1. Create an AWS account
2. Set up the flask environment. Navigate to the directory of the flask website.
3. Create a virtual environment (e.g. virtualenv flask-aws)
4. Activate the virtual environment (source flask-aws/bin/activate)
5. Install the packages needed for the website (e.g. pip install -r requirements.txt)
6. Set up the Elastic Beanstalk Environment (pip install awsebcli)
7. Create a new user by going to the AWS console. Search for Identity and Access Management (IAM) and select “Users”)

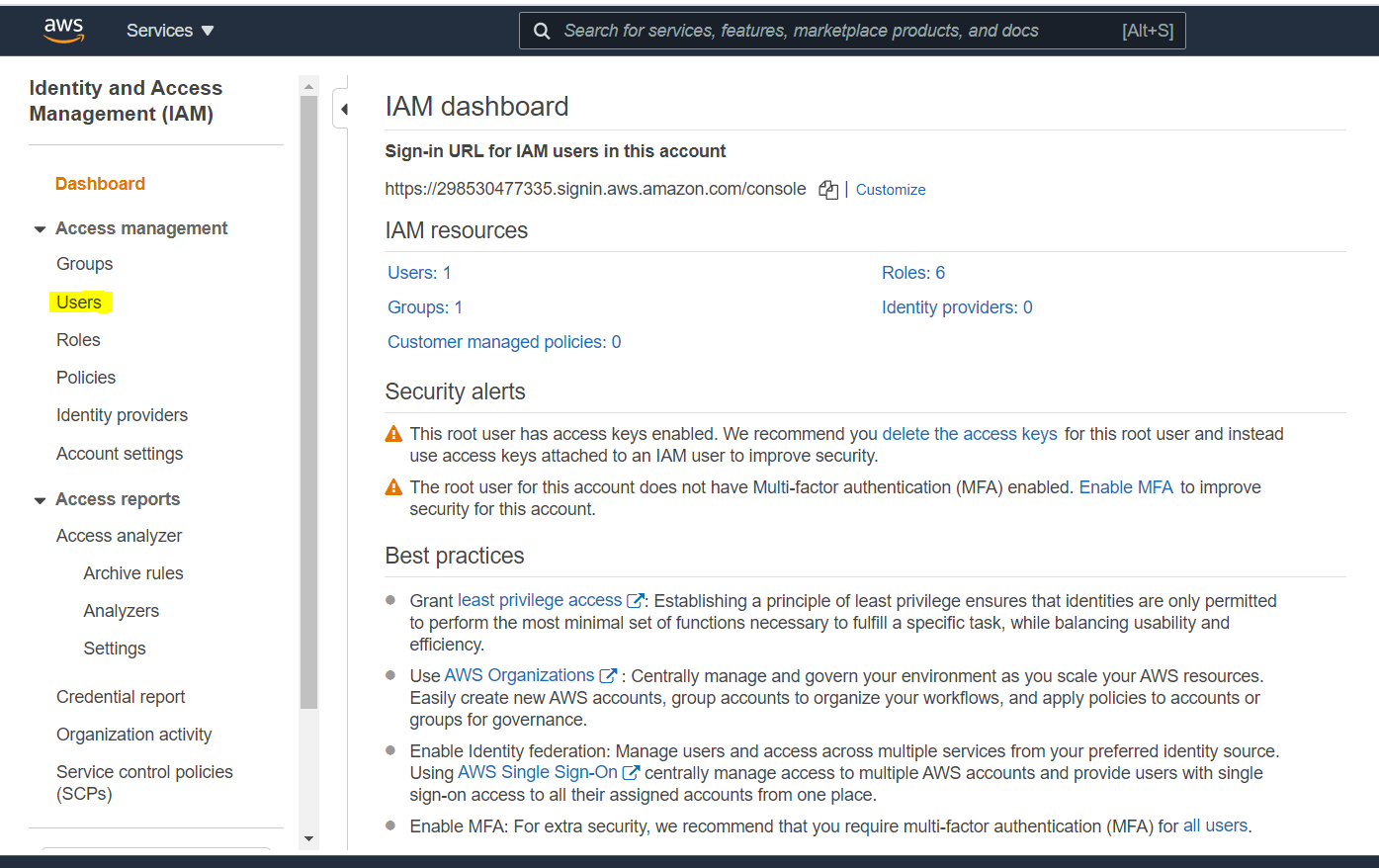


Figure 1

1. Select “Add User”

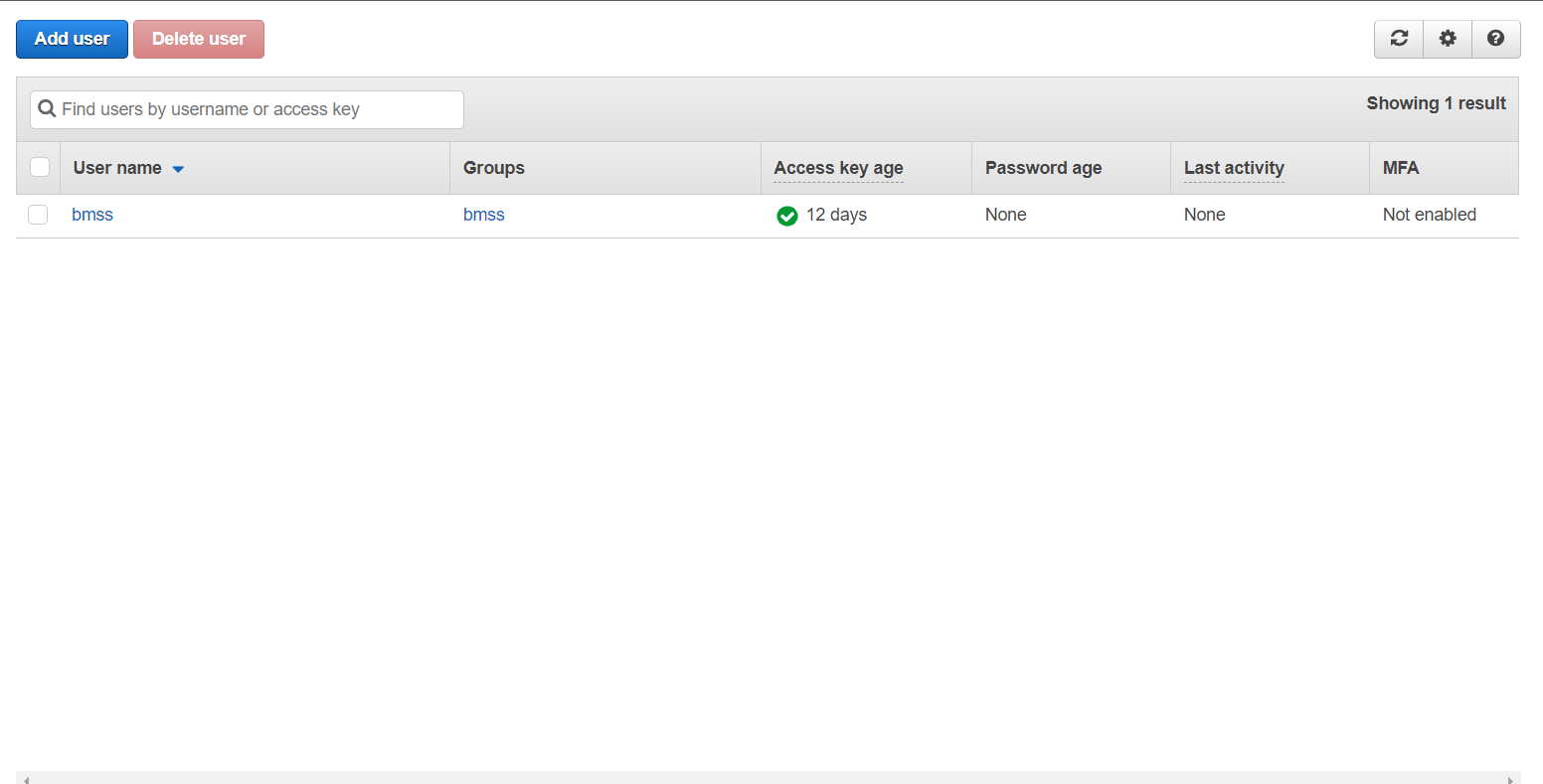


Figure 2

1. Enter a username and check “Programmatic Access”. Click “Next Permissions”.

Graphical user interface, text, application, email

Description automatically generated

Figure 3

1. The user has been created but he currently has no permissions. To grant admin access, the user would have to be assigned to a group. Select “Create group”.

