

Secure Starter Kit Cloud Connect Quick Start Guide

Date: March 03, 2021 | Version 1.1



The Solutions People



CONTENTS

1	Introduction	3
1.1	Purpose of the Document	3
1.2	AWS Cloud Services Descriptions and its background information	3
2	AWS Account Creation	4
2.1	Login or Create your AWS Account	4
2.2	Create New Key Pair to enable SSH access to the EC2 instance	5
3	CloudFormation code execution	6

1 INTRODUCTION

1.1 Purpose of the Document

The Cloud Connect Quick Start Guide provides an overview of How to Provision/Create and configure EC2 instance, RDS, S3 buckets and IAM User. This 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 AWS Cloud Services Descriptions and its background information

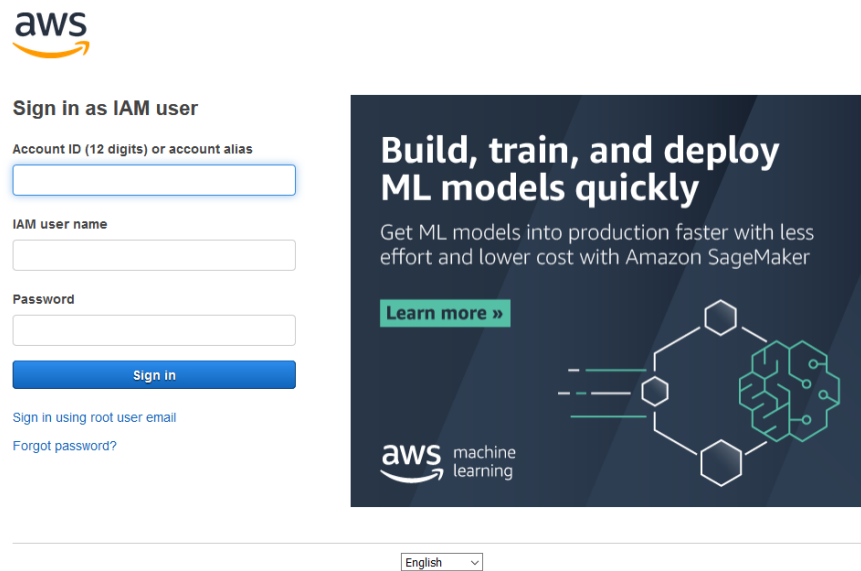
For AWS Cloud Services descriptions and its background information, follow the [SSK Cloud Connect Installation Setup Guide](#)

2 AWS ACCOUNT CREATION

2.1 Login or Create your AWS Account

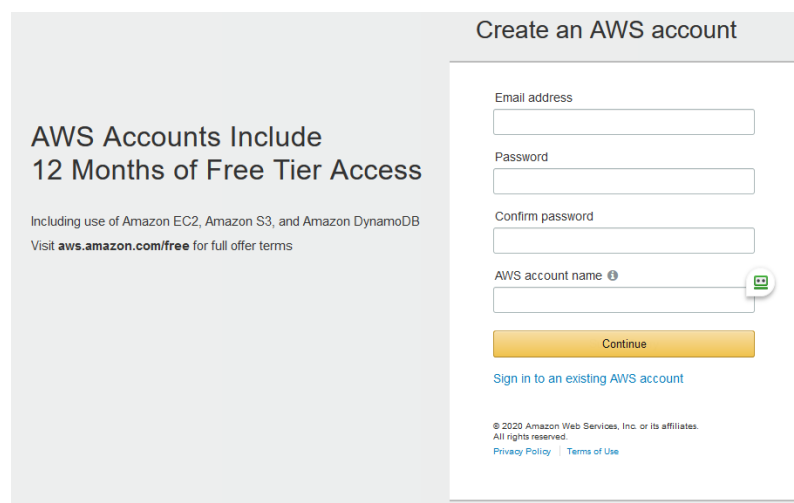
Note: If the User does not have an AWS Account, you will need to create one. This is used as the basis for the configuration of the other services required to run the demo's provided in the Security Starter Kits.

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



The screenshot shows the AWS login page. On the left, under the AWS logo, is the 'Sign in as IAM user' section. It contains three input fields: 'Account ID (12 digits) or account alias', 'IAM user name', and 'Password'. Below these is a blue 'Sign in' button. There are also links for 'Sign in using root user email' and 'Forgot password?'. On the right is a promotional banner for 'Build, train, and deploy ML models quickly' featuring the AWS Machine Learning logo and a 'Learn more »' button. At the bottom center, there is a language dropdown menu set to 'English'.

Figure 1: Login page



The screenshot shows the 'Create an AWS account' page. On the left, it states 'AWS Accounts Include 12 Months of Free Tier Access' and mentions 'Including use of Amazon EC2, Amazon S3, and Amazon DynamoDB'. It also directs users to visit 'aws.amazon.com/free' for full offer terms. On the right is a form with four input fields: 'Email address', 'Password', 'Confirm password', and 'AWS account name'. Below the form is a yellow 'Continue' button. At the bottom of the form area is a link to 'Sign in to an existing AWS account'. The footer contains copyright information: '© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved.' and links to 'Privacy Policy' and 'Terms of Use'.

Figure 2: Create New Account page

2.2 Create New Key Pair to enable SSH access to the EC2 instance

1. Please choose **AWS Console** >> **Services** >> Select **EC2** (Under Compute section) >> **Network & Security** >> **Select Key Pairs**
2. Click on “**Create key pair**” as shown in below image

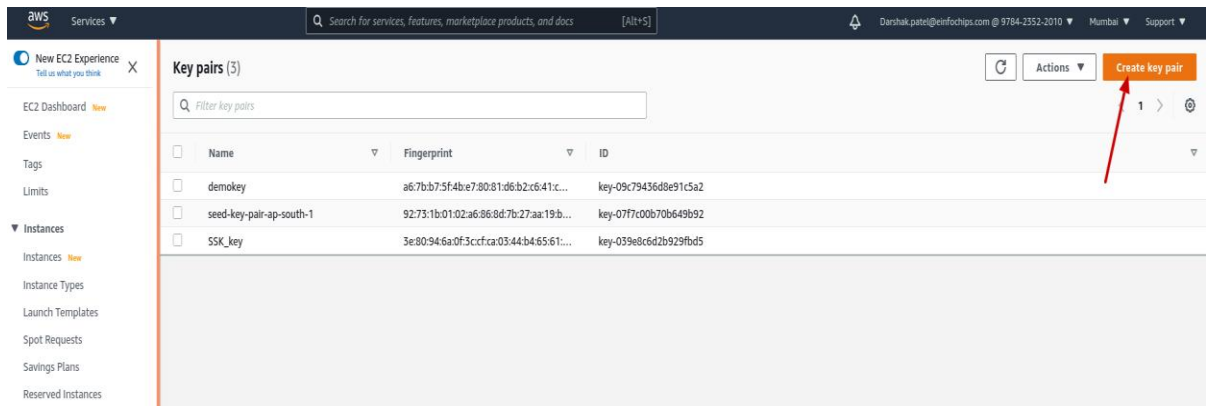


Figure 1: Create Key Pair page

3. Follow the below instructions as depicted in screenshot:
 - Enter a Name for the key pair
 - Select appropriate file format (.pem for **Linux users** and .ppk for **Windows user**) to download private key
 - Add tags (Optional)
 - Click on Create key pair

