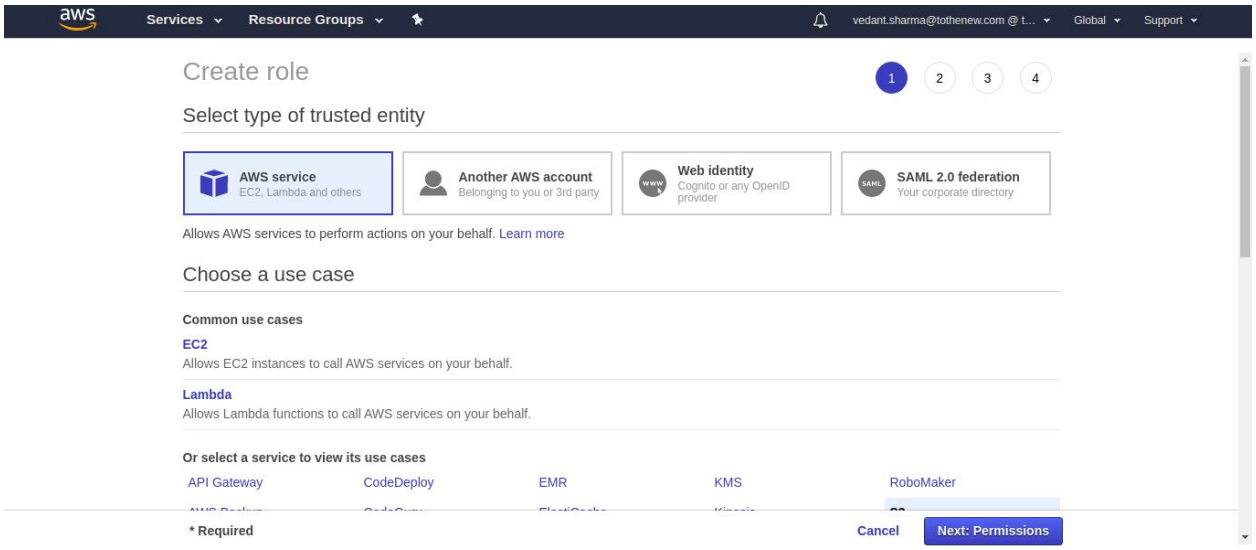
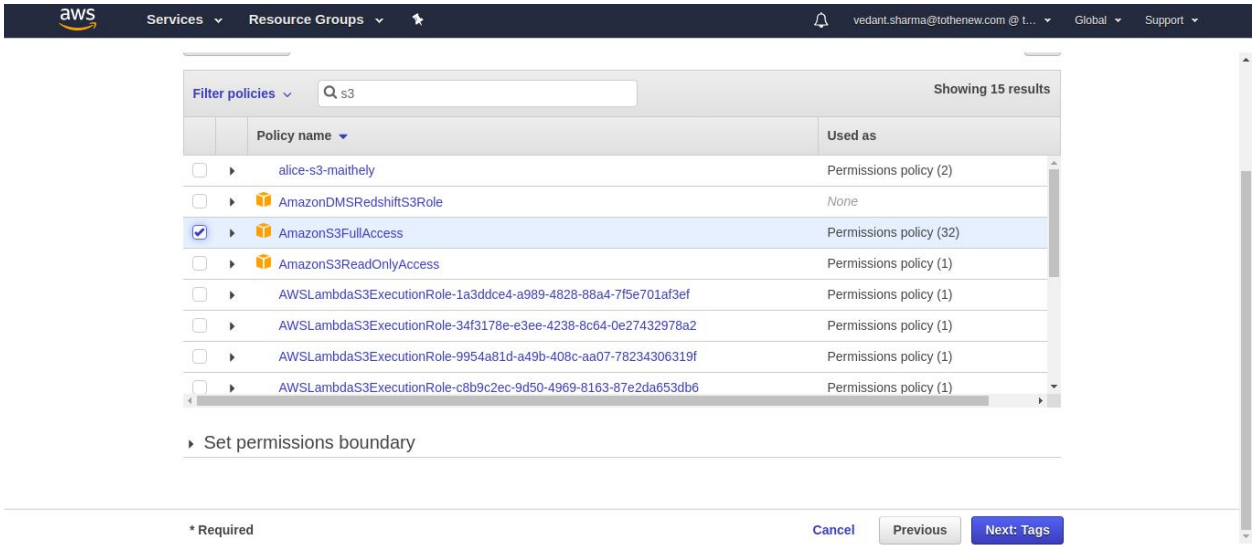
IAM:

Q1. Create a Role with full access to S3

Creating a role under the IAM service



Allowing S3 full access:



aws Services Resource Groups vedant.sharma@tothenew.com @ t... Global Support

Create role

1 2 3 4

Review

Provide the required information below and review this role before you create it.

Role name* Vedant-S3-access

Use alphanumeric and '+=,.@-_' characters. Maximum 64 characters.

Role description Allows S3 to call AWS services on your behalf.

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

Trusted entities AWS service: s3.amazonaws.com

Policies AmazonS3FullAccess

Permissions boundary Permissions boundary is not set

* Required

Cancel Previous Create role

aws Services Resource Groups vedant.sharma@tothenew.com @ t... Global Support

- Dashboard
- Access management
  - Groups
  - Users
  - Roles
  - Policies
  - Identity providers
  - Account settings
- Access reports
  - Access analyzer
    - Archive rules
    - Analyzer details
  - Credential report
  - Organization activity
  - Service control policies (SCPs)

Roles > Vedant-S3-access

Summary

Delete role

Role ARN arn:aws:iam::187632318301:role/Vedant-S3-access

Role description Allows S3 to call AWS services on your behalf. | Edit

Instance Profile ARNs

Path /

Creation time 2020-02-28 11:22 UTC+0530

Last activity Not accessed in the tracking period

Maximum CLI/API session duration 1 hour Edit

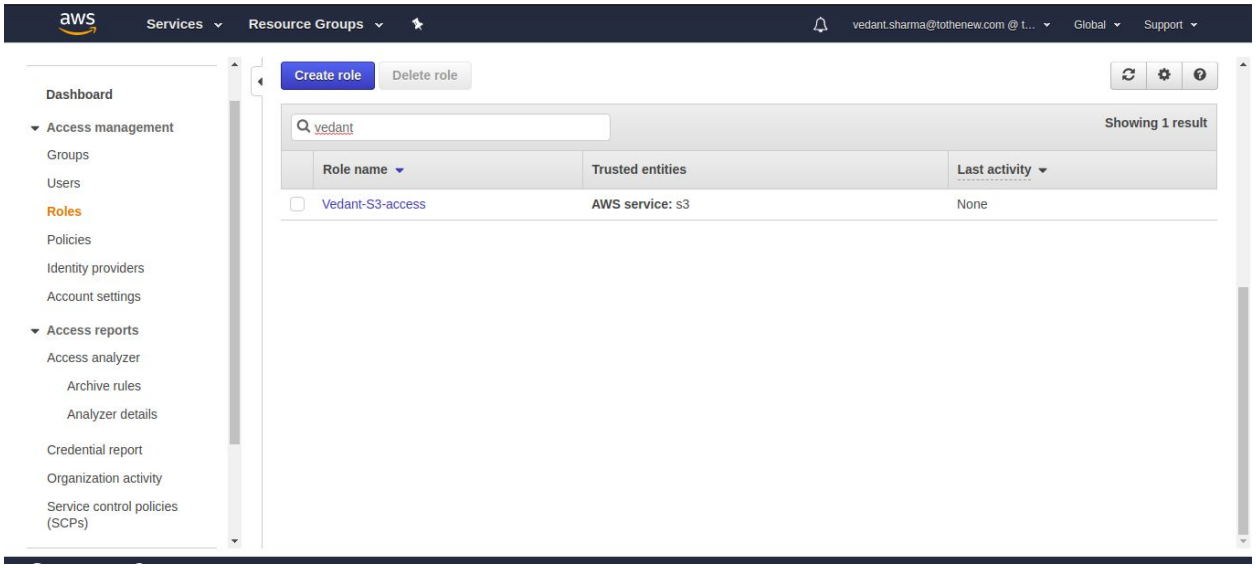
Permissions | Trust relationships | Tags (1) | Access Advisor | Revoke sessions

Permissions policies (1 policy applied)

Attach policies

Add inline policy

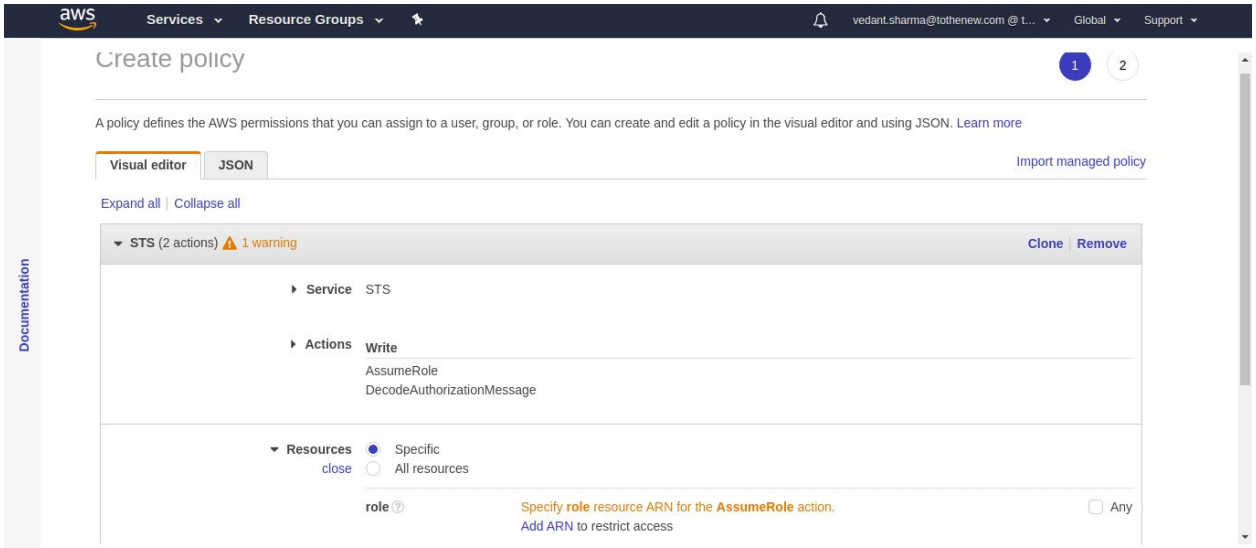
Policy name	Policy type
AmazonS3FullAccess	AWS managed policy



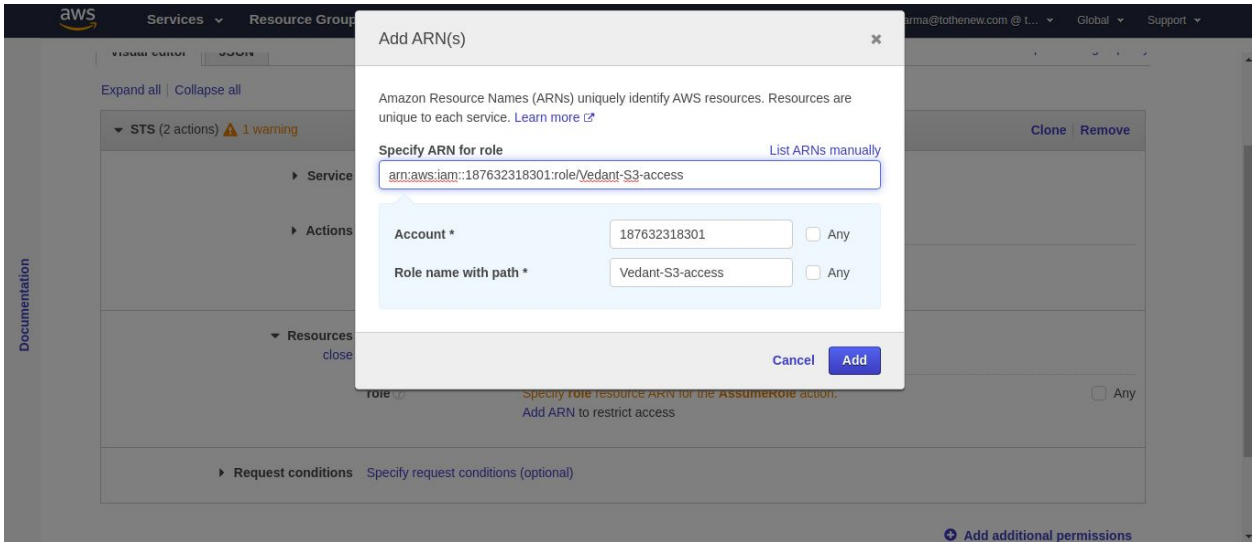
Q2. Create another which has the policy to assume the previous Role

