schema.sql



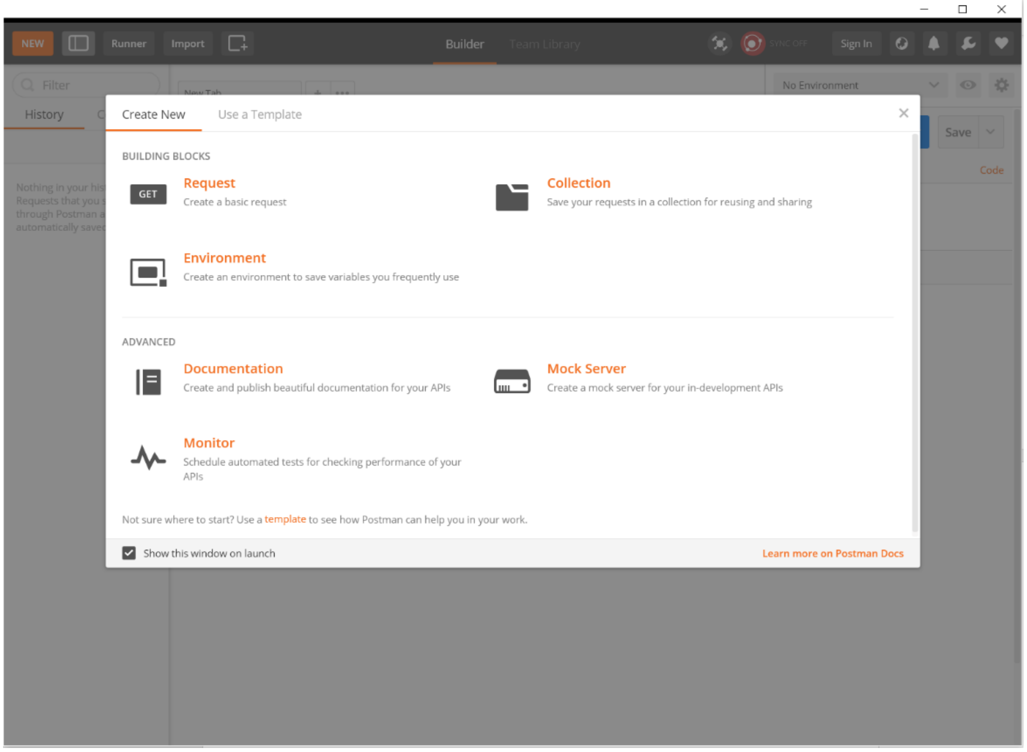
```sql

/*revoke all on db_example.* from 'springuser'@'localhost';
grant select, insert, delete, update on db_example.* to 'springuser'@'localhost';
*/
/*----------------------------------------------------------------------------------------------------------------------------------------------------------+*/
CREATE DATABASE IF NOT EXISTS `ssk-database`;
USE `ssk-database`;
/*----------------------------------------------------------------------------------------------------------------------------------------------------------+*/
 CREATE TABLE `roles` (
  `id` int NOT NULL AUTO_INCREMENT,
  `name` varchar(255) DEFAULT NULL,
  `role_type` varchar(64) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `uk_roles_role_type` (`role_type`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*----------------------------------------------------------------------------------------------------------------------------------------------------------+*/
 CREATE TABLE `users` (
  `userid` varchar(255) NOT NULL,
  `user_name` varchar(64) NOT NULL,
  `password` varchar(60) DEFAULT NULL,
  `reset_token` varchar(255) DEFAULT NULL,
  `token_expiry_date` datetime(6) DEFAULT NULL,
  `created_time` datetime(6) DEFAULT NULL,
  `modified_time` datetime(6) DEFAULT NULL,
  PRIMARY KEY (`userid`),
  UNIQUE KEY `uk_users_username` (`user_name`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*----------------------------------------------------------------------------------------------------------------------------------------------------------+*/
CREATE TABLE `user_roles` (
  `user_id` varchar(255) NOT NULL,
  `role_id` int NOT NULL,
  PRIMARY KEY (`user_id`,`role_id`),
  KEY `FK_user_roles_roles` (`role_id`),
```

11. Edit and Execute data.sql

    - Go to File->Open SQL Script…, navigate to the location of data.sql on your PC, and select it
    - Locate the User name, Access key ID, and Secret access key of your IAM user. Note these credentials can be found in the 'new_user_credentials.csv' file that was downloaded after creating an IAM user
    - Modify line 18 of data.sql by entering your IAM credentials. It should have a similar structure shown below