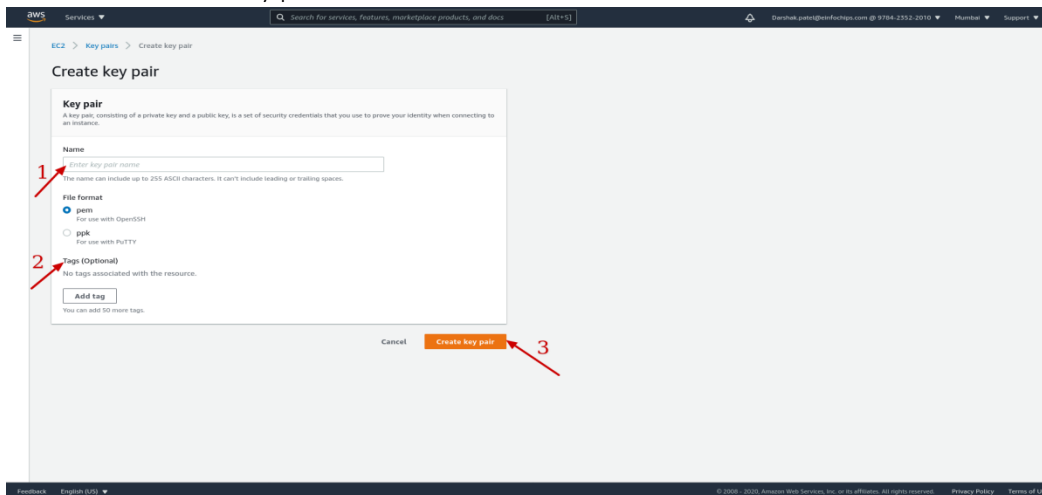


Figure 2: Creating Key Pair page

- It will Download key pair as per the file format you have selected (To Connect EC2 Instance)

Note: Keep key file at secure place, which will be used to connect Ec2 instance.

3 CLOUDFORMATION CODE EXECUTION

1. Go to the AWS console and search for the S3 services and click on it to launch as shown in the below image:

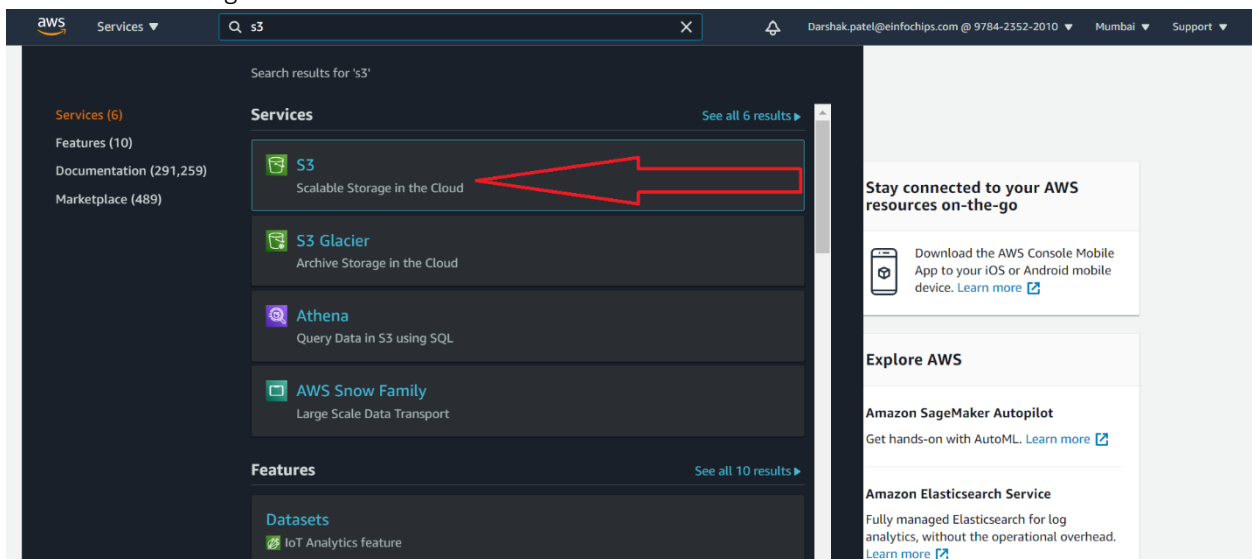


Figure 1: Searching for S3 service in Home Page

2. Click on create bucket as shown in figure below:

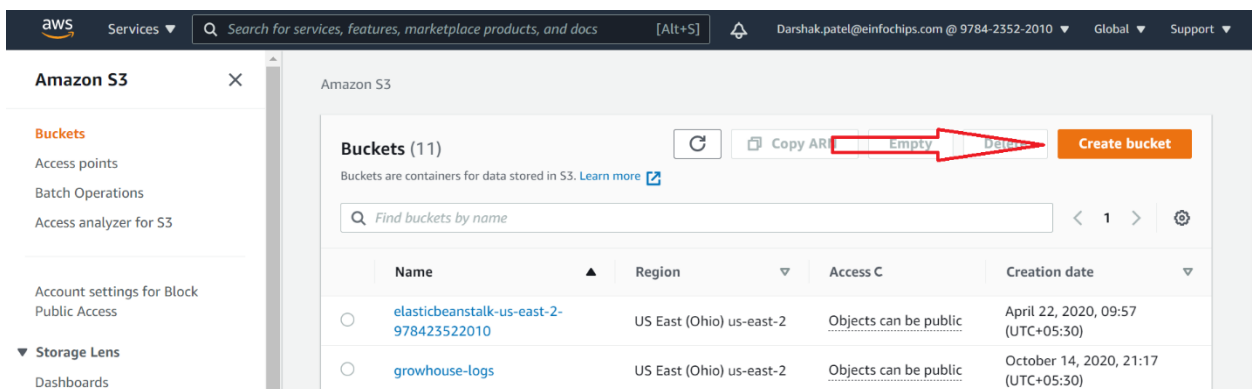


Figure 2: Create bucket

3. Enter unique bucket name after create bucket page is launched as shown below and then click on create bucket option provided at the end of the page. This will create your S3 bucket with the unique

name you provided.