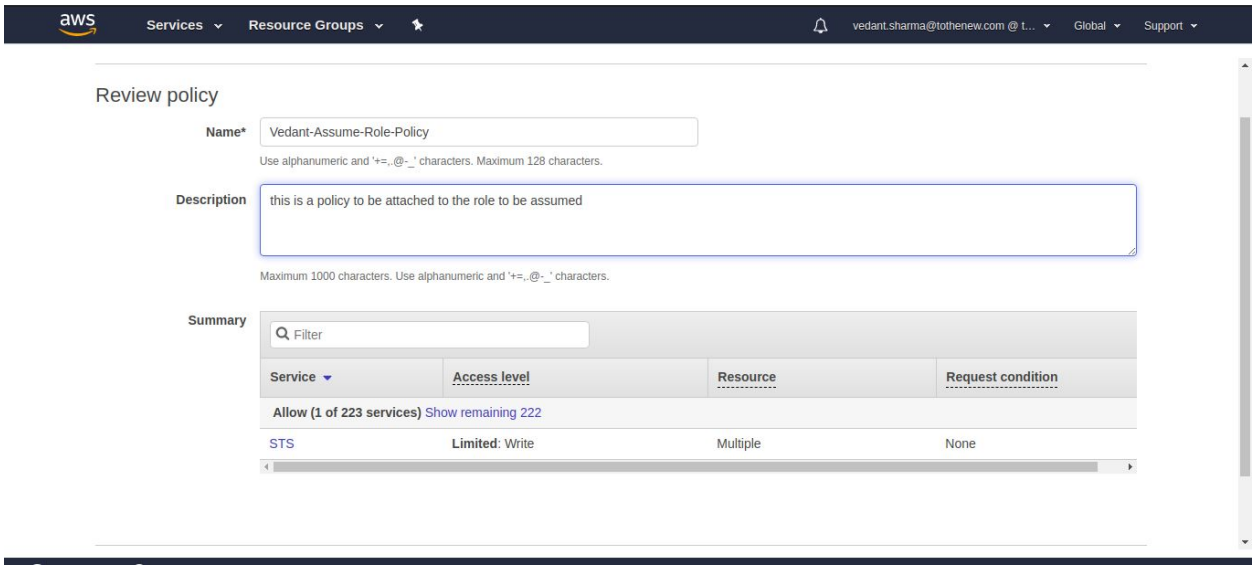
After creating a new role we create a new policy and then we select a assume role action

Selection of STS service:

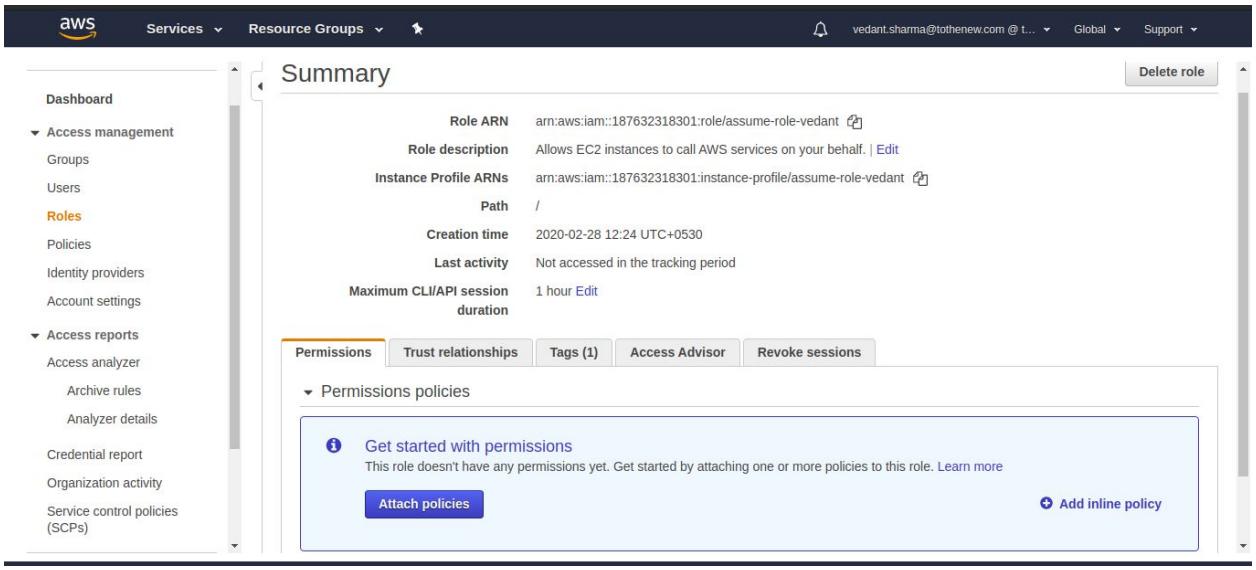


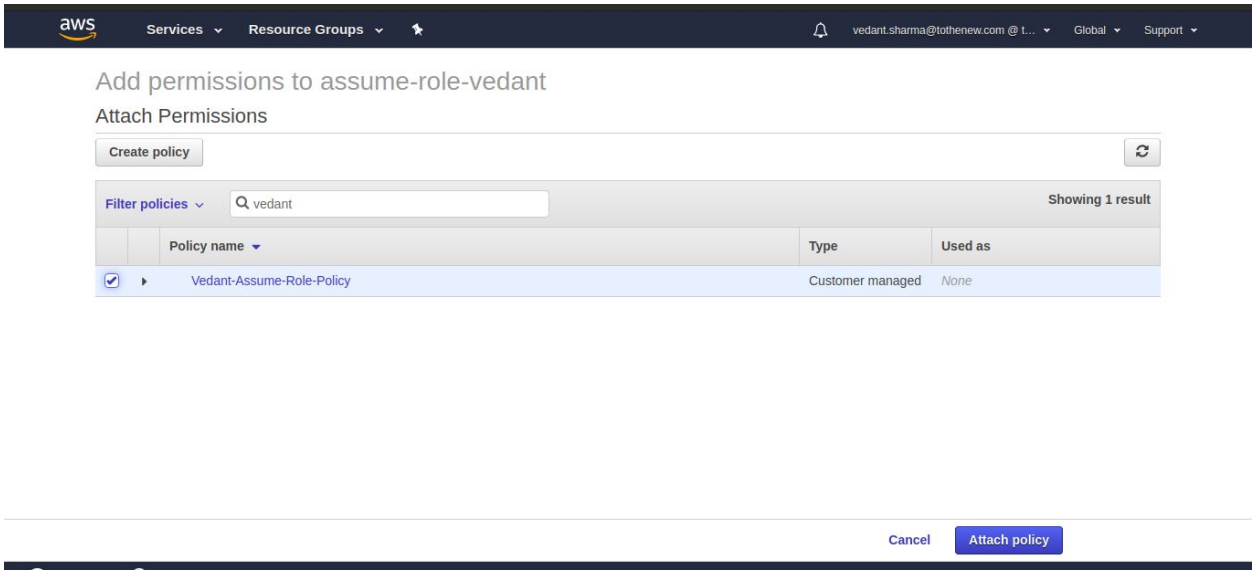
Adding ARN of the previously created account