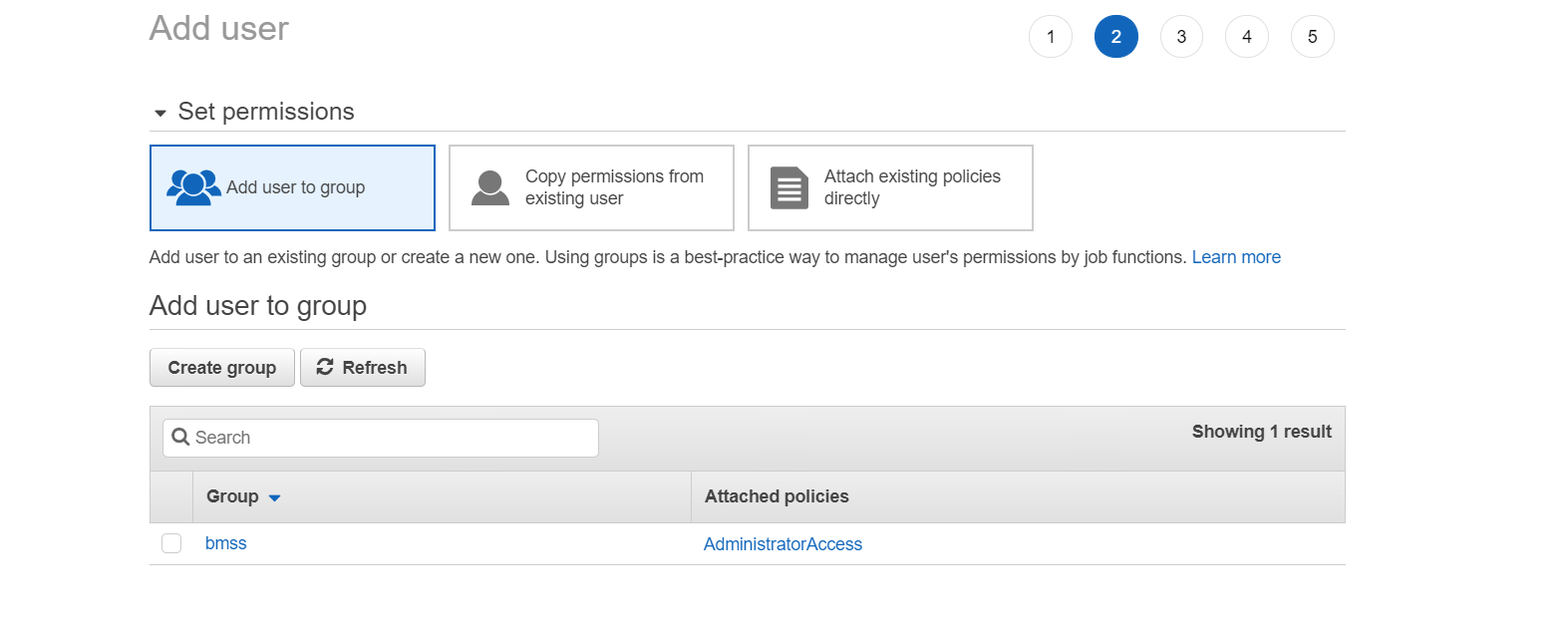


Figure 4

1. Select “Administrator Access” and create group.

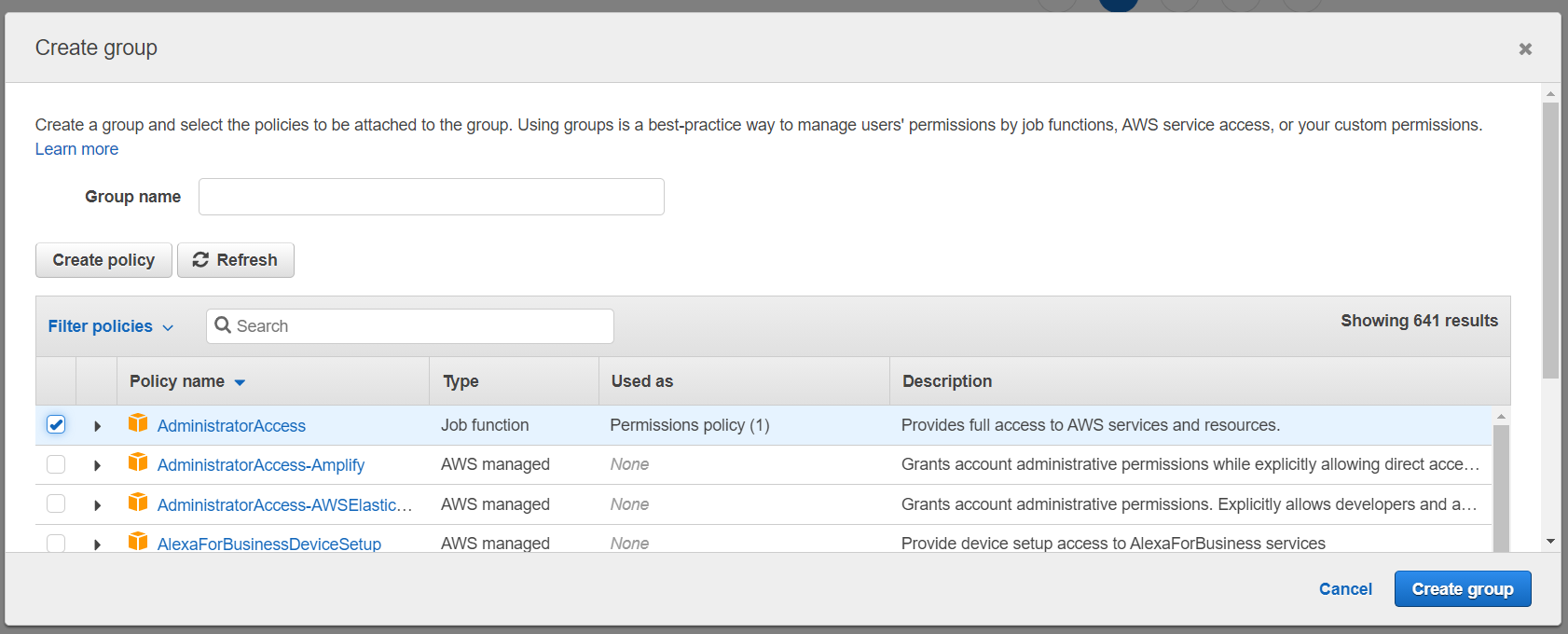


Figure 5

1. Click “Next” and create the user.
2. Download and take note of the Access key ID and Secret Access Key

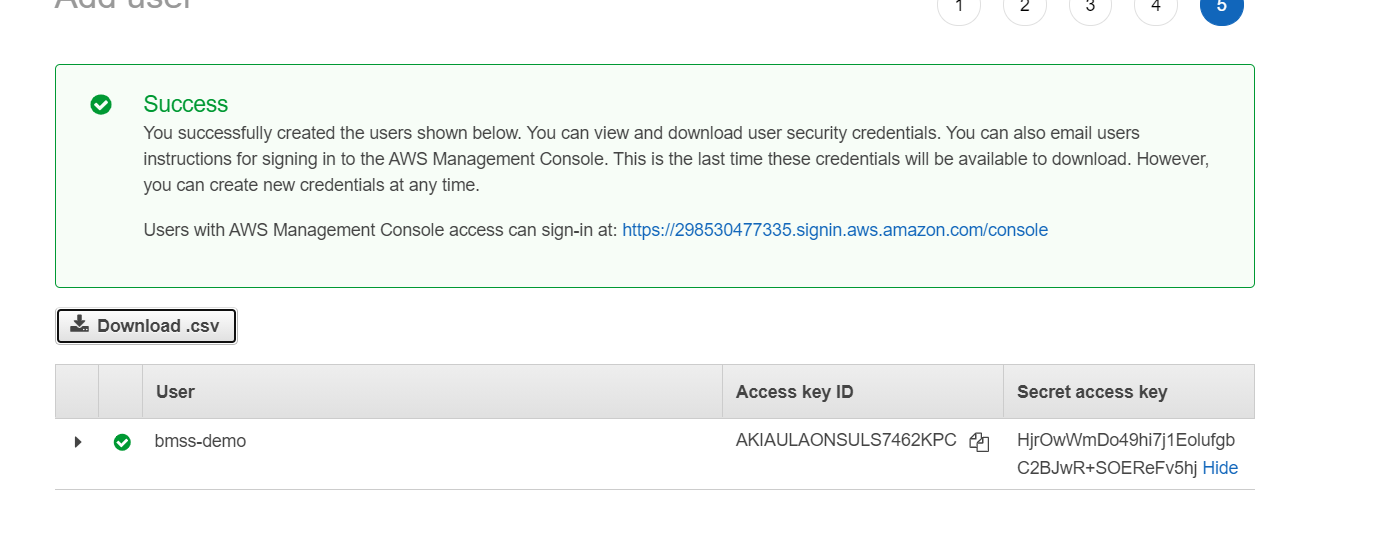


Figure 6

1. Initialize the Elastic Beanstalk environment. In the command line interface enter “eb init”.
2. You will see a list of regions. Select the region closest to your location.

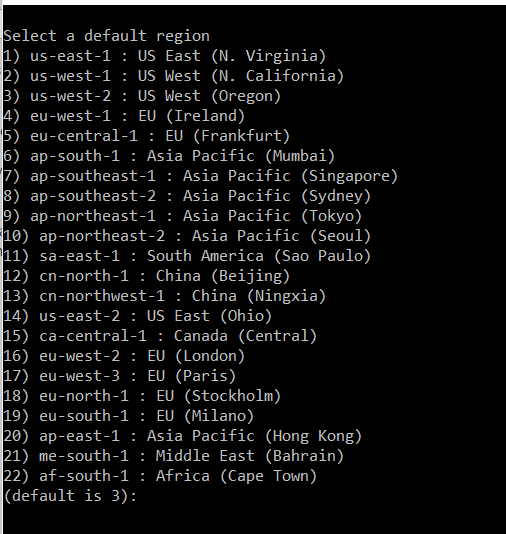


Figure 7

1. You will then be prompted for the AWS Access Key ID and Secret key from **step 13**. Enter these into the terminal.
2. Next you will see:
   1. **Select an application to use  
      1) [ Create new Application ]  
      (default is 1): 1**