      INSERT IGNORE INTO awsuser_token(user_name,access_key_id,secret_access_key,is_root_account,create_date) values('<your_iam_username>','<your_access_key>','<your_secret_key>',1,now());

    - Launch the Postman app
    - Under the 'Create New' tab select 'Request'



Create New | Use a Template

BUILDING BLOCKS

Request – Create a basic request

Collection – Save your requests in a collection for reusing and sharing

Environment – Create an environment to save variables you frequently use

ADVANCED

Documentation – Create and publish beautiful documentation for your APIs

Mock Server – Create a mock server for your in-development APIs

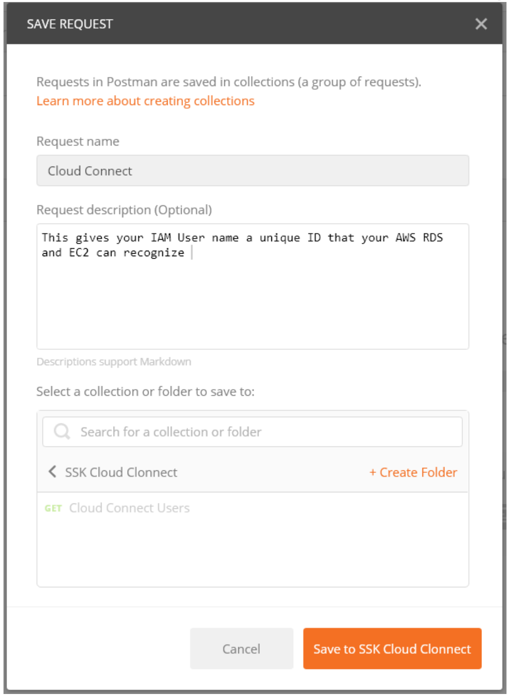
Monitor – Schedule automated tests for checking performance of your APIs

Not sure where to start? Use a template to see how Postman can help you in your work.

☑ Show this window on launch

Learn more on Postman Docs

- Enter a request name, create a new collection to save the request to, and save it



SAVE REQUEST

Requests in Postman are saved in collections (a group of requests).
Learn more about creating collections

Request name: Cloud Connect

Request description (Optional):

```
This gives your IAM User name a unique ID that your AWS RDS
and EC2 can recognize
```

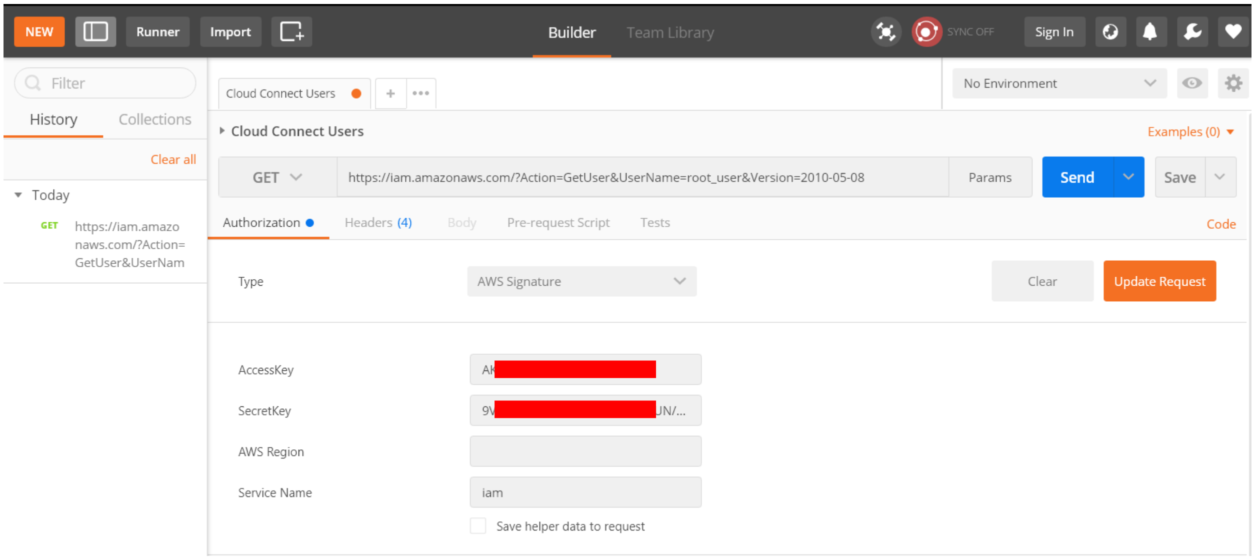
Descriptions support Markdown

Select a collection or folder to save to:

- Under the ‘Authorization’ tab select ‘AWS Signature’ next to Type and enter your IAM credentials into the ‘AccessKey’ and ‘SecretKey’ text boxes
- Next to ‘Get’ <Enter request URL> copy and paste the below URL and change the highlighted text with the username of your IAM

https://iam.amazonaws.com/?Action=GetUser&UserName=IAM-username&Version=2010-05-08

- Leave 'AWS Region' empty and enter 'iam' (all lowercase) next to Service Name as shown below



GET https://iam.amazonaws.com/?Action=GetUser&UserName=root_user&Version=2010-05-08

Type: AWS Signature

AccessKey: AK[redacted]

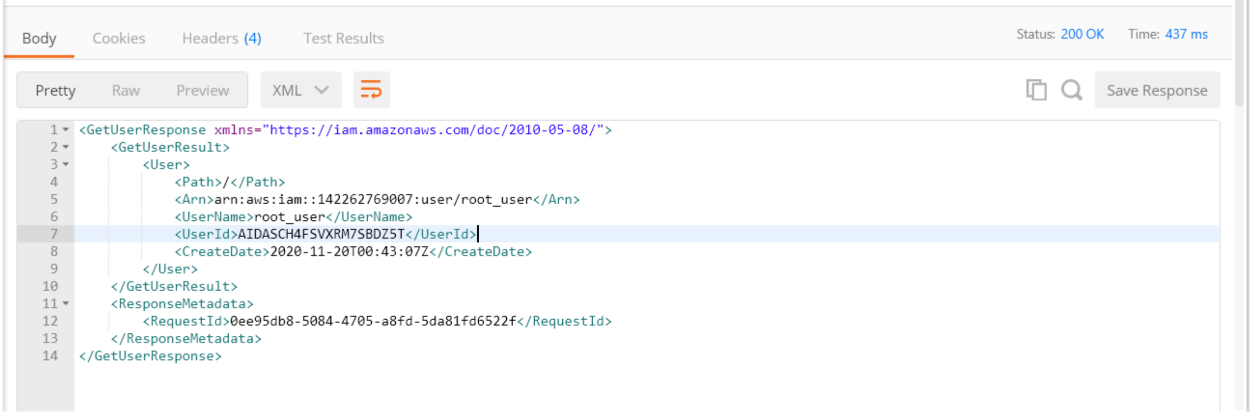
SecretKey: 9V[redacted]UN/...

AWS Region:

Service Name: iam

□ Save helper data to request

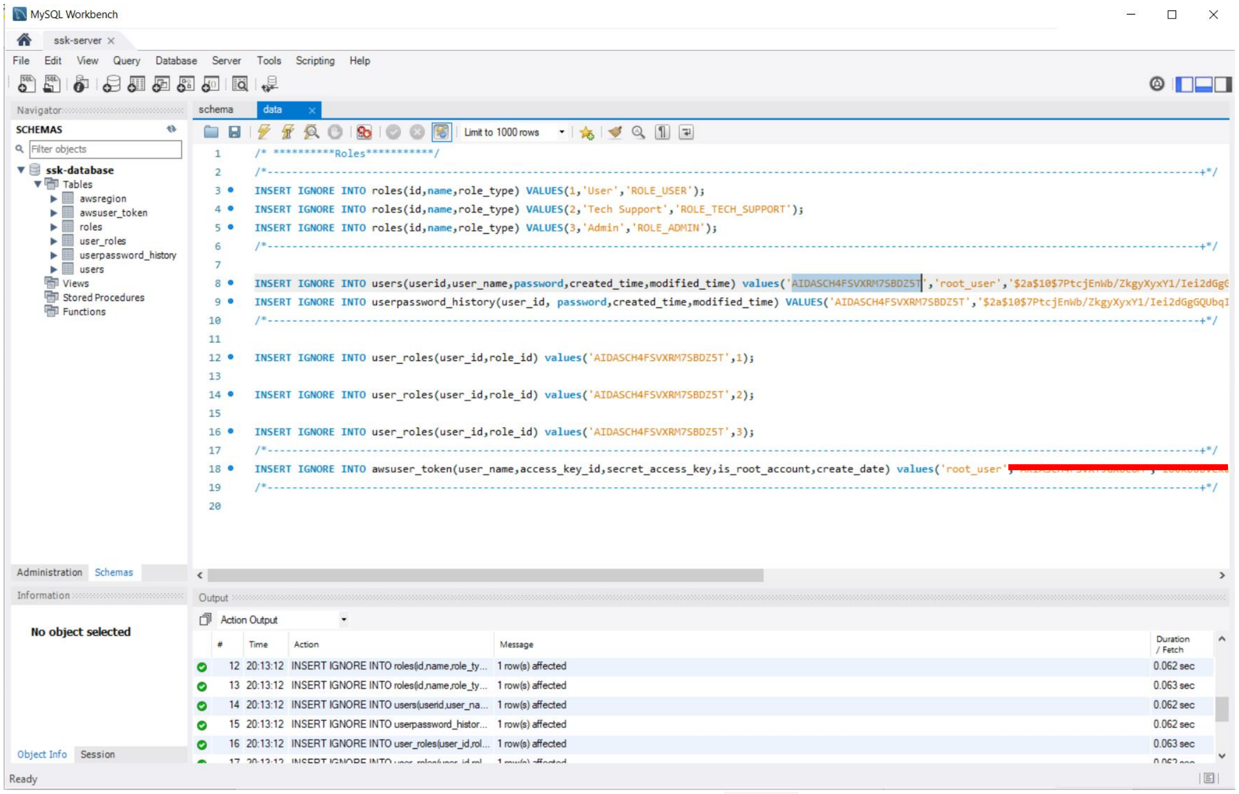
- Now choose 'Send' and copy the ID that was generated below next to '<UserID>'