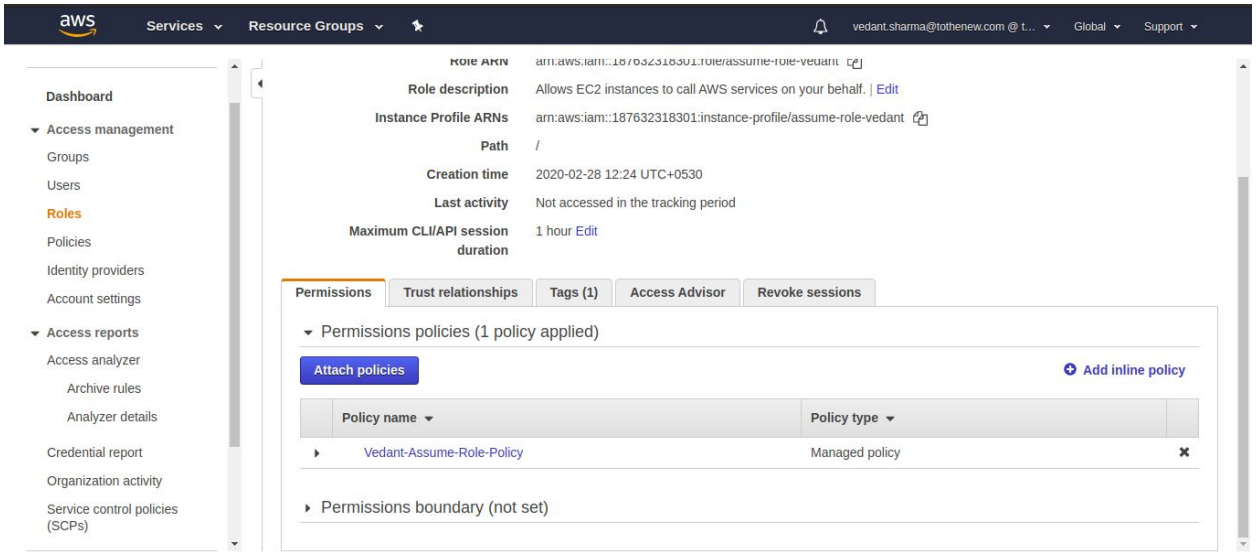


Attaching policy to newly created role

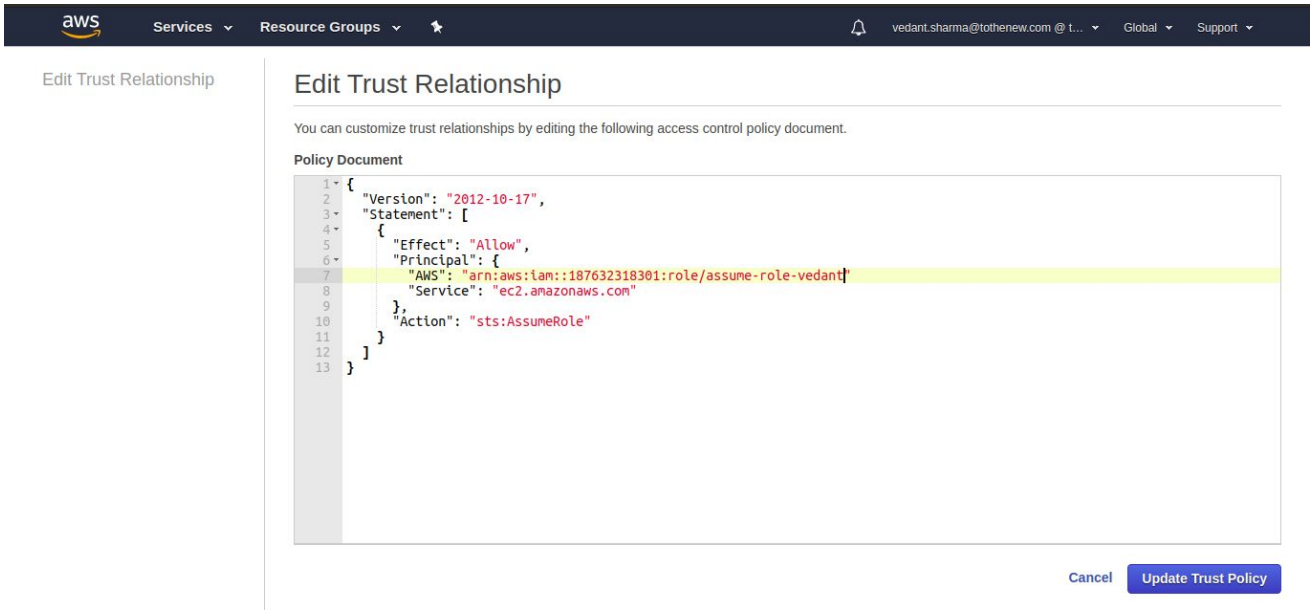




Assume policy attached

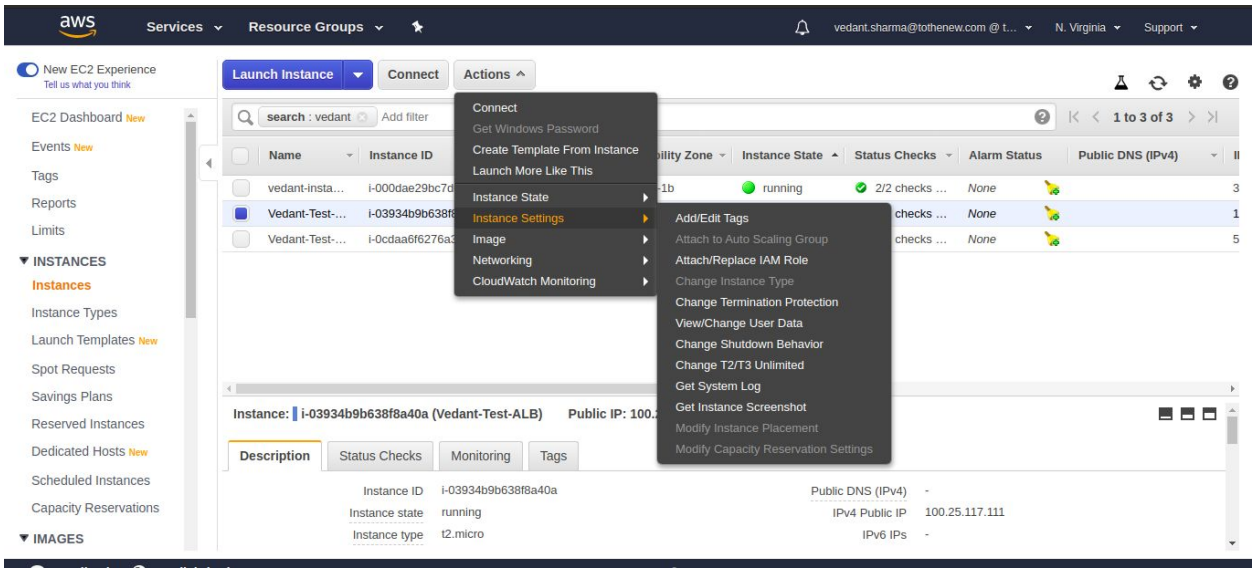


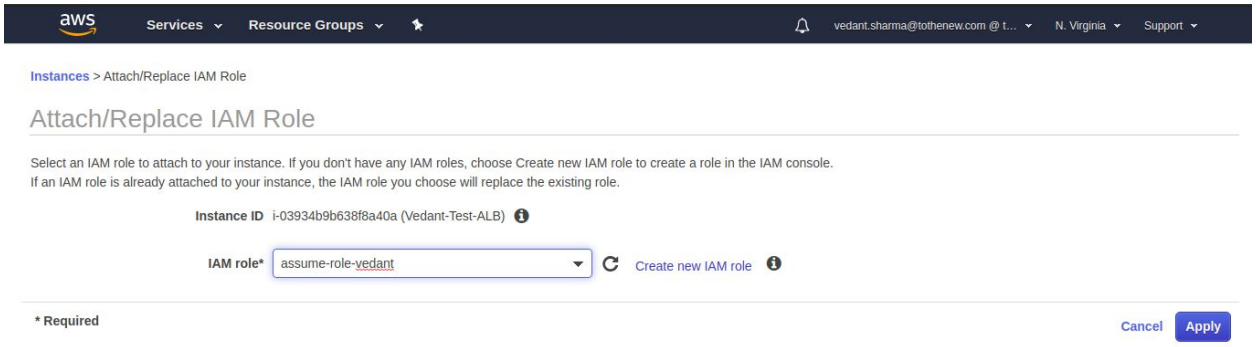
Adding ARN of assumed role to the trusted relationship of the policy for full access:



Q3. Attach this to an instance and get an sts token.

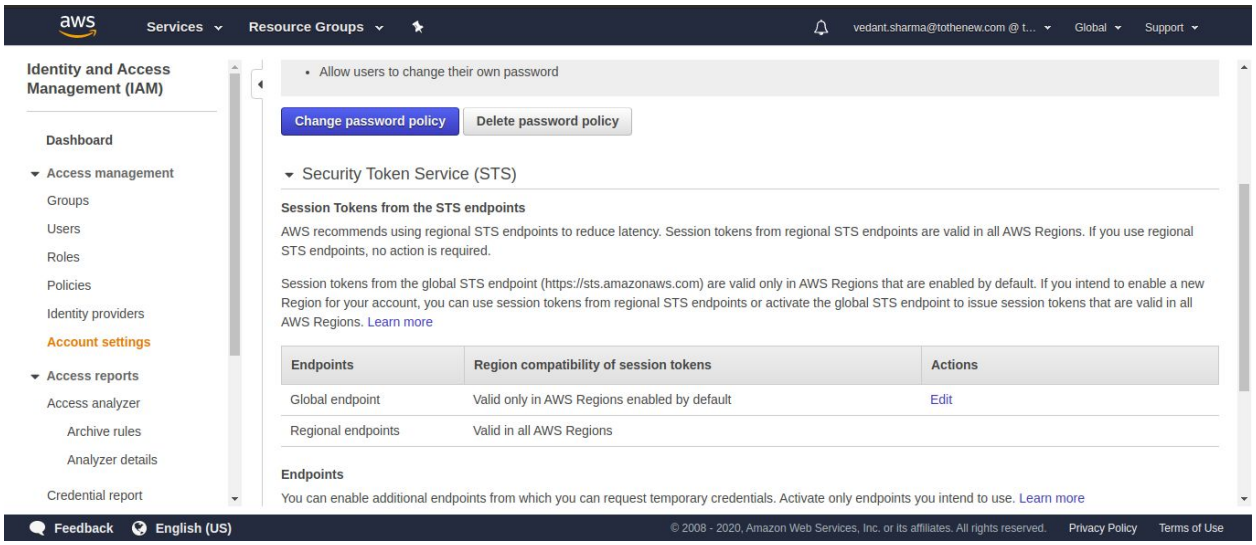
Attaching an IAM role to the instance:


aws
Services
Resource Groups
vedant.sharma@tothenew.com @ t...
N. Virginia
Support
New EC2 Experience
Tell us what you think
EC2 Dashboard New
Events New
Tags
Reports
Limits
INSTANCES
Instances
Instance Types
Launch Templates New
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts New
Scheduled Instances
Capacity Reservations
IMAGES
Launch Instance
Connect
Actions
search : vedant
Add filter
1 to 3 of 3
Name
Instance ID
Instance State
Status Checks
Alarm Status
Public DNS (IPv4)
vedant-insta...
running
2/2 checks ...
None
Vedant-Test-...
Vedant-Test-...
Connect
Get Windows Password
Create Template From Instance
Launch More Like This
Instance State
Instance Settings
Image
Networking
CloudWatch Monitoring
Add/Edit Tags
Attach to Auto Scaling Group
Attach/Replace IAM Role
Change Instance Type
Change Termination Protection
View/Change User Data
Change Shutdown Behavior
Change T2/T3 Unlimited
Get System Log
Get Instance Screenshot
Modify Instance Placement
Modify Capacity Reservation Settings
Instance: i-03934b9b638f8a40a (Vedant-Test-ALB)
Public IP: 100.
Description
Status Checks
Monitoring
Tags
Instance ID
i-03934b9b638f8a40a
Instance state
running
Instance type
t2.micro
Public DNS (IPv4)
IPv4 Public IP
100.25.117.111
IPv6 IPs

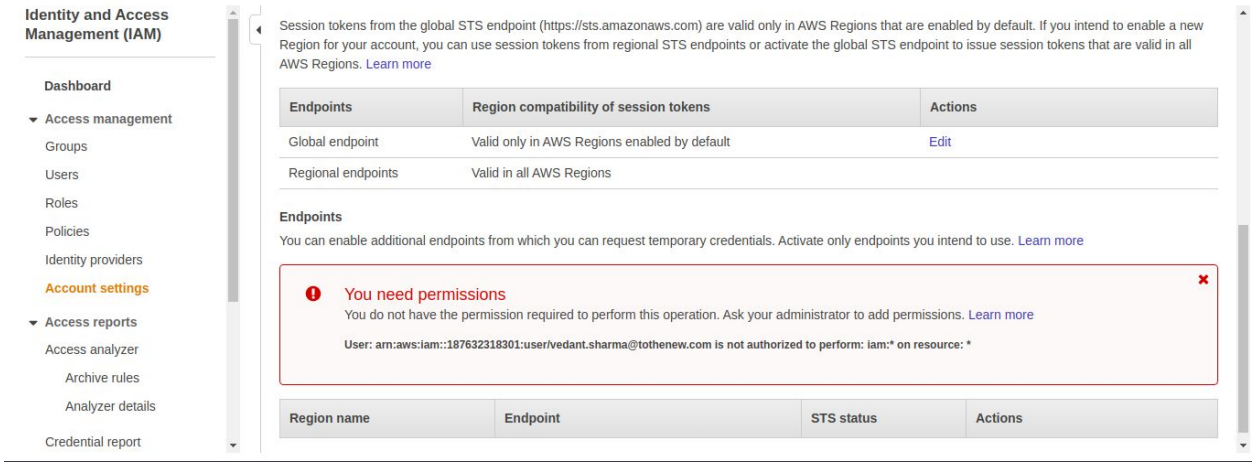


Accessing aws s3 bucket using instance and aws cli:

Getting STS details after going to the IAM service console and then selecting account settings:



No permission to perform STS token fetching:



Q4. Create a group for "Data Administrator" where the user 'Alice' be a member of this group. This group will prepare the data for the analysis. So Provide the following access to the group.
Service: Amazon S3;

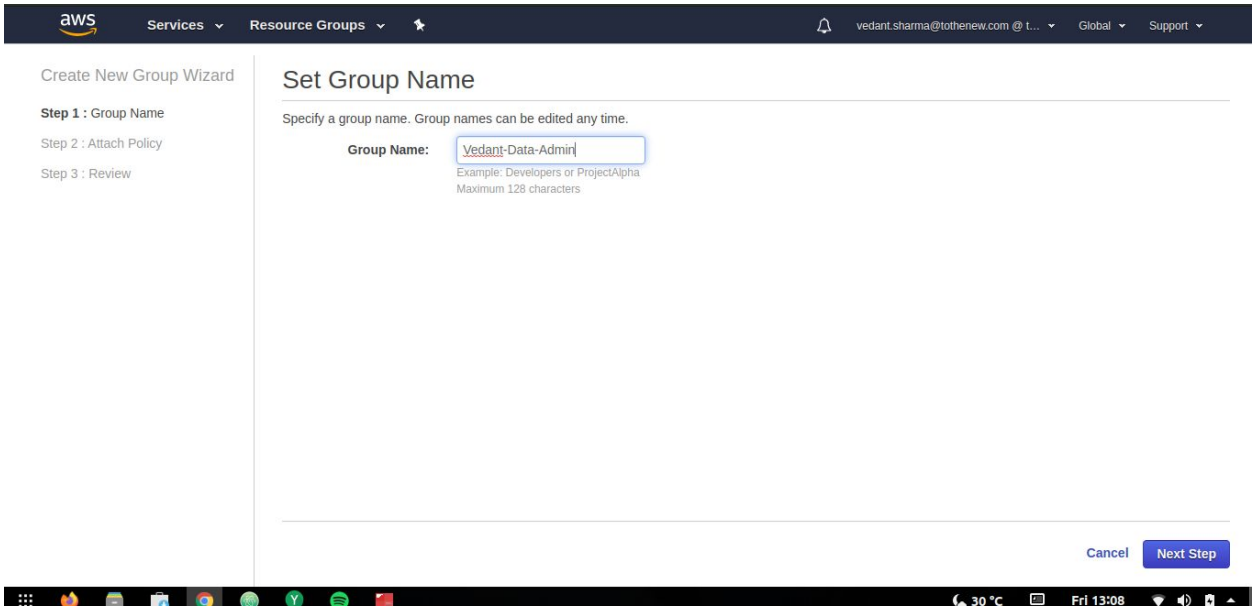
Action:

Get*,

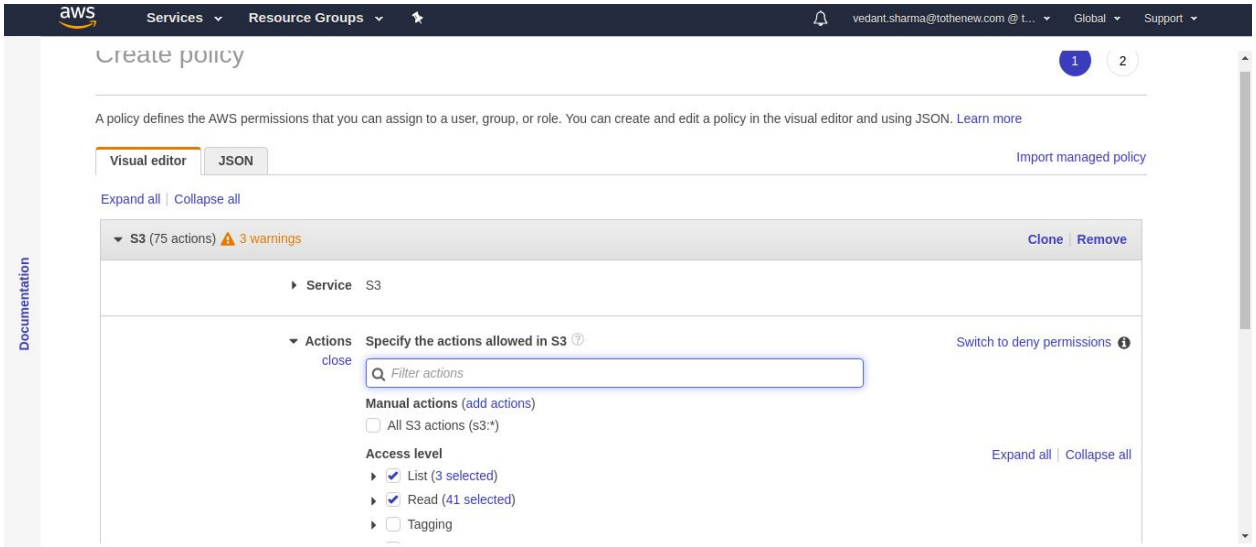
List*,

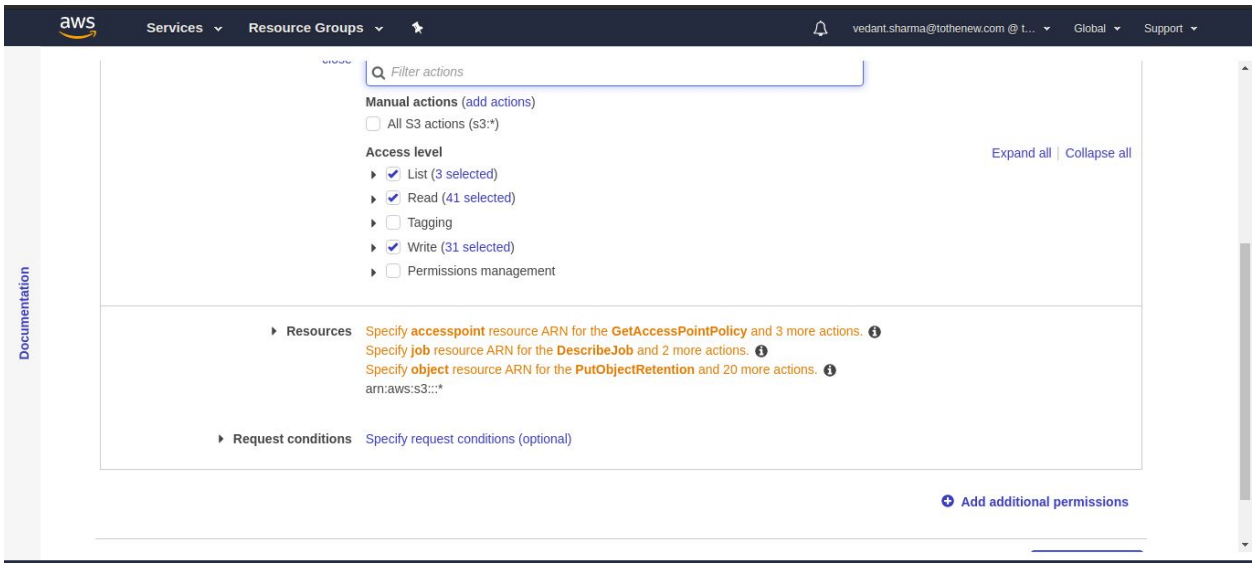
Put*,

ARN: Input and output Buckets (no conditions)

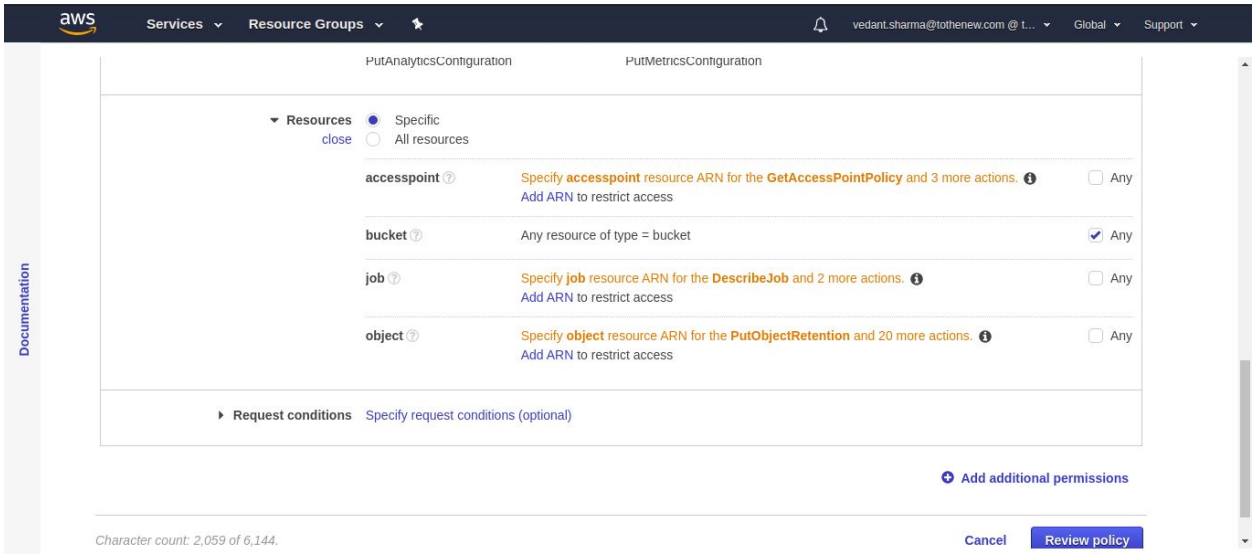


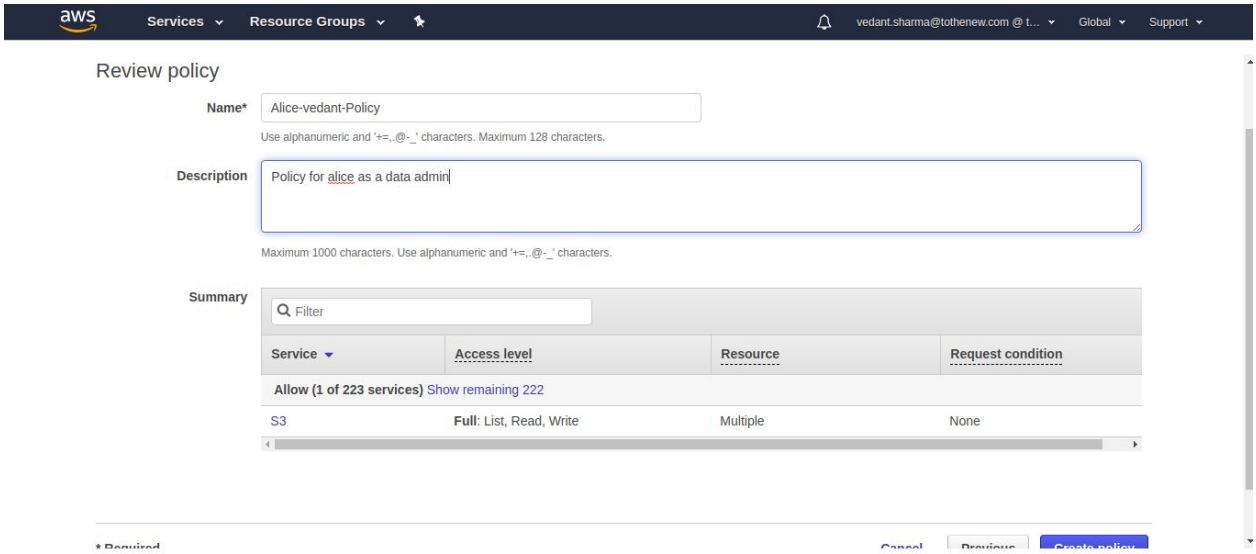
Creating a Policy for mentioned actions:



