USE CASE – 2

Scenario: -

Usually an organization has different stages/environments/group like Development, Quality Assurance (QA), Pre-Production and Production in the application life cycle. And there are multiple users in each of these Groups. For ex., in the Development Group there might be 10 Developers, 5 Testers in QA and so on.

When a resource like an EC2 Instance or any other AWS resource is created for the Production Group then only the members in the Production Group need to have the access to that particular EC2 Instance and no one else. The same is the case with the other Groups also, the respective members should have access to only the Instances they are authorized to.

How should we start: -

The same can be achieved via IAM. The 600 odd policies which come with IAM doesn't meet this requirement, so in this use case we would be creating a custom policy to meet the above requirement, assign it to an IAM User, create a set of EC2 Instances. And we will also ensure that the above policy works via the EC2 and the CloudTrail Console. All of these is achieved by tagging the EC2 Instances appropriately.

Services we will work with: -

EC2, IAM and CloudTrail

How to start: -

1. we would be setting up a VPN Endpoint in the Default VPC. And create a MyAppVPC along with an EC2 instance in the private subnet.

2. Finally, we would be establishing a VPN connection to the VPN Endpoint and connect to the EC2 instance using the private IP address from our Laptop.

3. Along the way we also need to setup a Peering connection across the two VPCs.

USE CASE – 2

Follow Along: -

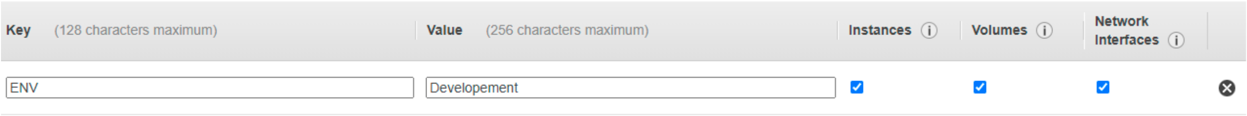
- **Step: - 1**

  Create 2 EC2 instances. The OS the EC2 instances doesn’t matter. But try to create t2.micro as they fall under the free tier.

  Select one of the EC2 instance. Go to the Tags tab and add a Tag with the Env as the Key and Production as the Value. This is to designate that this EC2 instance belongs to the Production instance.

- **Step: - 2**

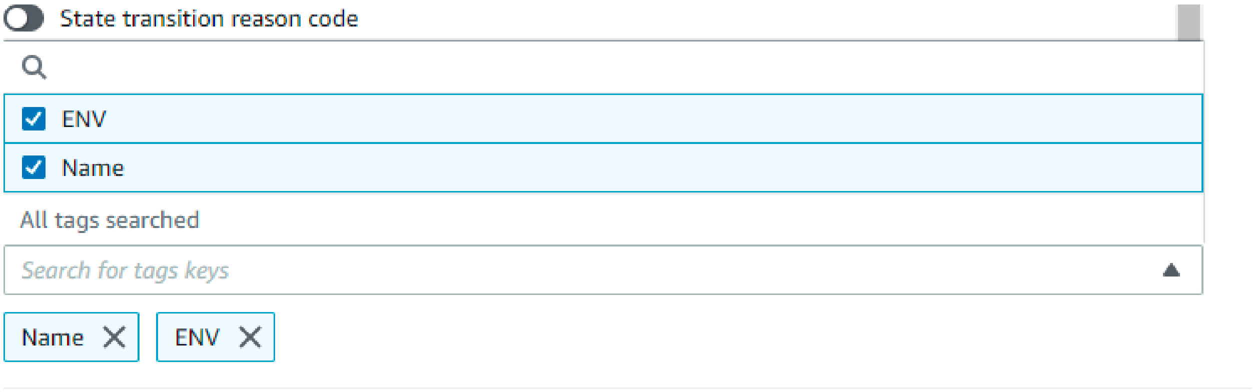
  Do the same with the other instance, but Tag it as a Development for the Value.