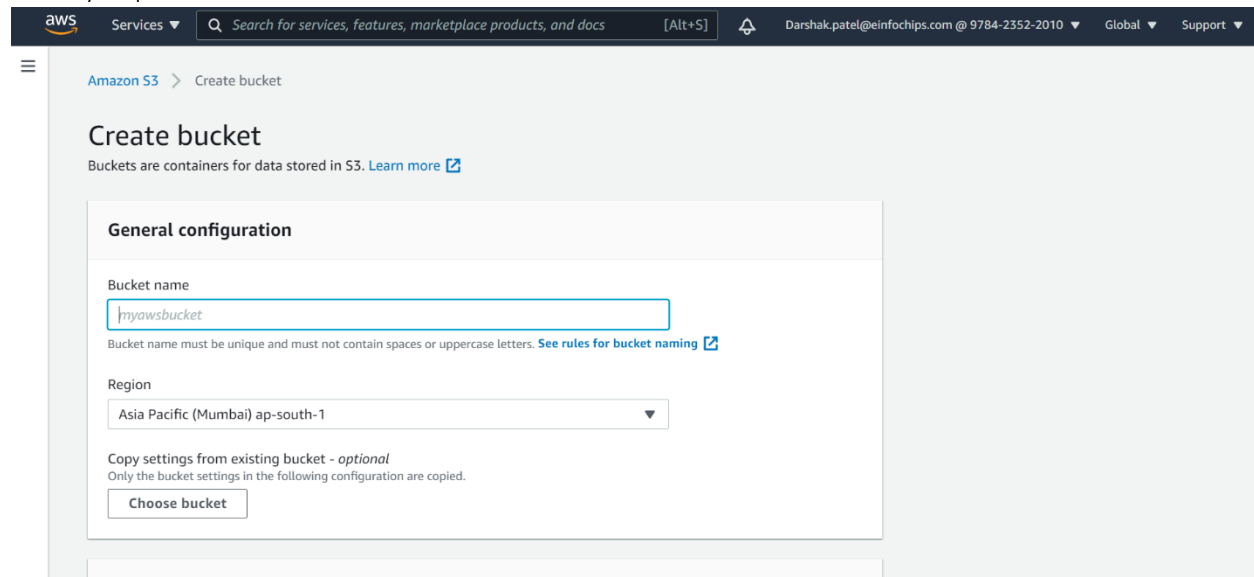


Figure 3: Creating S3 bucket

4. After creating bucket successfully, download the provided [SSK_Database.zip](#) from the Cloud-Connect-Tool branch on [GitHub](#) to upload files to the newly created S3 bucket.
5. Unzip the **SSK_Database.zip** and you will find below contents:

Name	Status	Date modified	Type	Size
<input type="checkbox"/> ec2.yaml	✓	12/17/2020 12:08 PM	YAML File	17 KB
<input type="checkbox"/> iam.yaml	✓	12/17/2020 12:10 PM	YAML File	2 KB
<input type="checkbox"/> rds.yaml	✓	12/16/2020 1:59 PM	YAML File	2 KB
<input type="checkbox"/> root.yaml	✓	12/17/2020 10:51 AM	YAML File	7 KB
<input type="checkbox"/> s3bucket.yaml	✓	12/16/2020 1:56 PM	YAML File	1 KB

Figure 4: Extracting Contents of SSK_Database.zip

6. Open the newly created S3 Bucket and choose “upload” option as shown below:

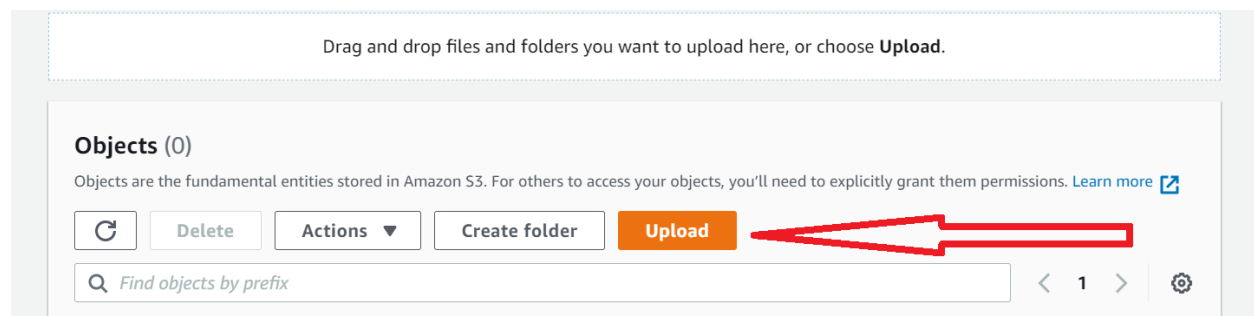


Figure 5: To upload files in S3 bucket

- Choose the “Add files” option provided in your S3 bucket and select all files except “root.yaml” from provided folder “SSK_Database”, then click on “upload” and this will upload files like shown below:

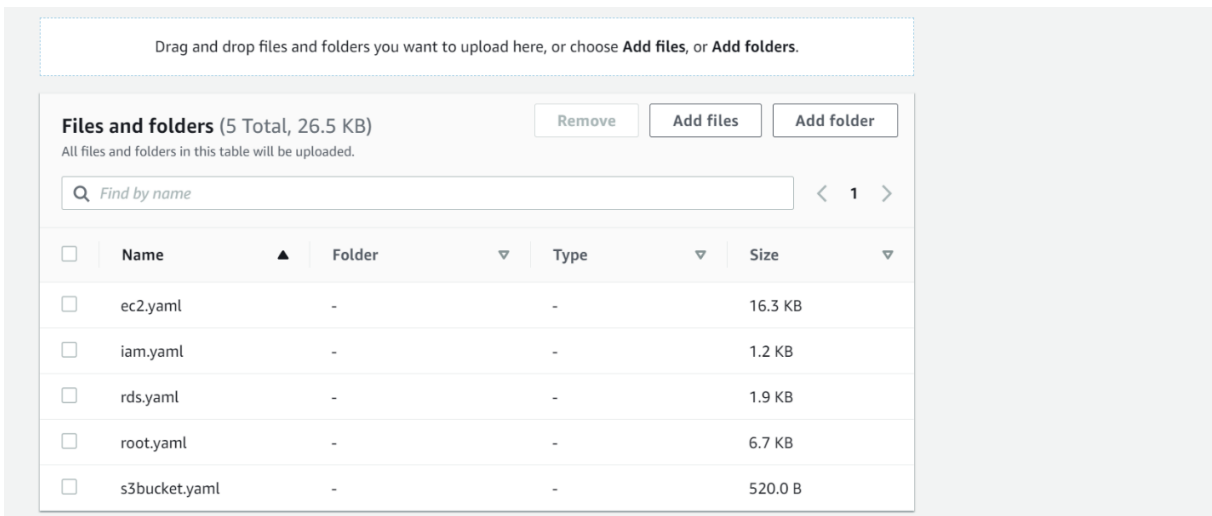


Figure 6: Uploading files in S3 bucket

- Select each of the uploaded “.yaml” files as depicted below and copy the object URLs; you will need these to modify the “root.yaml” with your new s3-bucket name.