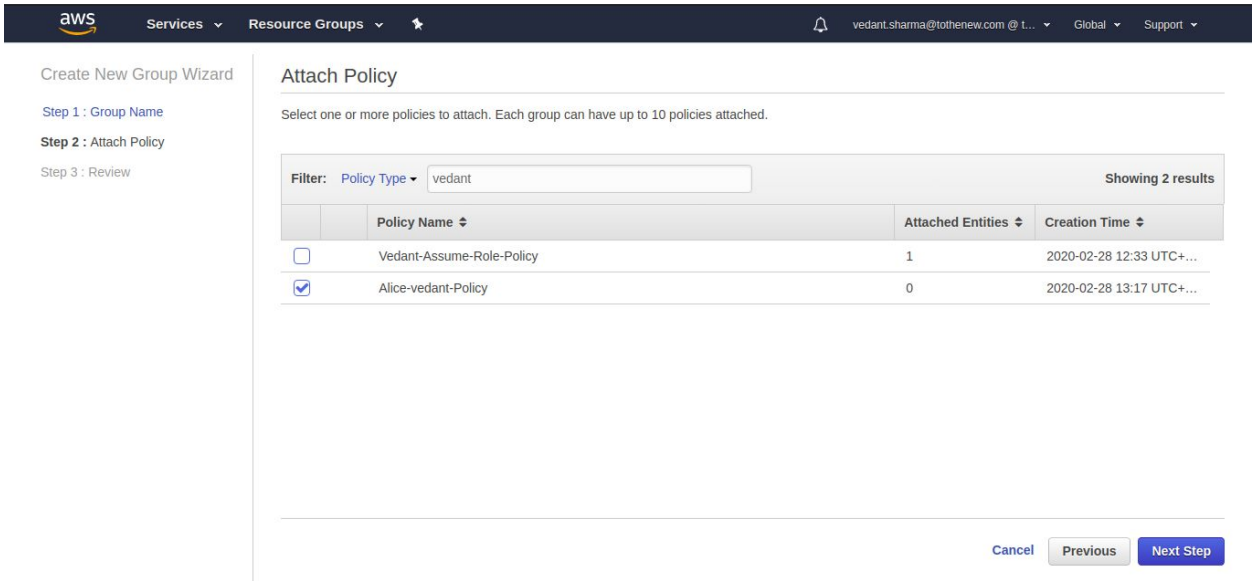


Selecting ARN as a resource

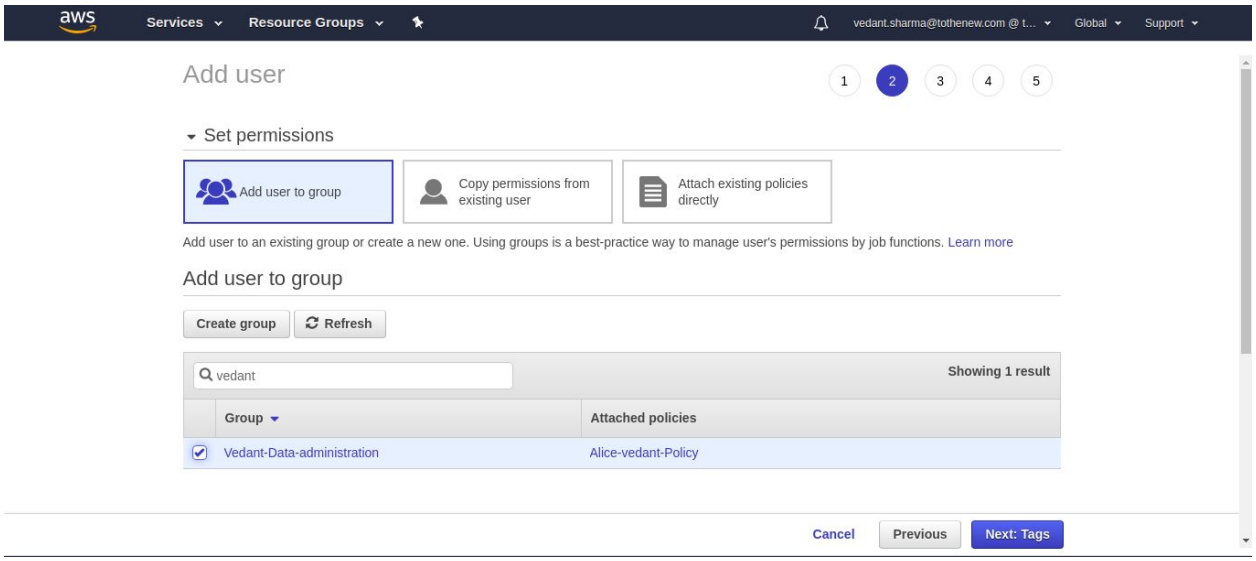


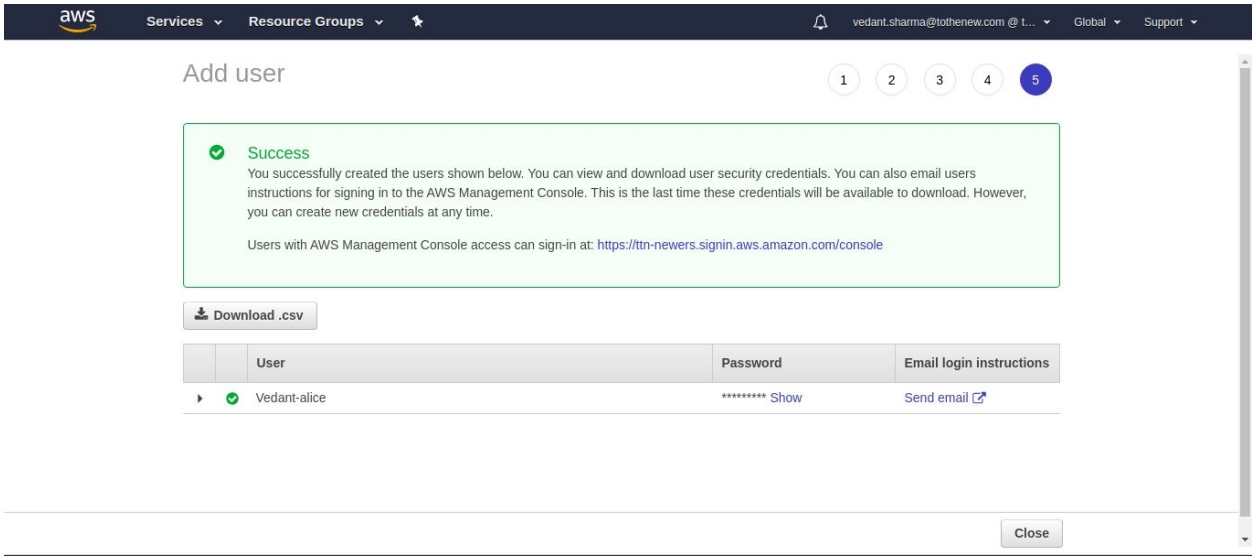


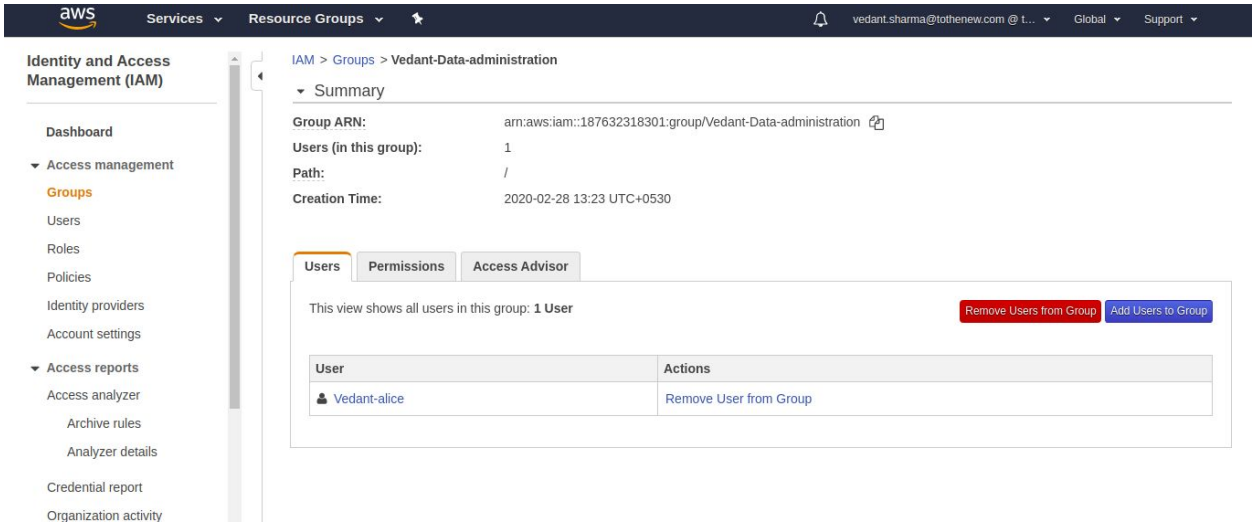
Now creating a group and attaching the previously created policy to the group



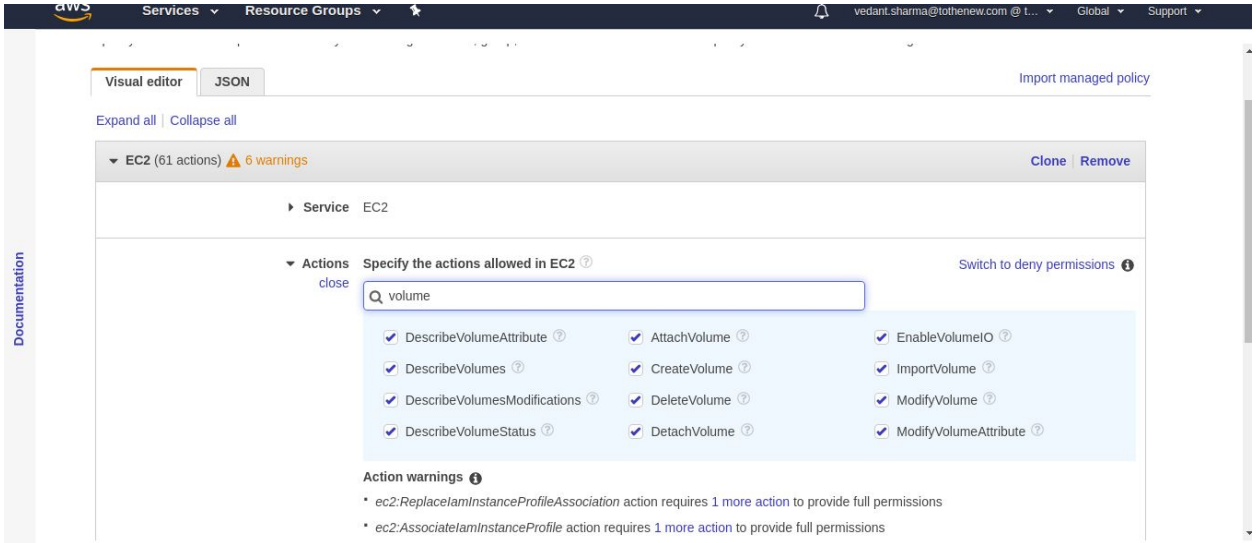
Adding alice as a user to the group created previously:







Q5. Create a group for the "Developer group " where the user 'bob ' is a member of this group. This group with Test Newly Developed Features for which they require access to EC2

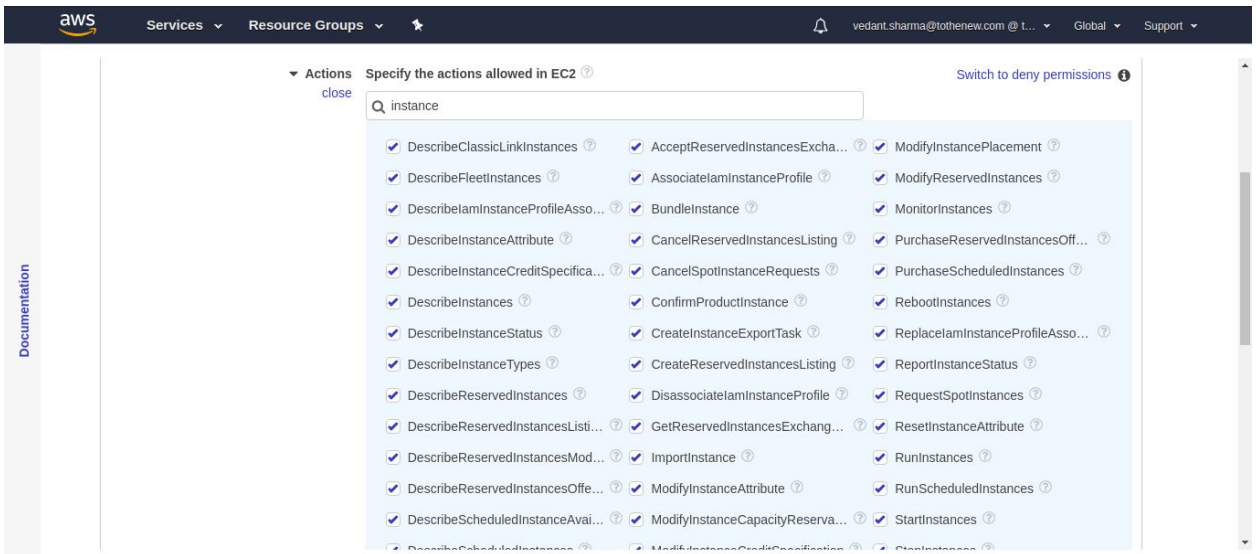


instances. Provide the following access to this group:
Service: Amazon EC2

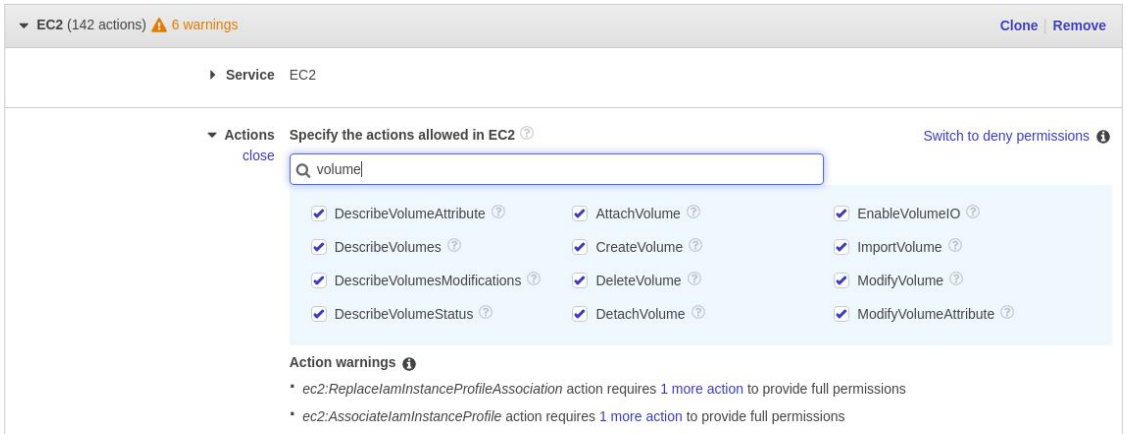
Action: *Instances, *Volume, Describe*, CreateTags;