Key (128 characters maximum)	Value (256 characters maximum)	Instances	Volumes	Network Interfaces	
ENV	Developement	☑	☑	☑	⊗

- **Step: - 3**

  Click on the Gear button on the top right of the EC2 screen to “Show/Hide Columns” and select Env as the column to be displayed.



○ State transition reason code

☑ ENV

☑ Name

All tags searched

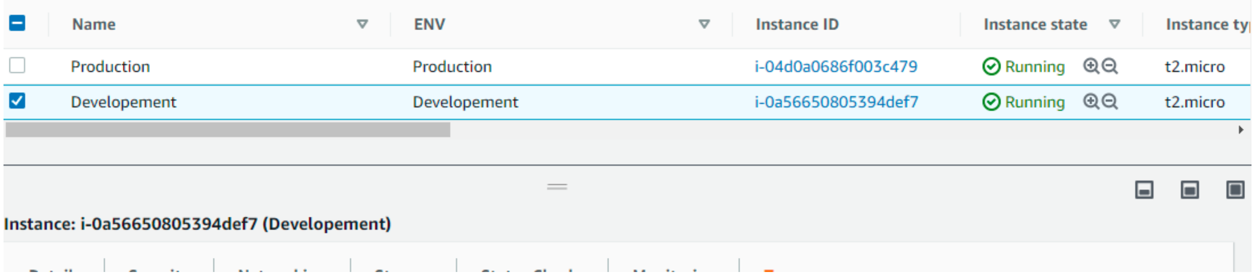
Search for tags keys

Name ✕ ENV ✕

USE CASE – 2

- **Step: -4**

  Now the EC2 instances can be easily identified as the Production and Development instances as shown below.

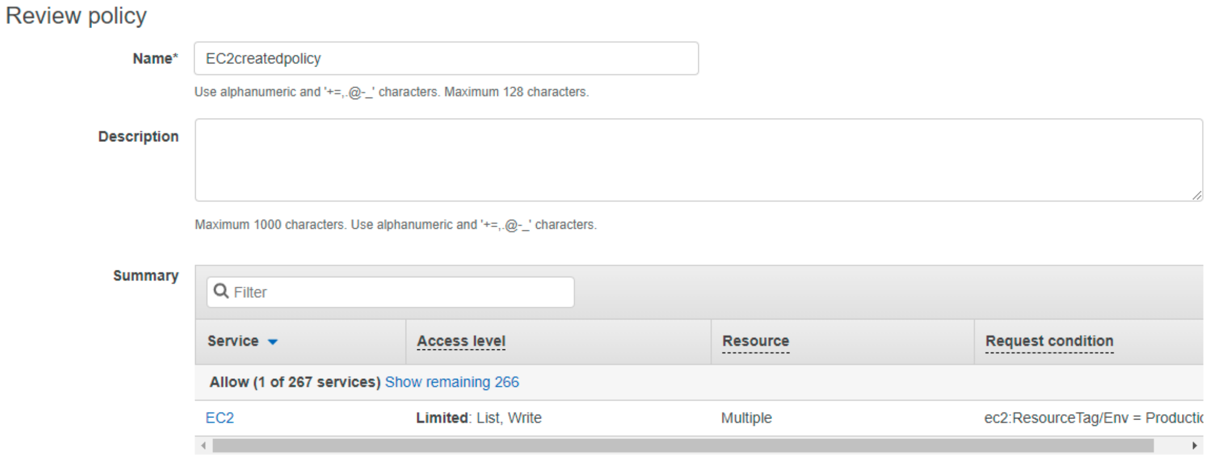


▬	Name	ENV	Instance ID	Instance state	Instance ty
□	Production	Production	i-04d0a0686f003c479	✓ Running	t2.micro
☑	Developement	Developement	i-0a56650805394def7	✓ Running	t2.micro

Instance: i-0a56650805394def7 (Developement)

- **Step: - 5**

  Go to the IAM Management Console, click on Policies and again click on "Create policies".