Status: 200 OK Time: 437 ms

```xml
<GetUserResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
    <GetUserResult>
        <User>
            <Path>/</Path>
            <Arn>arn:aws:iam::142262769007:user/root_user</Arn>
            <UserName>root_user</UserName>
            <UserId>AIDASCH4FSVXRM7SBDZ5T</UserId>
            <CreateDate>2020-11-20T00:43:07Z</CreateDate>
        </User>
    </GetUserResult>
    <ResponseMetadata>
        <RequestId>0ee95db8-5084-4705-a8fd-5da81fd6522f</RequestId>
    </ResponseMetadata>
</GetUserResponse>
```

- In MySQL Workbench modify data.sql by replacing the generic User ID in lines 8, 9, 12, 14, and 16 with the User ID that was created above
- In line 8 also replace 'seed.dev@einfochips.com' with the username of your IAM user



```
/* **********Roles***********/
/*------------------------------------------------------------------------------------------------------------------------+*/
INSERT IGNORE INTO roles(id,name,role_type) VALUES(1,'User','ROLE_USER');
INSERT IGNORE INTO roles(id,name,role_type) VALUES(2,'Tech Support','ROLE_TECH_SUPPORT');
INSERT IGNORE INTO roles(id,name,role_type) VALUES(3,'Admin','ROLE_ADMIN');
/*------------------------------------------------------------------------------------------------------------------------+*/

INSERT IGNORE INTO users(userid,user_name,password,created_time,modified_time) values('AIDASCH4FSVXRM7SBDZ5T','root_user','$2a$10$7PtcjEnWb/ZkgyXyxY1/Iei2dGgG
INSERT IGNORE INTO userpassword_history(user_id, password,created_time,modified_time) VALUES('AIDASCH4FSVXRM7SBDZ5T','$2a$10$7PtcjEnWb/ZkgyXyxY1/Iei2dGgGQUbqI
/*------------------------------------------------------------------------------------------------------------------------+*/

INSERT IGNORE INTO user_roles(user_id,role_id) values('AIDASCH4FSVXRM7SBDZ5T',1);

INSERT IGNORE INTO user_roles(user_id,role_id) values('AIDASCH4FSVXRM7SBDZ5T',2);

INSERT IGNORE INTO user_roles(user_id,role_id) values('AIDASCH4FSVXRM7SBDZ5T',3);
/*------------------------------------------------------------------------------------------------------------------------+*/
INSERT IGNORE INTO awsuser_token(user_name,access_key_id,secret_access_key,is_root_account,create_date) values('root_user'[redacted]
/*------------------------------------------------------------------------------------------------------------------------+*/
```

	#	Time	Action	Message	Duration / Fetch
✓	12	20:13:12	INSERT IGNORE INTO roles(id,name,role_ty...	1 row(s) affected	0.062 sec
✓	13	20:13:12	INSERT IGNORE INTO roles(id,name,role_ty...	1 row(s) affected	0.063 sec
✓	14	20:13:12	INSERT IGNORE INTO users(userid,user_na...	1 row(s) affected	0.062 sec
✓	15	20:13:12	INSERT IGNORE INTO userpassword_histor...	1 row(s) affected	0.062 sec
✓	16	20:13:12	INSERT IGNORE INTO user_roles(user_id,rol...	1 row(s) affected	0.063 sec
✓	17	20:13:12	INSERT IGNORE INTO user_roles(user_id,rol...	1 row(s) affected	0.063 sec