Condition: Dev Subnets only

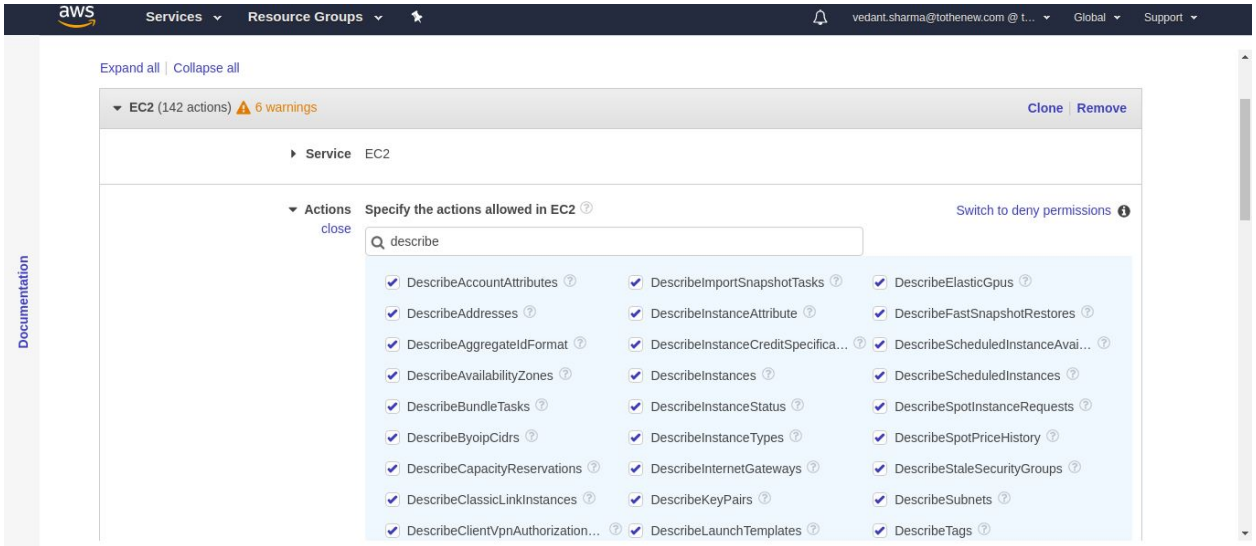
Creating a new policy with instance actions



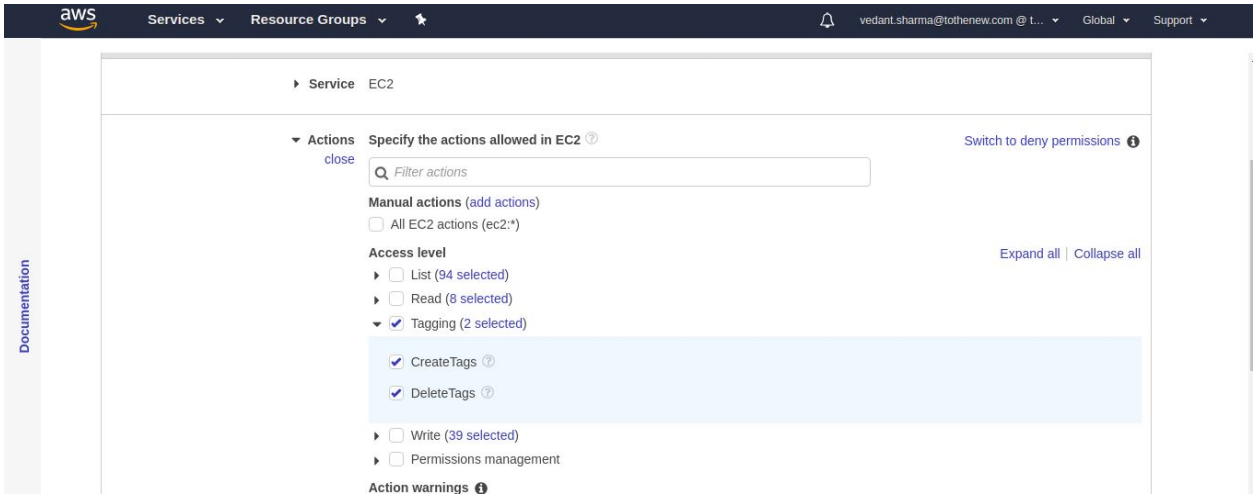
Volume*


EC2 (142 actions) 6 warnings
Clone | Remove
Service EC2
Actions
close
Specify the actions allowed in EC2
Switch to deny permissions
volume
DescribeVolumeAttribute
AttachVolume
EnableVolumeIO
DescribeVolumes
CreateVolume
ImportVolume
DescribeVolumesModifications
DeleteVolume
ModifyVolume
DescribeVolumeStatus
DetachVolume
ModifyVolumeAttribute
Action warnings
ec2:ReplaceIamInstanceProfileAssociation action requires 1 more action to provide full permissions
ec2:AssociateIamInstanceProfile action requires 1 more action to provide full permissions

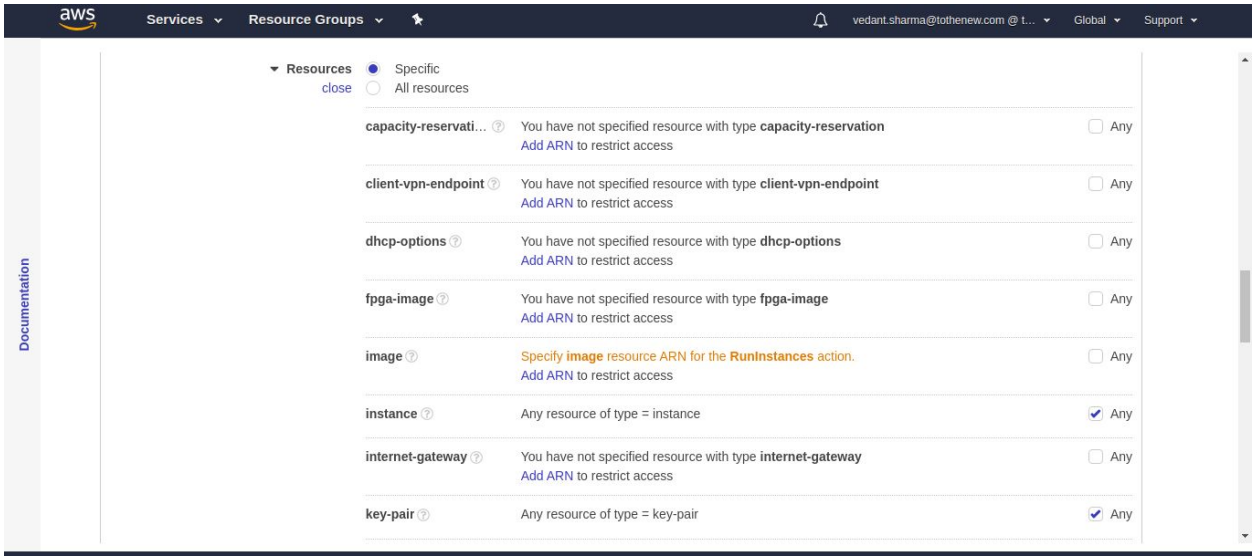
Describe*

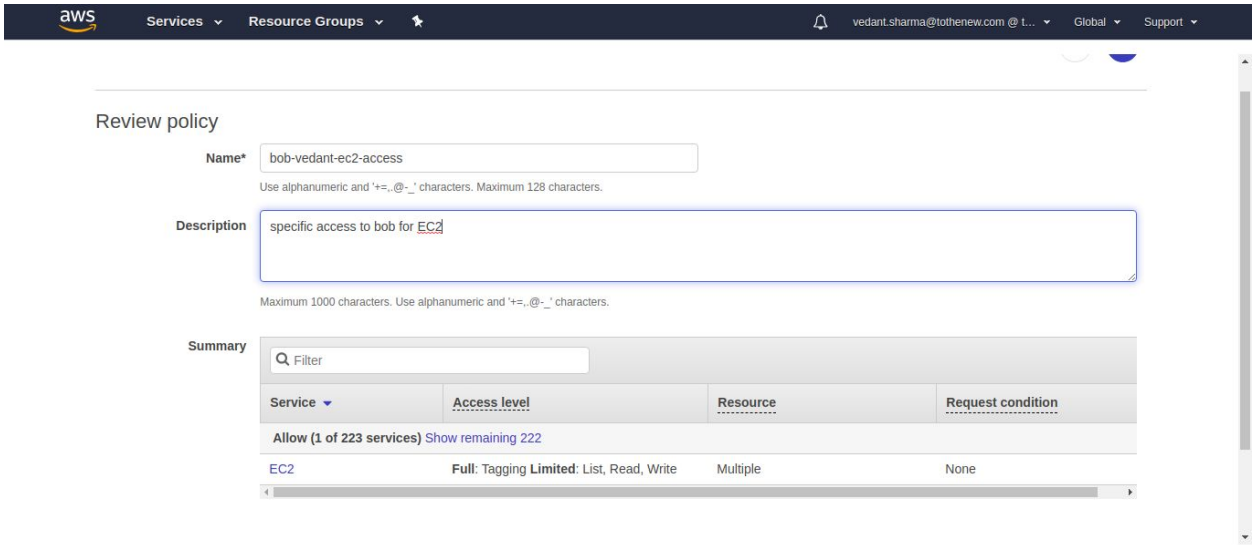

aws
Services
Resource Groups
vedant.sharma@tothenew.com @ t...
Global
Support
Documentation
Expand all | Collapse all
EC2 (142 actions) 6 warnings
Clone | Remove
Service EC2
Actions
close
Specify the actions allowed in EC2
Switch to deny permissions
describe
DescribeAccountAttributes
DescribeImportSnapshotTasks
DescribeElasticGpus
DescribeAddresses
DescribeInstanceAttribute
DescribeFastSnapshotRestores
DescribeAggregateIdFormat
DescribeInstanceCreditSpecifica...
DescribeScheduledInstanceAvai...
DescribeAvailabilityZones
DescribeInstances
DescribeScheduledInstances
DescribeBundleTasks
DescribeInstanceStatus
DescribeSpotInstanceRequests
DescribeByoipCidrs
DescribeInstanceTypes
DescribeSpotPriceHistory
DescribeCapacityReservations
DescribeInternetGateways
DescribeStaleSecurityGroups
DescribeClassicLinkInstances
DescribeKeyPairs
DescribeSubnets
DescribeClientVpnAuthorization...
DescribeLaunchTemplates
DescribeTags

Tags (read and write)