Review policy

Name* EC2createdpolicy

Use alphanumeric and '+=,.@-_' characters. Maximum 128 characters.

Description

Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

Summary

Filter

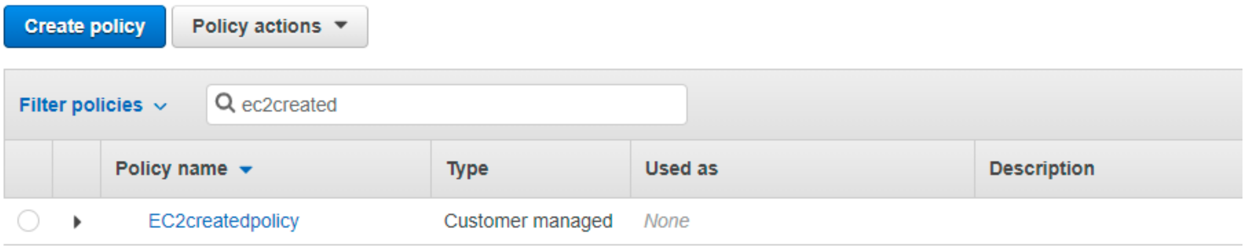
Service	Access level	Resource	Request condition
Allow (1 of 267 services) Show remaining 266			
EC2	**Limited**: List, Write	Multiple	ec2:ResourceTag/Env = Productic

USE CASE – 2

- **Step: - 6**

  Click on the JSON Tab and paste the below JSON and click on Review Policy. This policy gives the user permissions to Start/Stop the EC2 instances with the Tag Env equals Production. And also, the user has permission to describe the EC2 details for any of the EC2 instance.

```
{
        "Version": "2012-10-17",
        "Statement": [
        {
                "Effect": "Allow",
                "Action": [
                "ec2:StartInstances",
                "ec2:StopInstances"
],
        "Resource": "arn:aws:ec2:*:*:instance/*",
        "Condition": {
        "StringEquals": {
        "ec2:ResourceTag/Env": "Production"
                                }
                        }
        },
{
                "Effect": "Allow",
                "Action": "ec2:DescribeInstances",
                "Resource": "*"
}
]
}
```



		Policy name ▾	Type	Used as	Description
○	▸	EC2createdpolicy	Customer managed	None	

USE CASE – 2

- **Step: - 7**

  Give the policy a name and click on "Create policy".

