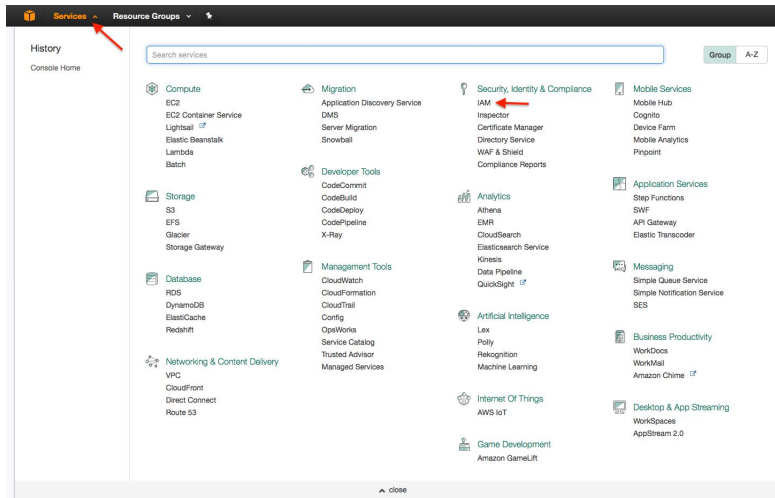


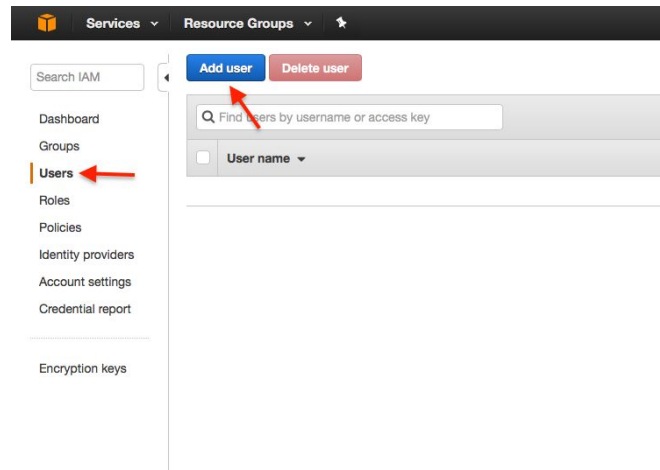
## AWS Administration

# Creating Users/Groups/Policies

1. Sign into AWS with root account (Assuming no other admin users have been created)
2. Click the Services tab and then click the IAM link



3. Click the Users link in the left hand navigation links, and then click Add User



### 3.1. Enter in User details, and then click Next

The screenshot shows the AWS IAM 'Add user' console. At the top, there's a navigation bar with 'Services' and 'Resource Groups'. Below it, the 'Add user' title is followed by a progress indicator with four steps: 1. Details (active), 2. Permissions, 3. Review, and 4. Complete. The 'Set user details' section includes a 'User name\*' field with the value 'orion' and a link to 'Add another user'. Below this, the 'Select AWS access type' section has a 'Learn more' link. The 'Access type\*' section has two options: 'Programmatic access' (checked) and 'AWS Management Console access' (checked). The 'Console password\*' section has two options: 'Autogenerated password' (checked) and 'Custom password' (unchecked). The 'Require password reset' section has a checkbox that is unchecked. At the bottom right, there are 'Cancel' and 'Next: Permissions' buttons.

3.1.1. Access type may vary depending on the type of user you're creating. In this case, this user will have programmatic access and AWS Management Console access.