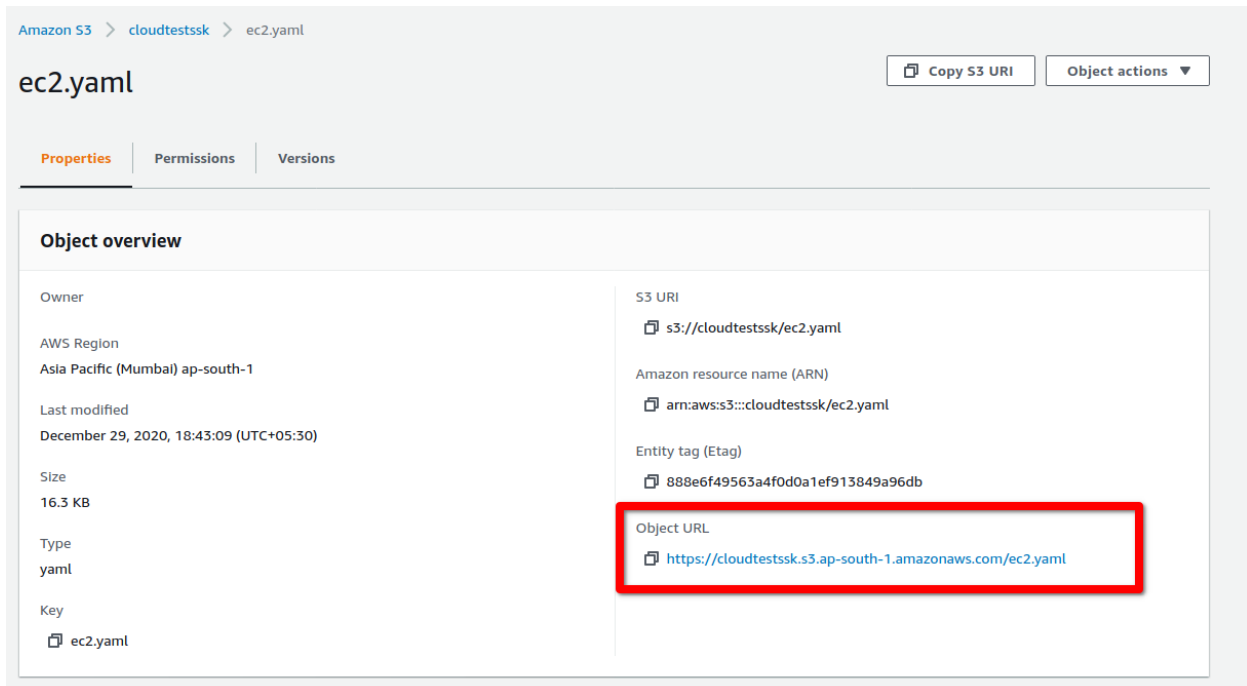


Figure 7: copying object URL

9. Please update the root.yaml file using a text editor with the copied Object URLs from above steps.

```

127 Type: AWS::CloudFormation::Stack
128 Properties:
129   TemplateURL: "https://cloudtestssk.s3.ap-south-1.amazonaws.com/rds.yaml"
130 Parameters:
131   DBInstanceID: !Ref DBInstanceID
132   DBName: !Ref DBName
133   DBInstanceClass: !Ref DBInstanceClass
134   DBAllocatedStorage: !Ref DBAllocatedStorage
135   DBUsername: !Ref DBUsername
136   DBPassword: !Ref DBPassword
137   ProjectName: !Ref ProjectName
138
139 # Create webserver (Ec2 Instance)
140 ServerStack:
141   Type: AWS::CloudFormation::Stack
142   DependsOn: ['DatabaseStack', 'IAMStack']
143   Properties:
144     TemplateURL: "https://cloudtestssk.s3.ap-south-1.amazonaws.com/ec2.yaml"
145     Parameters:
146       AWSIOTCoreEndpoint: !Ref AWSIOTCoreEndpoint
147       InstanceType: !Ref InstanceType
148       KeyName: !Ref KeyName
149       SSHLocation: !Ref SSHLocation
150       DBUsername: !GetAtt DatabaseStack.Outputs.DBUsername
151       DBPassword: !GetAtt DatabaseStack.Outputs.DBPassword
152       DBName: !GetAtt DatabaseStack.Outputs.DBName
153       DBHost: !GetAtt DatabaseStack.Outputs.DBEndpointAddress
154       IAMUserID: !GetAtt IAMStack.Outputs.UserID
155       IAMUserName: !Ref IAMUserName
156       IAMAccessKey: !GetAtt IAMStack.Outputs.AccessKey
157       IAMSecretKey: !GetAtt IAMStack.Outputs.SecretKey
158       DockerHubUsername: !Ref DockerHubUsername
159       DockerHubPassword: !Ref DockerHubPassword
160       OTABucketName: !GetAtt S3BucketStack.Outputs.OTABucketName
161       LogBucketName: !GetAtt S3BucketStack.Outputs.LogBucketName
162
163 # Create IAM Group and User
164 IAMStack:
165   Type: AWS::CloudFormation::Stack
166   Properties:
167     TemplateURL: "https://cloudtestssk.s3.ap-south-1.amazonaws.com/iam.yaml"
168     Parameters:
169       IAMUserName: !Ref IAMUserName
170       ProjectName: !Ref ProjectName
171
172 # Create S3 Bucket
173 S3BucketStack:
174   Type: AWS::CloudFormation::Stack
175   Properties:
176     TemplateURL: "https://cloudtestssk.s3.ap-south-1.amazonaws.com/s3bucket.yaml"
177     Parameters:
178       ProjectName: !Ref ProjectName
179

```

Figure 8: Updating object URL into the “root.yaml”.

10. Upload your edited root.yaml to your s3-bucket. After successfully uploading, click on the newly uploaded “root.yaml” file.

Files and folders
Configuration

Files and folders (5 Total, 26.5 KB)

Name	Folder	Type	Size	Status	Error
ec2.yaml	-	-	16.3 KB	✔ Succeeded	-
iam.yaml	-	-	1.2 KB	✔ Succeeded	-
rds.yaml	-	-	1.9 KB	✔ Succeeded	-
root.yaml	-	-	6.7 KB	✔ Succeeded	-
s3bucket.yaml	-	-	520.0 B	✔ Succeeded	-