- **Step: - 8**

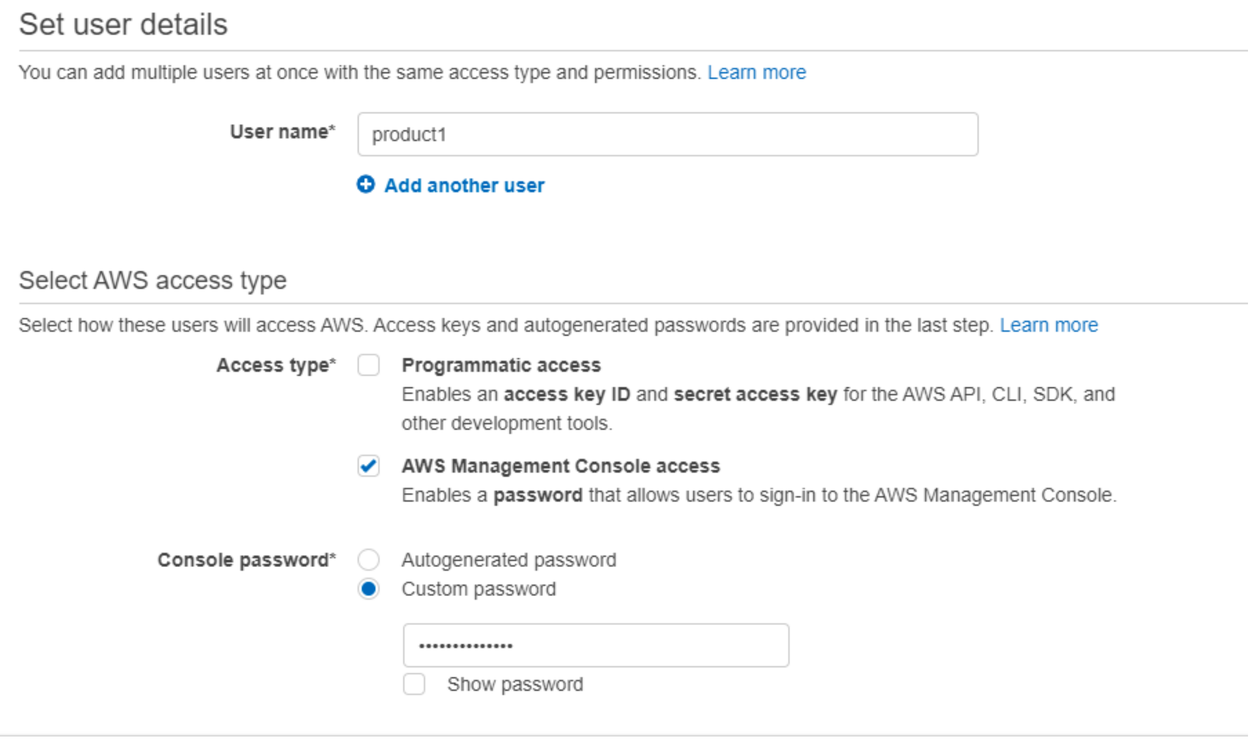
  The policy should be created as shown below.

- **Step: - 9**

  Click on the Users link and click on "Add Users".

- **Step: - 10**

  Give the user a name and select the options as shown in the below screen.



Set user details

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

User name* product1

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.

Console password* ○ Autogenerated password
● Custom password

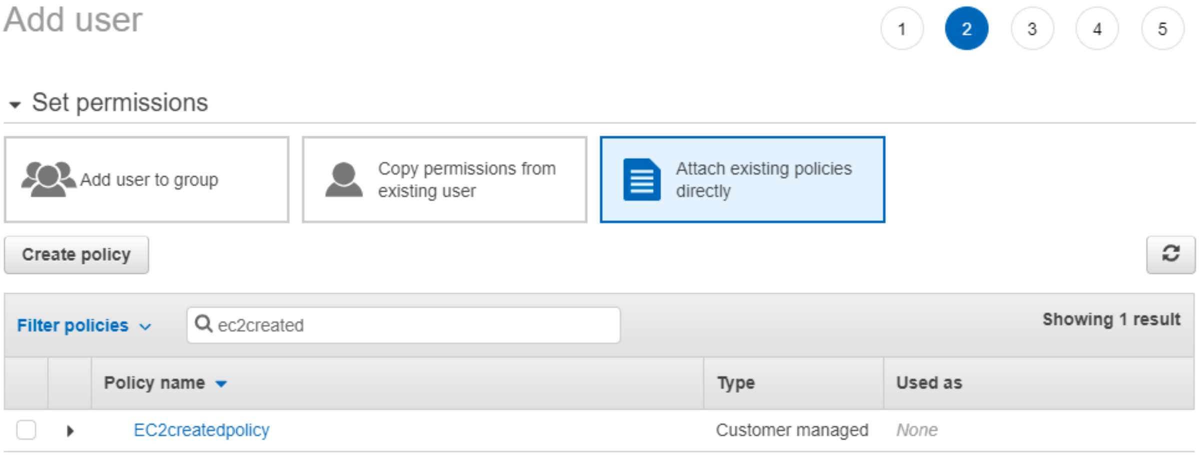
•••••••••••••••

□ Show password

USE CASE – 2

- **Step: - 11**

  Select “Attach existing policies directly” and select the policy which was created earlier. This will give the user the permissions mentioned in the policies. Click on “Next Tags”.



ec2created

		Policy name	Type	Used as
□	▸	EC2createdpolicy	Customer managed	None

- **Step: - 12**

  Tags are optional and can be avoided. Click on “Next Review”.