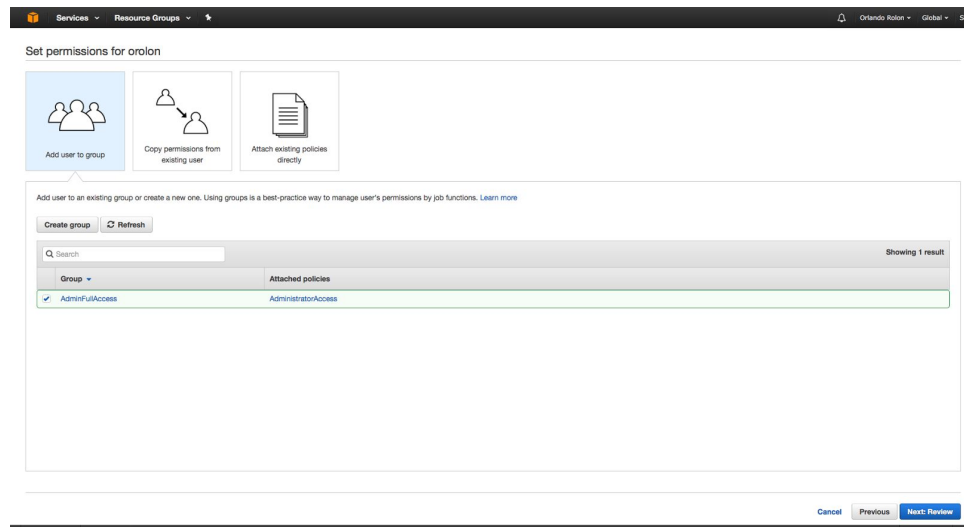
### 3.2. Click Create group (Assuming this is a fresh AWS account)

3.2.1. Enter a name for the group, assign relevant policies, and then click Create Group

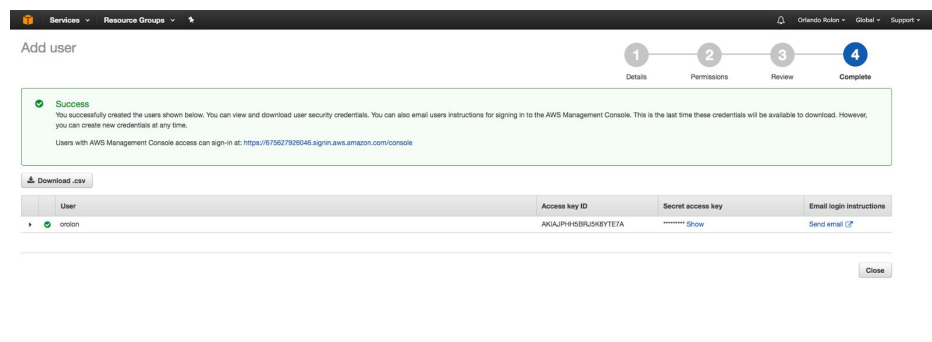
This screenshot is identical to the one above, showing the 'Set user details' step in the AWS IAM 'Add user' console. It displays the 'User name\*' field with 'orion', the 'Access type\*' section with 'Programmatic access' and 'AWS Management Console access' checked, the 'Console password\*' section with 'Autogenerated password' checked, and the 'Require password reset' checkbox unchecked. The 'Next: Permissions' button is visible at the bottom right.

3.2.1.1. In this instance, an AdminFullAccess group was created. This group is primarily for the account owner to conduct admin duties without using the root account.

3.3. Click Next:Review, then click Create User on the following screen.



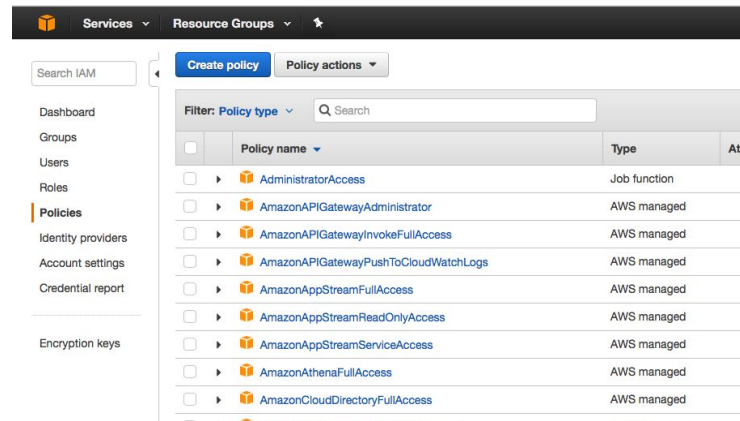
3.4. Within the success prompt, a url is provided which will be the URL that your created user will be able to sign into the AWS account (if AWS management console access was given).