- Execute ‘data.sql’ one time by clicking the yellow bolt ⚡

5 CONFIGURE IMAGE ON DOCKER AND EC2

5.1 Execute below commands in PuTTY

Step 1: Check “seed-server” container exists and running

```
$sudo docker ps -a
```

Step 2: Stop server & delete container (Only execute if exists & running otherwise skip it)

```
$sudo docker stop ssk-server
$sudo docker rm ssk-server
```

Step 3: Check “seed-server” image exists

```
$sudo docker images
```

Step 4: Delete image (Only execute if exists otherwise skip it)

```
$sudo docker rmi arrowelectronics/ssk
```

Step 5: Pull latest image

```
$sudo docker login -u <username> -p <password>
$sudo docker pull arrowelectronics/ssk:latest
```

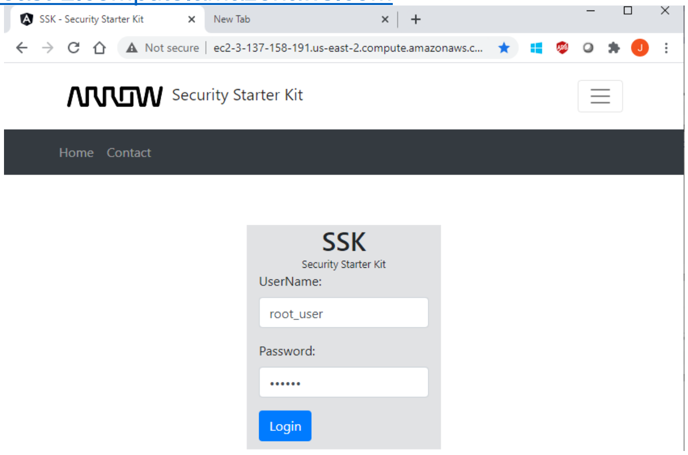
Step 6: Run image (Change highlighted text with your own values)

```
$sudo docker run –name ssk-server -e
MYSQL_URL=”jdbc:mysql://<RDS_ENDPOINT>:3306/<SCHEMA_NAME>?useSSL=false&serverTimezone=UTC&useLegacyDatetimeCode=false&allowPublicKeyRetrieval=true” -e MYSQL_UNAME=”<RDS_UserName>” -e
MYSQL_PASSWORD=”<RDS_Password>” -d -p 80:8080 -v
/home/ubuntu/seedserver:/var/lib/ arrowelectronics/ssk
```

where <SCHEMA_NAME> is the name configured in schema.sql and <RDS_ENDPOINT>, <RDS_UserName>, and <RDS_Password> are the same values that were used to initially connect to MySQL Workbench

5.2 Log into SSK Cloud Connect

- Open a web browser and enter the URL that was noted from section 4.1
  - Note this is the Public IPv4 DNS address of your EC2 Instance i.e. http://ec2-3-137-158-191.us-east-2.compute.amazonaws.com


SSK - Security Starter Kit
New Tab
Not secure
ec2-3-137-158-191.us-east-2.compute.amazonaws.c...
ARROW Security Starter Kit
Home
Contact
SSK
Security Starter Kit
Login

UserName: root_user

Password: ••••••

- Enter the UserName which is the name of the IAM username configured in ‘data.sql’ i.e. ‘root_user’
- Enter the default password: ArrowSSKportal@2020
- You can now login to your freshly installed SSK Cloud Connect Dashboard!
- Please refer to the SSK_Cloud_Connect Users Guide.pdf for configuring the different AWS services within the Arrow SSK Cloud Connect web-based tool.

6 CHECKOUT PROJECT

1. Execute following command

```
git clone ssh://<name>@git.einfochips.com:29418/secure-end-to-end-device
git checkout seed-server
```

2. Manage permission regarding project

   URL: https://git.einfochips.com:8080/q/status:open

7 BUILD PROJECT

7.1 Execute below command

1. Build Angular project (Go to /secure-end-to-end-device/seed-client)
   ng build --prod
2. Copy build file under static folder
   From path /secure-end-to-end-device/seed-client/dist/seed-client/
   To
   /secure-end-to-end-device/seed-cloud/seed-server/src/main/resources/static
3. To Build Spring boot application execute following command
   ./gradlew clean build
4. To Build image (Go to /secure-end-to-end-device/seed-cloud/seed-server)
   docker build -t arrowelectronics/ssk:latest .
5. Push image to Docker hub
   docker push arrowelectronics/ssk:latest

8 REFERENCES

[1] https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html

[2] https://docs.docker.com/engine/install/ubuntu/

[3] https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-18-04

[4] https://aws.amazon.com/console/