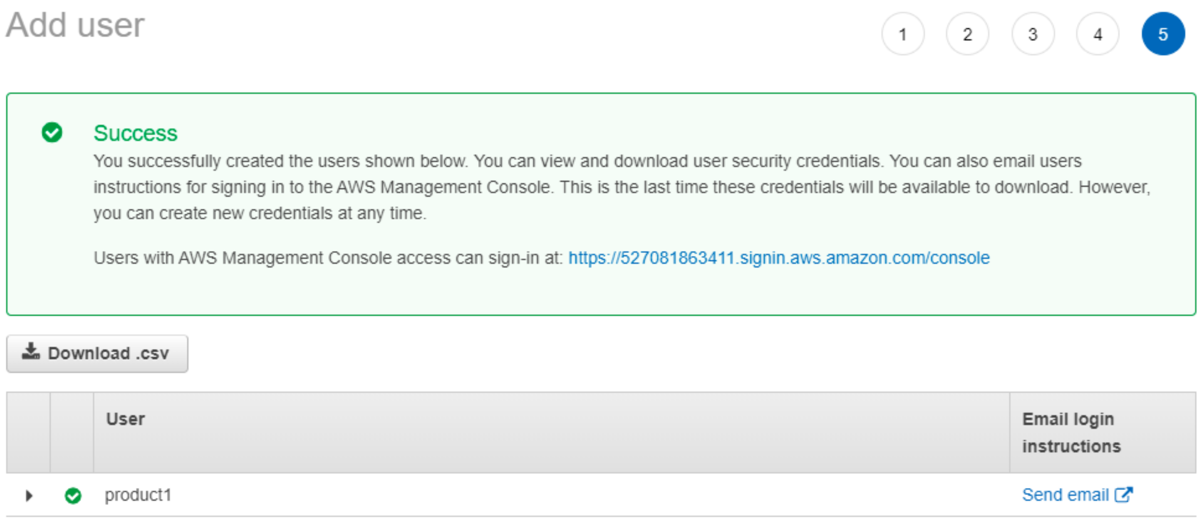
- **Step: - 13**

  Review all the details for the user to be created and click on “Create user”.

USE CASE – 2

- **Step: - 14**

  The IAM user will be created with the mentioned details. Note down the URL or the link mentioned in this screen. This is link to be used to login as an IAM user.



Add user

Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: https://527081863411.signin.aws.amazon.com/console

		User	Email login instructions
▸	✓	product1	Send email

- **Step: - 15**

  Logout as the current user and open the URL. Enter the credentials and click on "Sign in".

- **Step: - 16**

  Since the user has Describe permissions for all the EC2 instances, the user should be able to see the details of both the EC2 instances.

- **Step: -17**

  The user has read-only access to the Development EC2 instance. Select the Development EC2 instance and try to stop it. The "not authorized" message would be shown and displayed below.

- **Step: - 18**

  For the Production instances the user has access to Start/Stop. Try to Terminate the Production instance. The "not authorized" message would be shown and displayed below.

USE CASE – 2

- **Step: -19**

  Try to stop the Production instances and it would be successful as the user has permissions to Stop the EC2 instances with the tag Env as Key and Production as Value.

- **Step: - 20**

  Go to the CloudTrail and then to the “Event history” option. The CloudTrail feature is by default on and we should be able to see all the actions performed by the root use and as well as the IAM user. Note that it will usually take about 15-20 minutes for the action to be propagated to the IAM CloudTrail Service.

- **Clearing the session: -**

  The following AWS resources have to be cleaned up in the same order.

  - Terminate the EC2 instances
  - Delete the IAM User
  - Delete the IAM Policy.