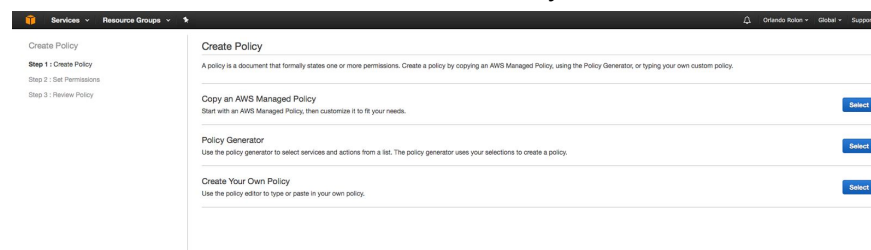
3.4.1. Within this screen, it is recommended you send an initial email using the link. Also download the .csv, which will contain the user's credentials and securely forward them their credentials.

# Creating Elastic Beanstalk Developer Policy

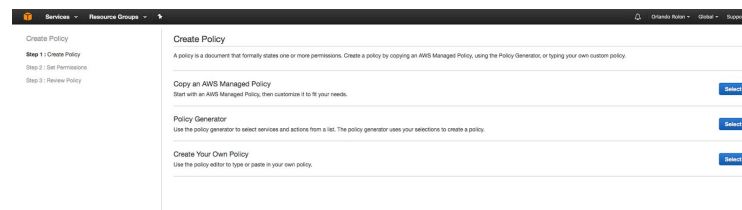
1. Within IAM services section, click Policies and then click Create Policy



2. Click the select button for Create Your Own Policy



- 2.1. Fill out the Review Policy Form, Click Validate Policy, and then click Create Policy



- 2.1.1.1. Amazon's Example Policies Based on Resource Permissions guide was used in creating policies that pertain to creating a User Group for Developers which have permissions to develop, test, and deploy a specific Elastic Beanstalk application instance.

Please see

<http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/AWSHowTo.iam.example.resource.html> for reference.

# EB Initialization/Deployment

1. Install the Elastic Beanstalk Command Line Interface (EB CLI)
  - 1.1. <http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/eb-cli3-install.html>
2. Navigate to the root folder of the application (must be a git repo)
3. Configure EB CLI by typing eb init
  - 3.1. <http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/eb-cli3-configuration.html>

```
(OLY ~) $ cd dev/hhp/
(OLY (dev) hhp) $ eb init

Select a default region:
1) us-east-1 : US East (N. Virginia)
2) us-west-1 : US West (N. California)
3) us-west-2 : US West (Oregon)
4) eu-west-1 : EU (Ireland)
5) eu-central-1 : EU (Frankfurt)
6) ap-south-1 : Asia Pacific (Mumbai)
7) ap-southeast-1 : Asia Pacific (Singapore)
8) ap-southeast-2 : Asia Pacific (Sydney)
9) ap-northeast-1 : Asia Pacific (Tokyo)
10) ap-northeast-2 : Asia Pacific (Seoul)
11) sa-east-1 : South America (Sao Paulo)
12) cn-north-1 : China (Beijing)
13) us-east-2 : US East (Ohio)
14) ca-central-1 : Canada (Central)
15) eu-west-2 : EU (London)
((default is 3): 1)
You have not yet set up your credentials or your credentials are incorrect
You must provide your credentials.
(aws-access-id): AKIAJFVDUM5GBLDCTG2A
(aws-secret-key): dzXPX0vpp47StVHucdkeHYVjKy1K6w4bFVA7HEKS

Select an application to use
1) HHP
2) [ Create new Application ]
((default is 2): 1)
Select the default environment.
You can change this later by typing "eb use [environment_name]".
1) dev-env
2) HHP-dev
((default is 1): 2)
Note:
Elastic Beanstalk now supports AWS CodeCommit; a fully-managed source control service. To learn more, see Docs: https://aws.amazon.com/codecommit/
Do you wish to continue with CodeCommit? (y/n) (default is n): n
(OLY (dev) hhp) $
```

4. To deploy the application, type eb deploy.
  - 4.1. When running the deploy command, it will deploy the current state of the git repository (changes that have not been committed will not be included).