Figure 9: Launching root.yaml page

11. Once **root.yaml** page is launched, copy **object URL** for further use in step 14 as shown below.

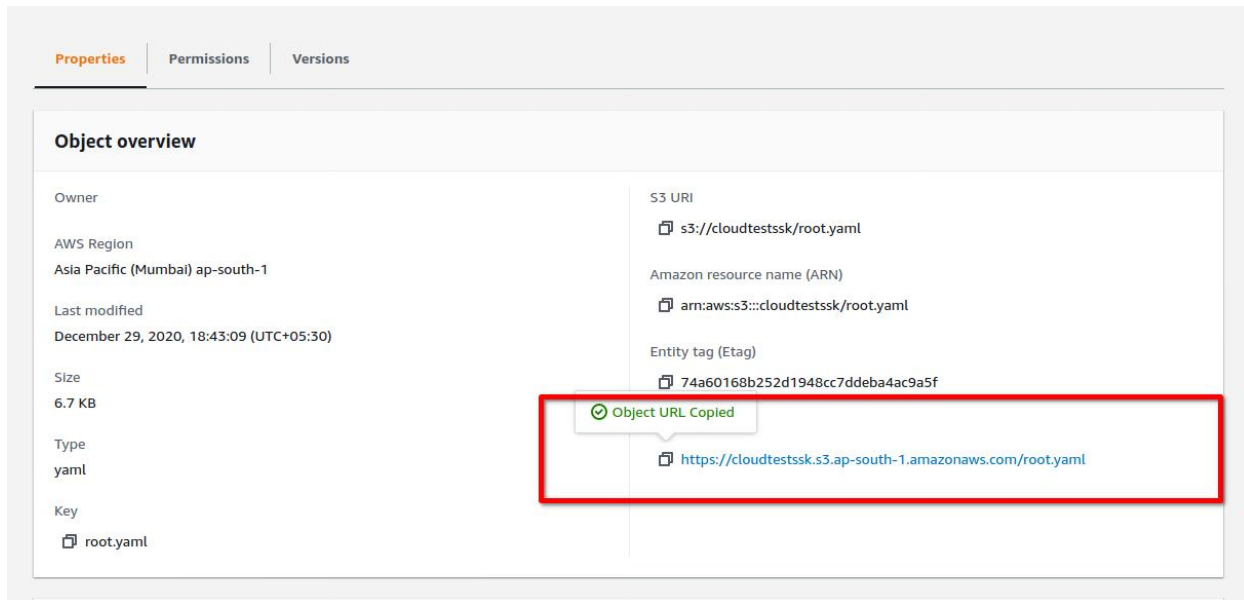


Figure 10: Copying Object URL

12. Now search for the **CloudFormation** service as shown in the below image and click on it.

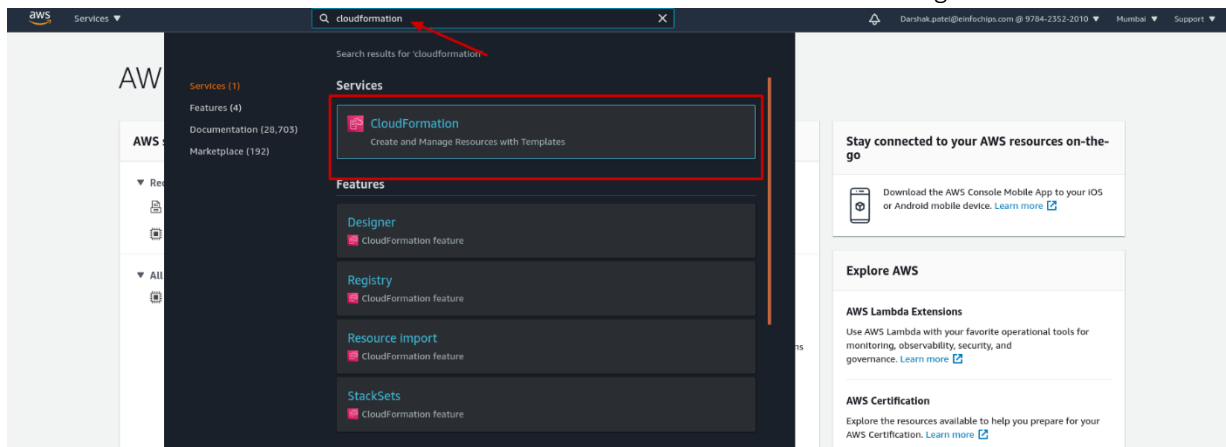


Figure 11: Searching for CloudFormation in Home Page

13. It will display page as shown below, Click on Create Stack button.

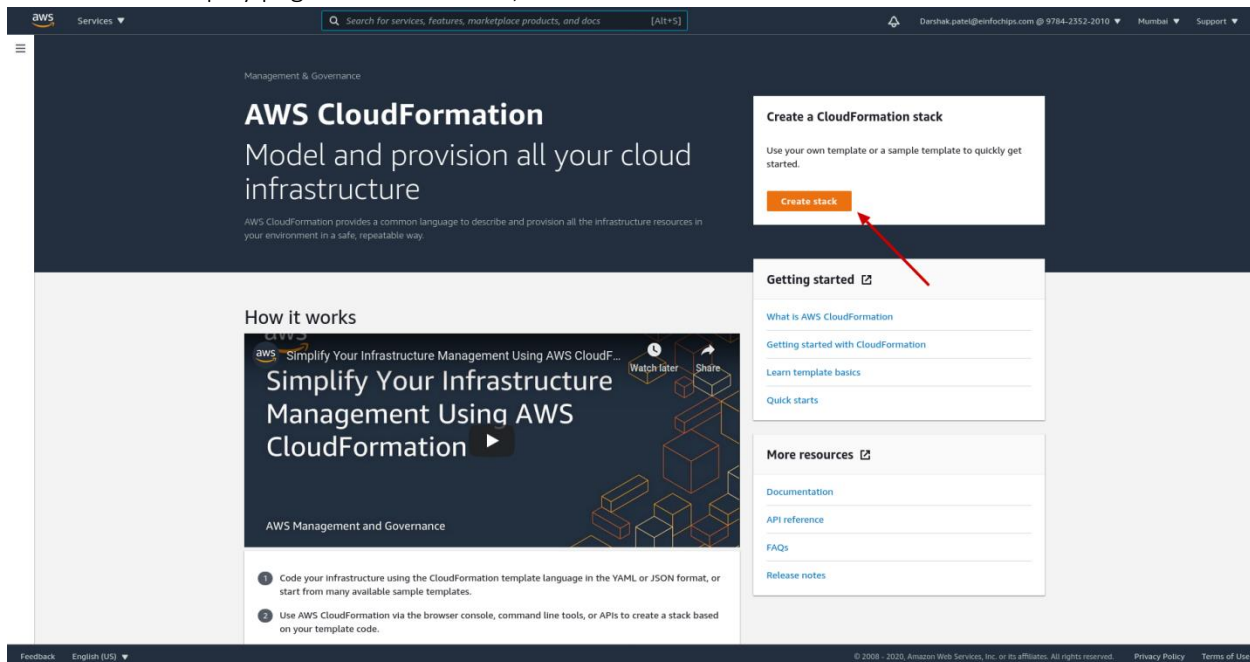


Figure 12: Create Stack

14. Enter the Object URL for your root.yaml which you have copied from step 11 in Amazon S3 URL and click on **Next** Button.