3. Press enter to go to the next screen.
4. You will then see:
   1. **Enter Application Name  
      (default is “flask-aws-tutorial”):**
5. Press enter to use the default name or change it as you see fit.
6. You will then see:
   1. **It appears you are using Python. Is this correct?  
      (y/n):**
7. Enter “y” and press the enter key.
8. Select your python version.
   1. **Select a platform version.  
      1) Python 3.4  
      2) Python 2.7  
      3) Python  
      4) Python 3.4 (Preconfigured — Docker)  
      (default is 1):**
9. You will then see:
   1. **Do you want to set up SSH for your instances?  
      (y/n):**
10. Enter “n” and press the enter key. (This setting can be changed later if there is a need to ssh into the instance later).
11. In the command line, enter “eb create”.
12. We then need to create an environment name and DNS CNAME for the app. The DNS CNAME will show up in the **url** and will have to be unique (i.e. bmss-test-dev.elasaticbeanstalk.com).
    1. **Enter Environment Name  
       (default is bmss-test-dev):  
       Enter DNS CNAME prefix  
       (default is bmss-test):**
13. After selecting the DNS CNAME, there will be status updates as the app is deployed. After deployment is complete, you should see:
    1. **INFO: Application available at bmss-test-dev.elasticbeanstalk.com.  
       INFO: Successfully launched environment: bmss-test-dev**
14. The website can then be found at the first link.
15. Updating: Whenever a file is updated, you can deploy the changes with “eb deploy”.