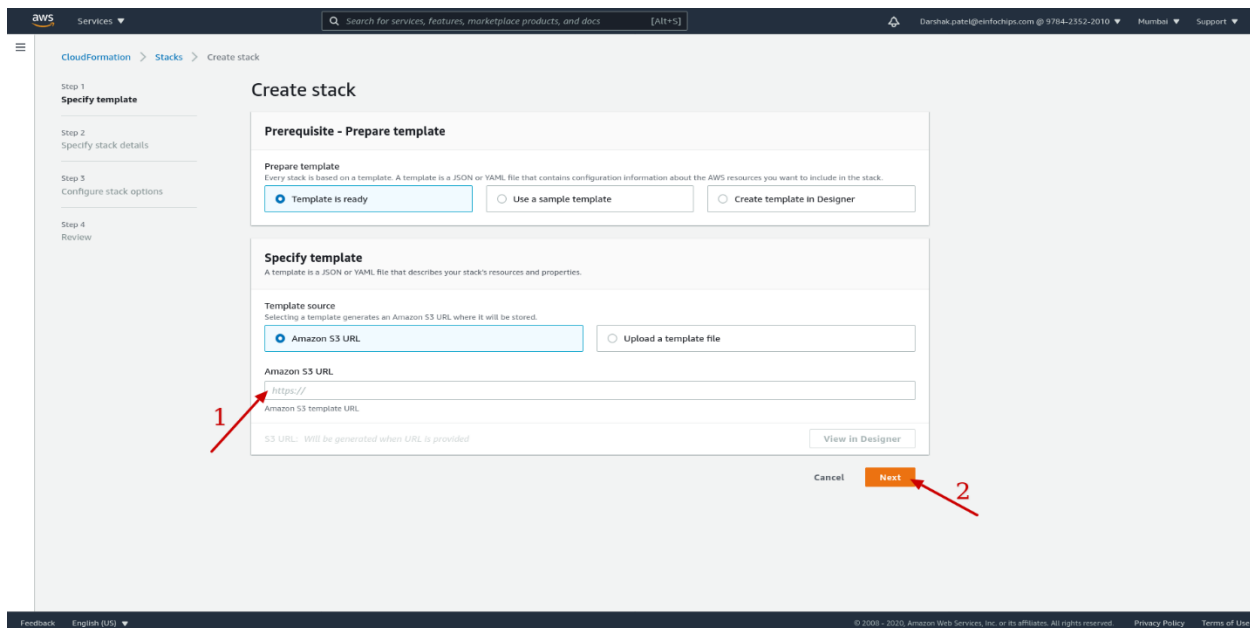


Figure 13: Creating Stack step1

15. Enter the **unique** stack **name** and fill the required parameters in the page while keeping in mind the below rules:

- In **KeyName** parameter, need to select **keypair** name which we have created in section 2.2.
- For **DBUsername** parameter, username should not contain any special characters.
- Enter **unique IAMUserName** and **ProjectName** here. Remember-repeat use of IAMUsername and ProjectName can create problem while creating stack.
- For Dockerhub username and Password, please provide below credentials:
 - **Dockerhub ID:** arrowelectronics
 - **Password:** Arrow1234
- After filling all the details, click next.

Example:

AWSIoTCoreEndpoint	xxxxxx-ats.iot.ap-south-1.amazonaws.com
DBAllocatedStorage	20
DBInstanceClass	db.t2.micro
DBInstanceID	sskdbinstance
DBPassword	einfochips123 (should be alpha-numeric)
DBUsername	admin
DockerHubPassword	Arrow1234
DockerHubUserName	arrowelectronics
IAMUserName	testusr (should be unique)
InstanceType	t2.micro
KeyName	SSK_Test
ProjectName	abcseed (should be unique)

SECURITY STARTER KIT CLOUD CONNECT QUICK START GUIDE

The screenshot shows the AWS CloudFormation console interface for creating a new stack. The breadcrumb navigation at the top indicates the path: CloudFormation > Stacks > Create stack. On the left, a sidebar shows the progress of the four-step process: Step 1 (Specify template), Step 2 (Specify stack details - currently active), Step 3 (Configure stack options), and Step 4 (Review).

The main content area is titled 'Specify stack details' and contains the following sections:

- Stack name:** A text input field with a placeholder 'Enter a stack name'. Below it, a note states: 'Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-)'.
- Parameters:** A section titled 'Parameters are defined in your template and allow you to input custom values when you create or update a stack.' containing several parameter fields:
 - AWSIoTCoreEndpoint:** A text input field with a placeholder 'Enter AWS IoT Core Endpoint'.
 - DBAllocatedStorage:** A numeric input field with a value of '20' and a unit selector set to 'GiB'.
 - DBInstanceClass:** A dropdown menu with 'db.t2.micro' selected.
 - DBInstanceID:** A text input field with a placeholder 'SSK database instance name, use lowercase letters' and the value 'sskdbinstance'.
 - DBName:** A text input field with a placeholder 'SSK database name' and the value 'sskdatabase'.
 - DBPassword:** A text input field with a placeholder 'Password MySQL database access'.
 - DBUsername:** A text input field with a placeholder 'Username for MySQL database access'.
 - DockerHubPassword:** A text input field with a placeholder 'Docker hub User Password'.
 - DockerHubUserName:** A text input field with a placeholder 'Docker hub Username'.
 - IAMUserName:** A text input field with a placeholder 'IAM User Name to create'.
 - InstanceType:** A dropdown menu with 't2.micro' selected.
 - KeyName:** A dropdown menu with a placeholder 'Name of an existing EC2 KeyPair to enable SSH access to the instances'.
 - ProjectName:** A text input field with a placeholder 'Enter prefix for s3 bucket name, use lowercase letters'.
 - SSHLocation:** A text input field with a placeholder 'The IP address range that can be used to SSH to the EC2 instances' and the value '0.0.0.0/0'.

At the bottom right of the form, there are three buttons: 'Cancel', 'Previous', and 'Next'. A red arrow points to the 'Next' button.

The footer of the console shows 'Feedback', 'English (US)', and copyright information: '© 2008 – 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.' along with links to 'Privacy Policy' and 'Terms of Use'.

Figure 14: Creating Stack step2

16. On the next page, you can optionally add tags (Tags are used for billing/cost management). Click on **Next** Button.

The screenshot shows the AWS CloudFormation console's 'Configure stack options' page. The left sidebar indicates the current step is 'Step 3: Configure stack options'. The main content area is divided into several sections: 'Tags' (with a red box highlighting the 'Key' and 'Value' input fields and an 'Add tag' button), 'Permissions' (with an 'IAM role - optional' dropdown), and 'Advanced options' (which includes expandable sections for 'Stack policy', 'Rollback configuration', 'Notification options', and 'Stack creation options'). At the bottom right, there are 'Cancel', 'Previous', and 'Next' buttons, with a red arrow pointing to the 'Next' button.

Figure 15: Creating Stack step3

17. Review the stack details (for parameters value and tags value). Then select the checkboxes for acknowledgment as shown in below image and Click on **Create Stack** Button.

CloudFormation > Stacks > Create stack

Step 1
Specify template

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Review demo

Edit

Step 1: Specify template

Template

Template URL
https://us-east-1.amazonaws.com/root.yaml

Stack description
This AWS CloudFormation Template invoke EC2, RDS, s3 and IAM templates.

[Estimate cost](#)

Step 2: Specify stack details

Parameters (14)

Key	Value
AWSIoTCoreEndpoint	a3m3zunaszckgk-atz.iot.us-east-2.amazonaws.com
DBAllocatedStorage	20
DBInstanceClass	db.t2.micro
DBInstanceId	sdkdbinstance
DBName	sdkdatabase
DBPassword	*****
DBUsername	****
DockerHubPassword	*****
DockerHubUserName	arrowelectronics
IAMUserName	test1
InstanceType	t2.micro
KeyName	demokey
ProjectName	abcseed
SSHLocation	0.0.0.0/0

Step 3: Configure stack options

Tags (1)

Key	Value
purpose	demo

No permissions
There is no IAM role associated with this stack

No stack policy
There is no stack policy defined

Rollback configuration
Monitoring time
+
CloudWatch alarm ARN
+

Notification options
No notification options
There are no notification options defined

Stack creation options
Rollback on failure
Enabled
Timeout
+
Termination protection
Disabled

Quick-create link

Capabilities

The following resource(s) require capabilities: [AWS::CloudFormation::Stack]
This template contains Identity and Access Management (IAM) resources. Check that you want to create each of these resources and that they have the minimum required permissions. In addition, they have custom names. Check that the custom names are unique within your AWS account. [Learn more](#)
For this template, AWS CloudFormation might require an unrecognized capability: CAPABILITY_AUTO_EXPAND. Check the capabilities of these resources.
☐ I acknowledge that AWS CloudFormation might create IAM resources with custom names.
☐ I acknowledge that AWS CloudFormation might require the following capability:
CAPABILITY_AUTO_EXPAND

Cancel

Previous

Create change set

Create stack

Quick Start Guide

18. It will start creating stacks for IAM User, RDS, EC2 instance and S3 Bucket. You can see the stack status and refresh the events as shown in below image.

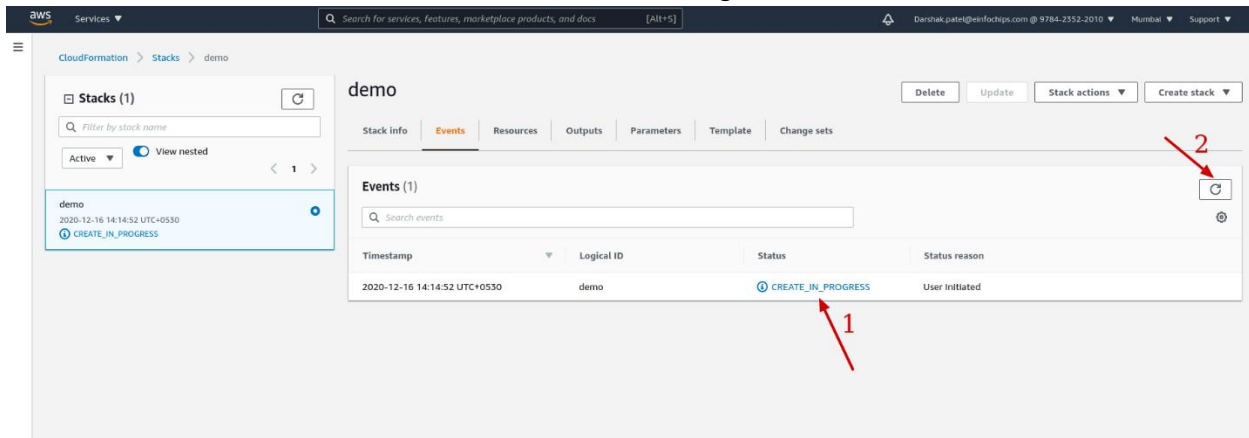


Figure 17: Stack Creation event/status page

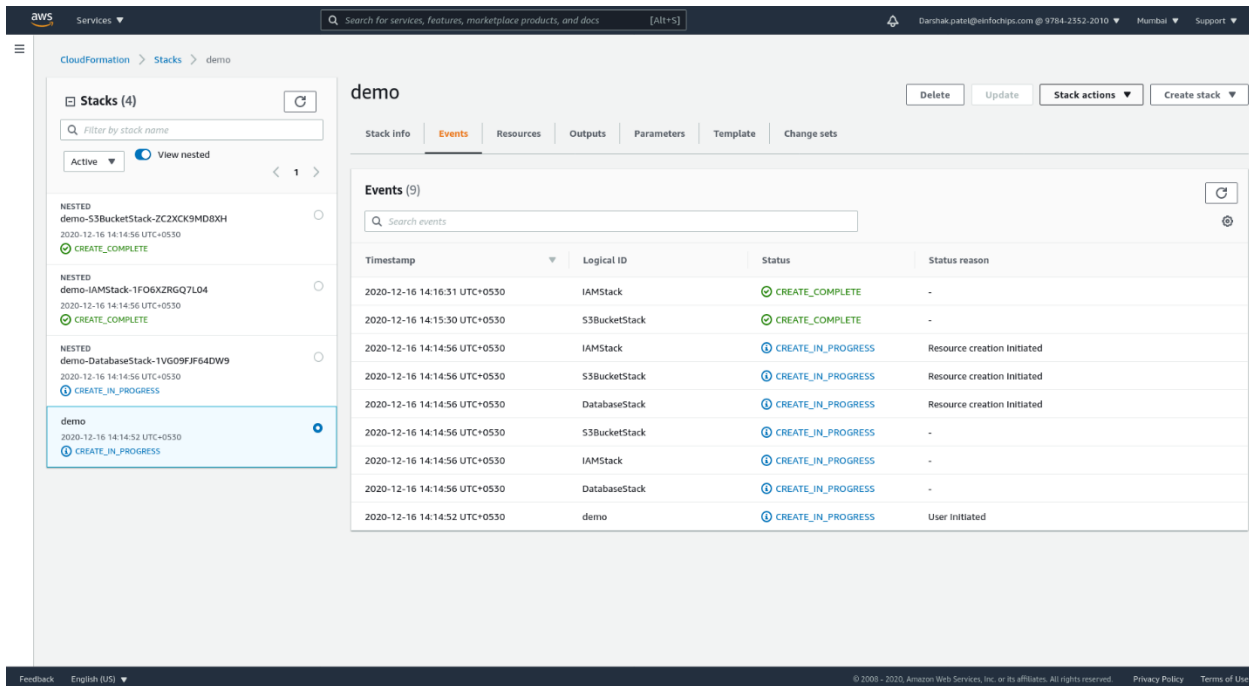


Figure 18: Stack Creation event/status page

19. After Stack creation, you can check for the website URL in the last row of the Output section.

Outputs (11)				
<input type="text" value="Search outputs"/>				
Key	Value	Description	Export name	
PublicIP	65.0.173.53	EC2 public IP	-	
RDSInstanceID	seeeedsskdbinstance	InstanceID of the newly created RDS Instance	-	
SecretKey	j1qtUQafzaWoDke APoRryUM3k+3Zsk +DHj5P85sA	the Access Key Secret	-	
UserName	kaushalava1	Master Username of DB	-	
WebsiteURL	http://ec2-65-0-173-53.ap-south-1.compute.amazonaws.com	Website URL	-	

Figure 19: Checking Website URL in output tab after Stack Creation

Note:

[Please login once with the below api link in order to provide access

<http://<ec2 domain name>/api/v1/aws/thing/configthingtypeandbucket>

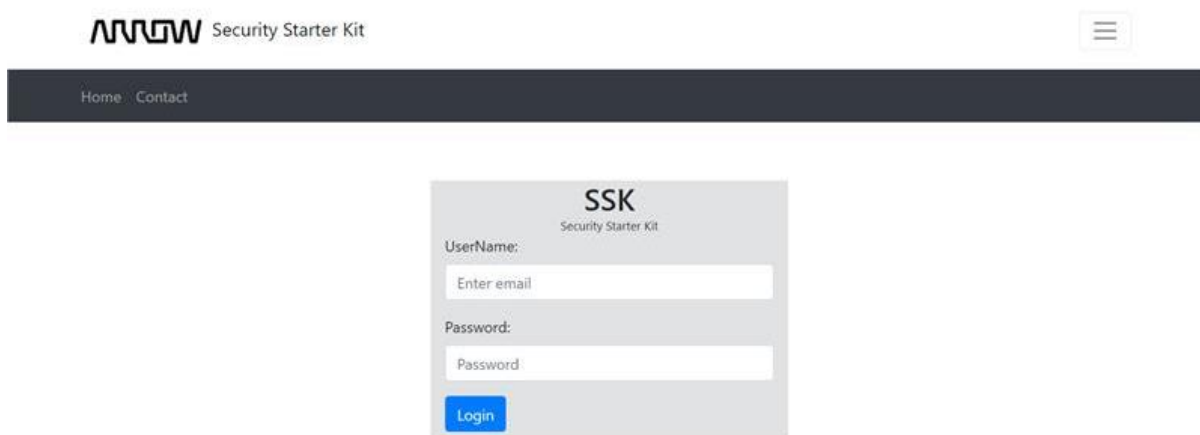
i.e.

<http://ec2-xx-xxx-xx-xx.ap-south-1.compute.amazonaws.com/api/v1/aws/thing/configthingtypeandbucket>]

User will be able to check the thing created successful page as per below.

JSON	Raw Data	Headers
Save Copy Collapse All Expand All Filter JSON		
<pre> success: true message: "Thing Type & S3 Bucket created successfully" result: null </pre>		

20. Double clicking on above **website URL**, user will be launched to the login SSK Cloud Connect Portal as shown below:



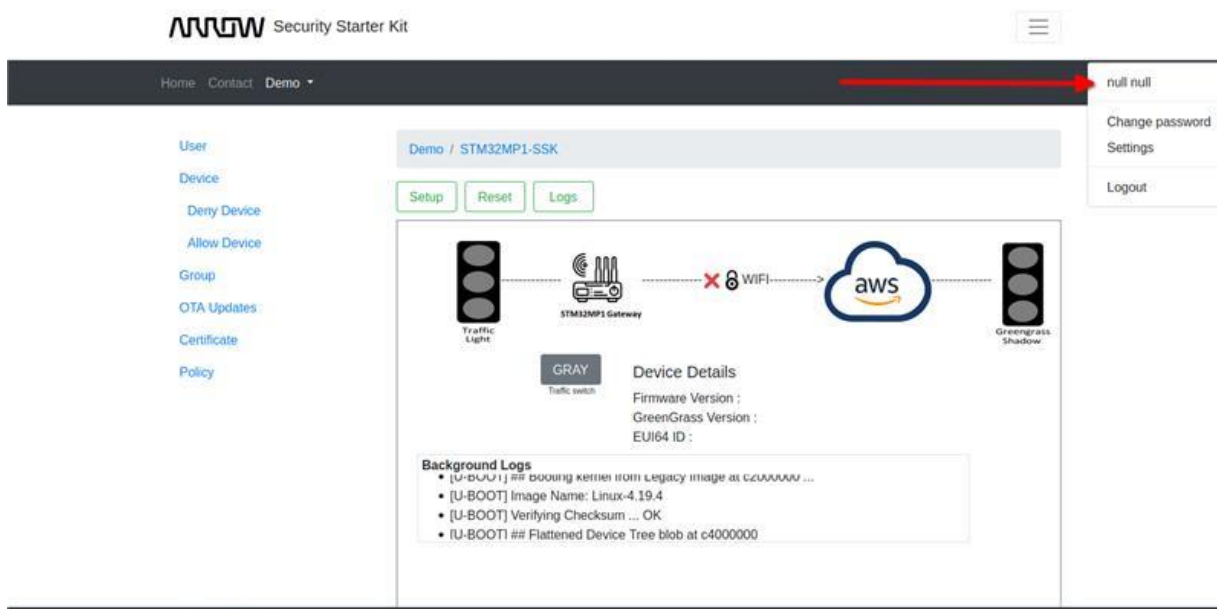
The image shows the SSK (Security Starter Kit) login page. At the top, there is a logo for 'Arrow Security Starter Kit' and a hamburger menu icon. Below the logo is a dark navigation bar with 'Home' and 'Contact' links. The main content area features a login form with the title 'SSK Security Starter Kit'. The form includes a 'UserName:' label with a text input field containing the placeholder 'Enter email', a 'Password:' label with a text input field containing the placeholder 'Password', and a blue 'Login' button.

Figure 20: SSK Login page

Note: Username: IAMUsername (User entered while creating Cloud Stack)

Password: ArrowSSKportal@2020 (Created for Temporary use only)

21. After logging in, user can also edit default username “null null” with their desired name.



The image shows the SSK Home page after login. The top navigation bar includes 'Home', 'Contact', and 'Demo'. A red arrow points to a dropdown menu in the top right corner, which contains the options: 'null null', 'Change password', 'Settings', and 'Logout'. The main content area is titled 'Demo / STM32MP1-SSK' and features three buttons: 'Setup', 'Reset', and 'Logs'. Below these buttons is a diagram showing a 'Traffic Light' icon connected to an 'STM32MP1 Gateway' icon, which is connected to an 'AWS' cloud icon, which is then connected to a 'GreenGrass Shadow' icon. A 'GRAY Traffic switch' button is also present. To the right of the diagram is a 'Device Details' section with the following information: 'Firmware Version:', 'GreenGrass Version:', and 'EU64 ID:'. Below the diagram is a 'Background Logs' section with a list of logs: '[U-BOOT] ## booting kernel from Legacy image at c2000000 ...', '[U-BOOT] Image Name: Linux-4.19.4', '[U-BOOT] Verifying Checksum ... OK', and '[U-BOOT] ## Flattened Device Tree blob at c4000000'.

Figure 21: SSK Home page

22. By clicking on “null null”, the below screen will display for the user to change their details as they want them to be displayed.

Figure 22: SSK Home/Edit profile page

23. User needs to Login into the AWS Console Account. Now search for **IOT Greengrass>>Settings>>Greengrass service role**
Now select Attach role option available there.

As we have successfully installed the cloud connect portal, please refer SSK Quick Start Guides to ensure performance of SSK Demos.