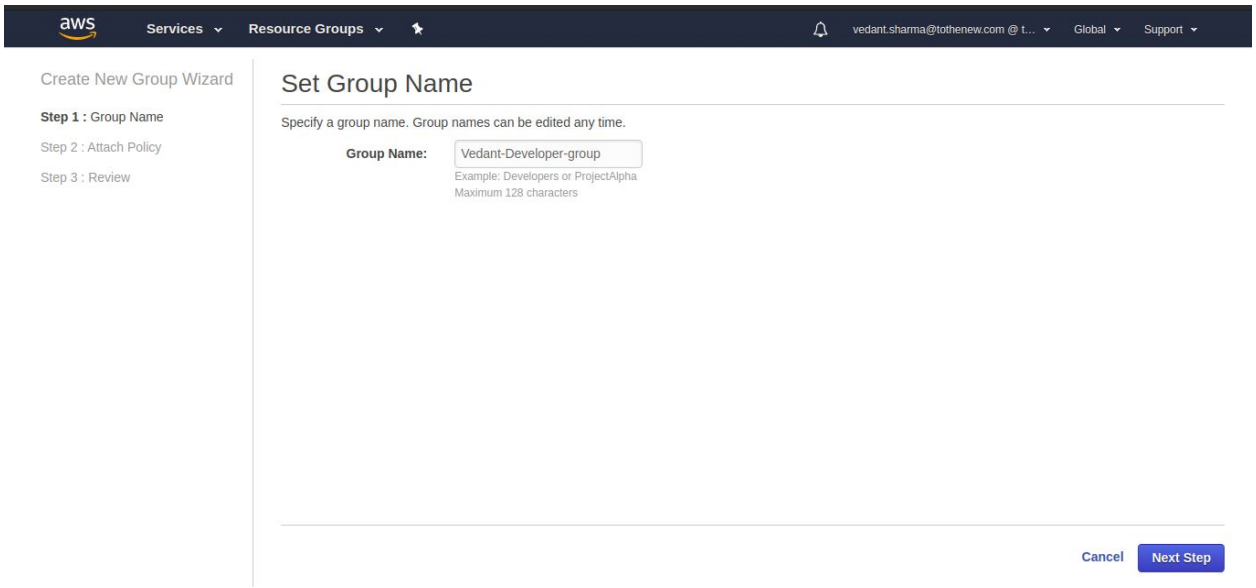


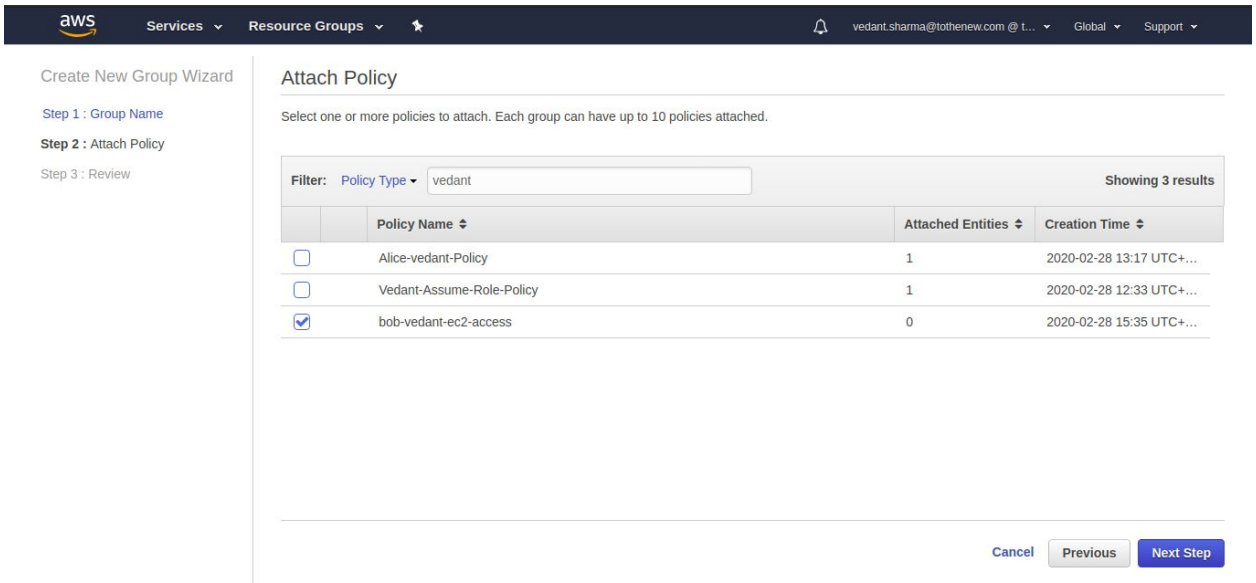
Specifying resources:



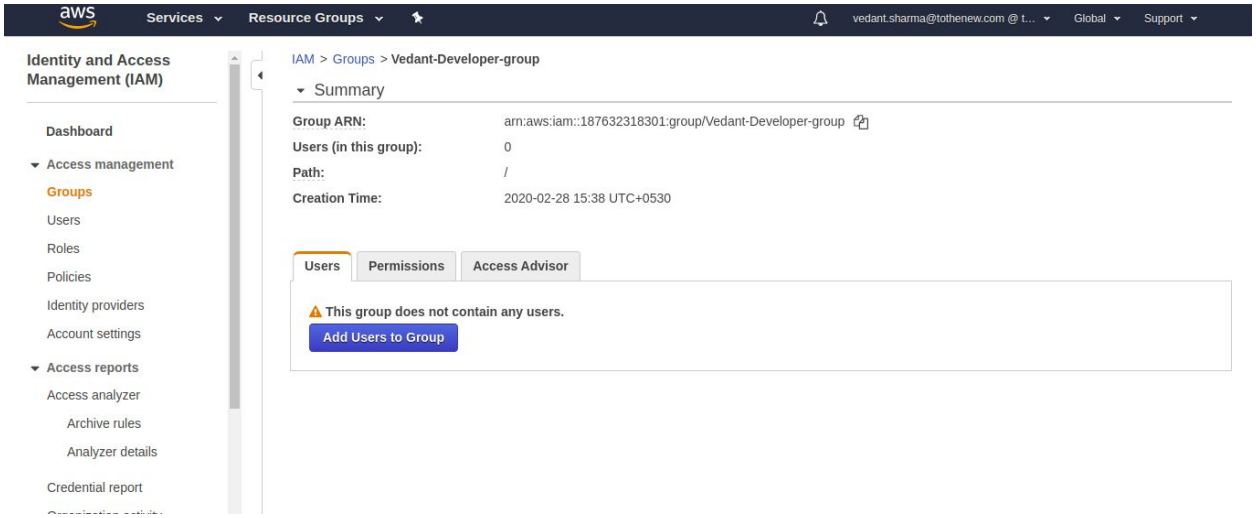


Creating a developer group and attached a previous policy to the group:

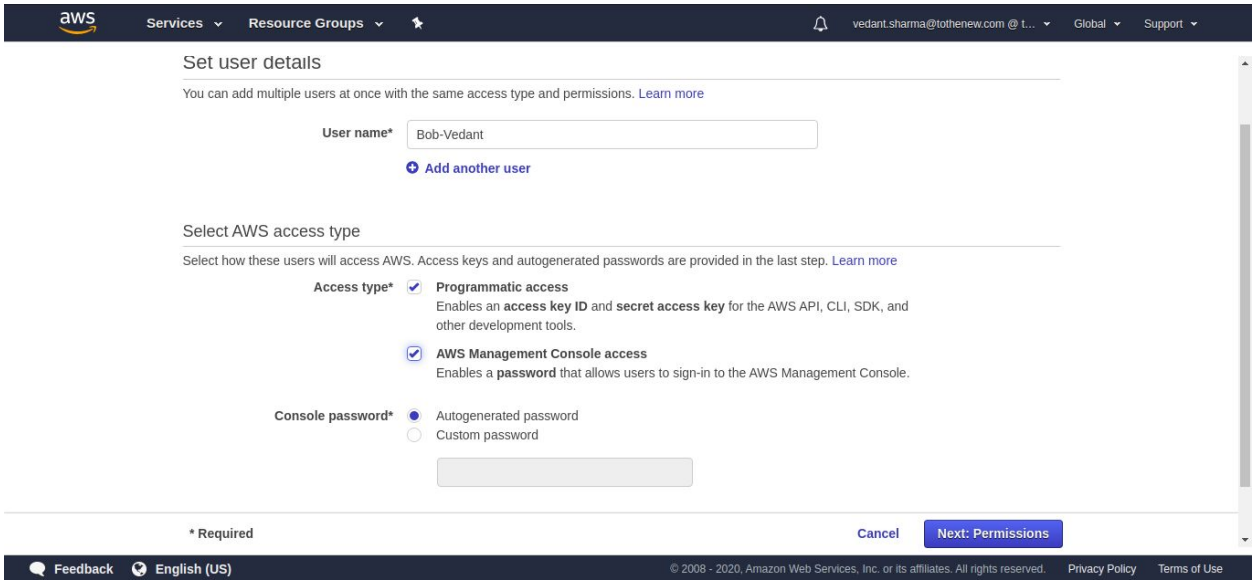




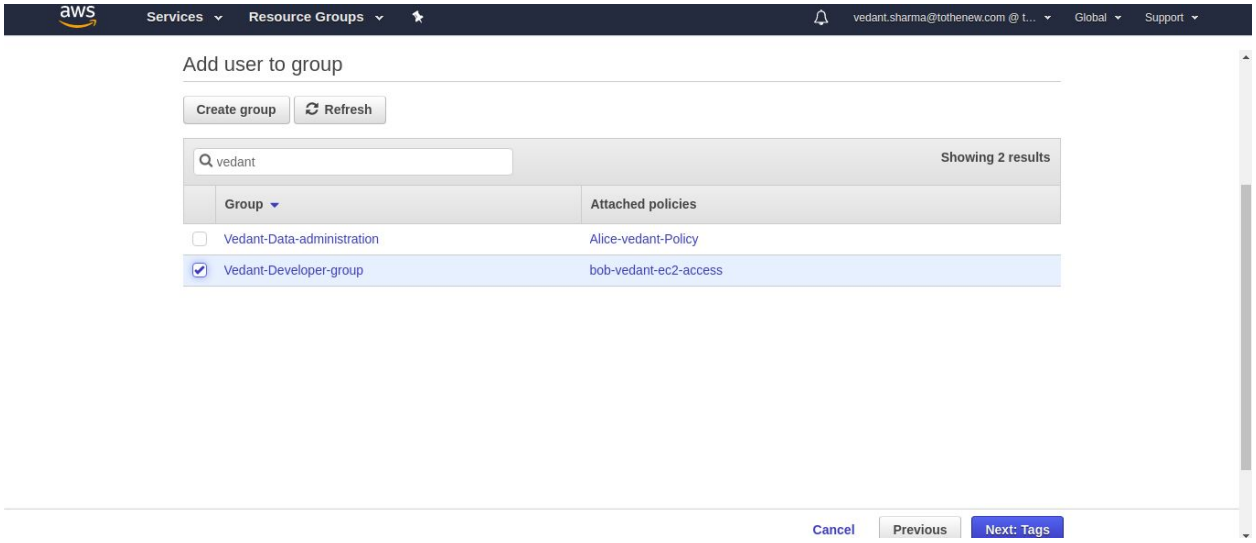
Adding bob to the created group:

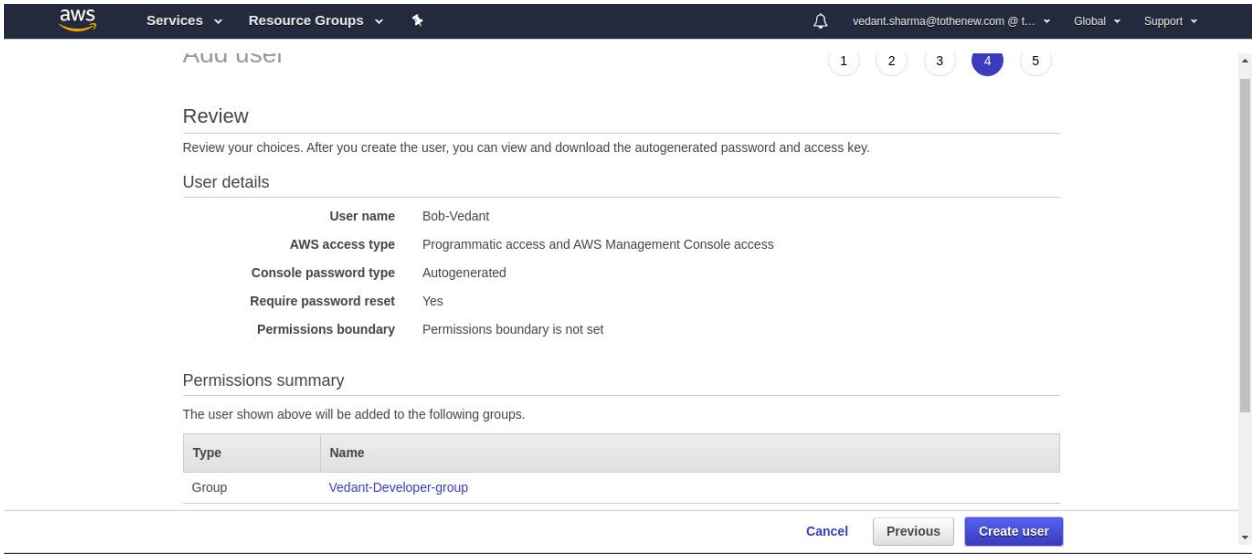


Creating bob user

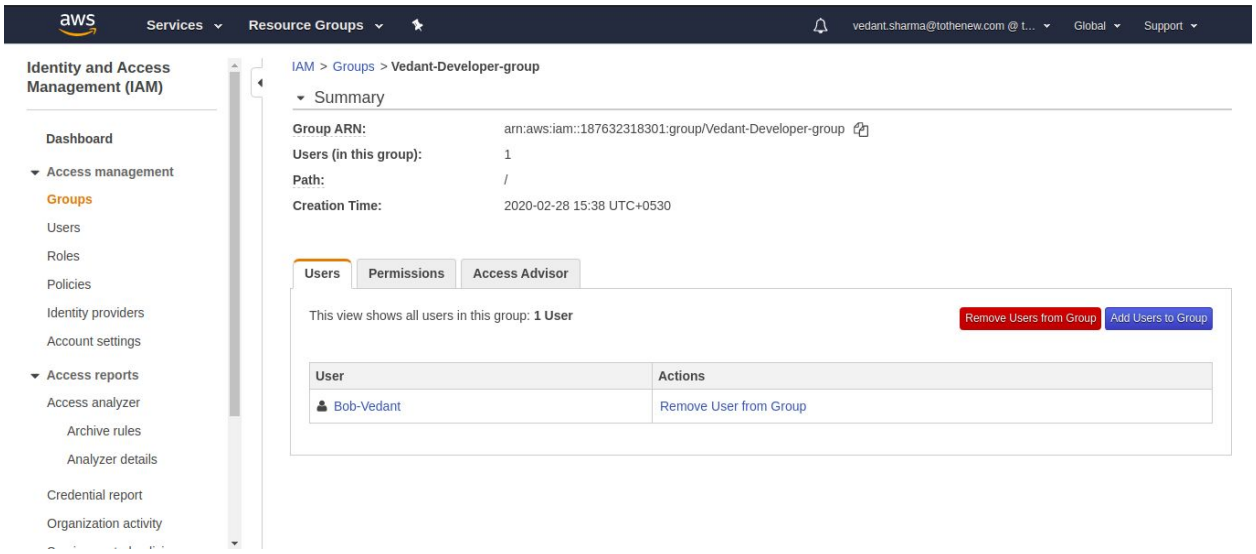


Adding bob to the previous group:





Added to the group with a specified policy:



Q6. Identify the unused IAM users/credentials using AWS CLI.

To identify the unused IAM user/credentials we use “aws iam list-users” this displays list of users with their details. Each with a PasswordLastUsed value. If the value is missing, then the user either has no password or the password has not been used since IAM began tracking password

```
ubuntu@ip-10-0-3-131:~$ aws iam get-user
{
    "User": {
        "UserName": "vedant.sharma@tothenew.com",
        "PasswordLastUsed": "2020-02-28T05:35:35Z",
        "CreateDate": "2020-02-19T11:04:05Z",
        "UserId": "AIDASXL6B65OY2RMI3I4O",
        "Path": "/",
        "Arn": "arn:aws:iam::187632318301:user/vedant.sharma@tothenew.com"
    }
}
ubuntu@ip-10-0-3-131:~$ aws iam get-user --user-name Bob-Vedant
{
    "User": {
        "UserName": "Bob-Vedant",
        "Tags": [
            {
                "Value": "vedant-bob-user",
                "Key": "Name"
            }
        ],
        "CreateDate": "2020-02-28T10:14:37Z",
        "UserId": "AIDASXL6B65O3OWW6OWHX",
        "Path": "/",
        "Arn": "arn:aws:iam::187632318301:user/Bob-Vedant"
    }
}
ubuntu@ip-10-0-3-131:~$ aws iam list-users | less
```

```
            "Arn": "arn:aws:iam::187632318301:user/aditya.upadhyay@tothene
w.com"
        },
        {
            "UserName": "akshay.shrivastava@tothenew.com",
            "PasswordLastUsed": "2020-02-28T04:20:30Z",
            "CreateDate": "2020-02-19T11:03:26Z",
            "UserId": "AIDASXL6B65OSGPOGZHFO",
            "Path": "/",
            "Arn": "arn:aws:iam::187632318301:user/akshay.shrivastava@toth
enew.com"
        },
        {
            "UserName": "Alice",
            "Path": "/",
            "CreateDate": "2020-02-27T12:11:40Z",
            "UserId": "AIDASXL6B65O6DXIQS5RS",
            "Arn": "arn:aws:iam::187632318301:user/Alice"
        },
        {
            "UserName": "Alice-Chhavi",
            "Path": "/",
            "CreateDate": "2020-02-27T10:46:11Z",
            "UserId": "AIDASXL6B65OVCZQE5BOM",
            "Arn": "arn:aws:iam::187632318301:user/Alice-Chhavi"
        },
        {
            "UserName": "alice-maithely",
            "Path": "/",
            "CreateDate": "2020-02-27T10:45:57Z",
            "UserId": "AIDASXL6B65O42J5DDSPA",
            "Arn": "arn:aws:iam::187632318301:user/alice-maithely"
```


The details missing PasswordLastUsed are the unused user/credentials

```
vedant@vedant:~$ aws iam list-users | jq '.Users[ ] | select(.PasswordLast
Used==null) | .UserName'
"Alice"
"Alice-baban"
"Alice-Chhavi"
"alice-maithely"
"alice-sampurna"
"Alice1"
"alice_aman"
"asusumeuser"
"Bob"
"Bob-Chirag"
"Bob-maithely"
"Bob-Srima"
"Bob-Vedant"
"bob1"
"bobpooja"
"bob_developer_baban"
"bob_sampurna"
"chhavidev"
"chhaviprod"
"Chhavi_DG"
"Chirag-Alice"
"CloudCheckr"
"Dev-diksha"
"Dev-vaibhav"
"Dev1-Arun"
"Dev1-Gargi"
"developer_baban"
```

Q7. Identify all the instances having the tag key-value "backup=true" using AWS CLI.

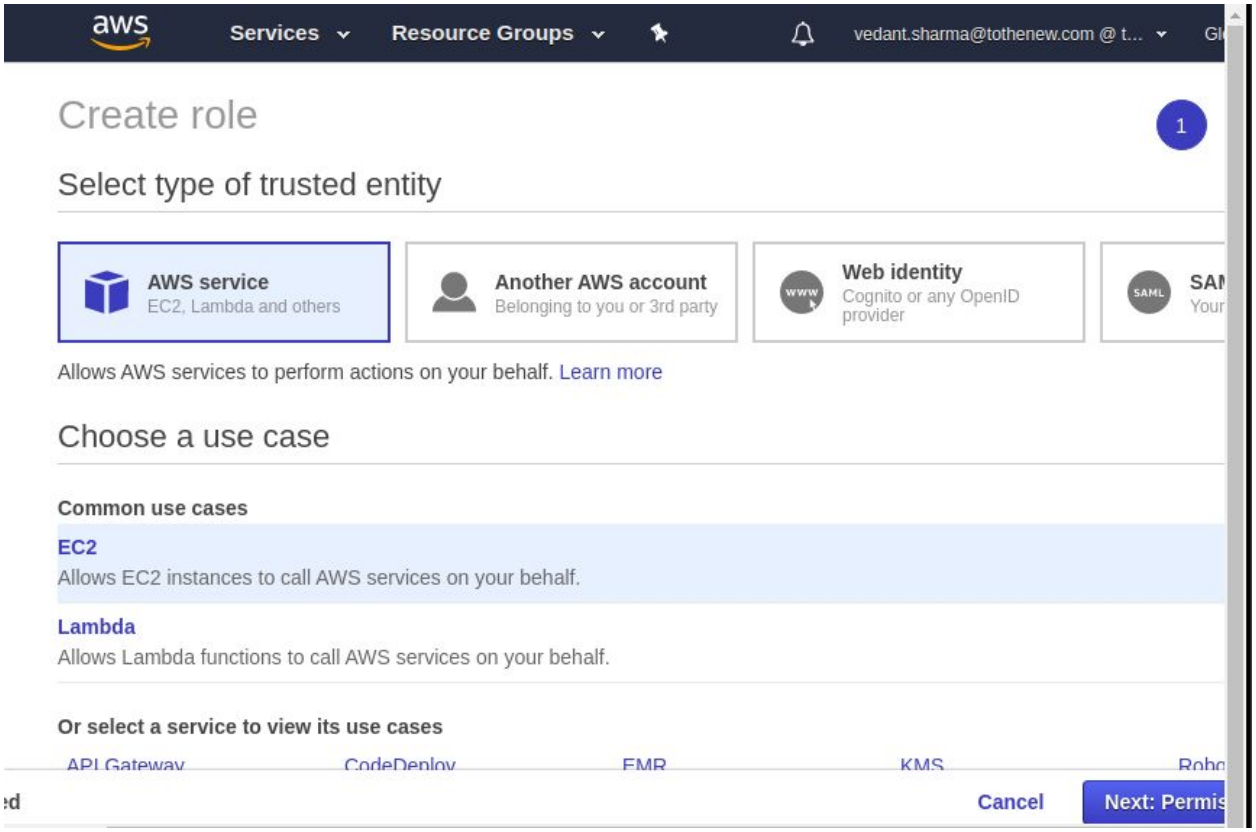
Accessing the instances mentioned in the ec2 account tagged in a particular manner taking vedant-instance as an example.

We can use aws ec2 describe-instance and using the filter utility to get the instance details based on the tags.

```
ubuntu@ip-10-0-3-131:~$ sudo aws ec2 describe-instances --filter Name=tag:
Name,Values=vedant-instance | jq
{
  "Reservations": [
    {
      "Groups": [],
      "Instances": [
        {
          "AmiLaunchIndex": 0,
          "ImageId": "ami-06273419dccc0dc8d",
          "InstanceId": "i-000dae29bc7d0ef85",
          "InstanceType": "t2.micro",
          "KeyName": "Vedant-Bootcamp",
          "LaunchTime": "2020-02-27T11:56:53.000Z",
          "Monitoring": {
            "State": "disabled"
          },
          "Placement": {
            "AvailabilityZone": "us-east-1b",
            "GroupName": "",
            "Tenancy": "default"
          },
          "PrivateDnsName": "ip-10-0-1-65.ec2.internal",
          "PrivateIpAddress": "10.0.1.65",
          "ProductCodes": [],
          "PublicDnsName": "",
          "PublicIpAddress": "3.226.248.169",
          "State": {
            "Code": 16,
            "Name": "running"
          },
          "StateTransitionReason": "",
          "SubnetId": "subnet-01d770a77bb69a1f8",
          "VpcId": "vpc-00470a42fc196d84e",
          "Architecture": "x86_64",
          "BlockDeviceMappings": [
            {
              "DeviceName": "/dev/sda1",
```

Q8. An EC2 Instance hosts a Java-based application that accesses an s3 bucket. This EC2 Instance is currently serving production users. Create the role and assign the role to EC2 instance.

Creating a new role for EC2 instance


aws
Services
Resource Groups
vedant.sharma@tothenew.com @ t...
Create role
1
Select type of trusted entity
AWS service
EC2, Lambda and others
Another AWS account
Belonging to you or 3rd party
Web identity
Cognito or any OpenID provider
Allows AWS services to perform actions on your behalf. Learn more
Choose a use case
Common use cases
EC2
Allows EC2 instances to call AWS services on your behalf.
Lambda
Allows Lambda functions to call AWS services on your behalf.
Or select a service to view its use cases
Cancel
Next: Permis

Create role

1

Attach permissions policies

Choose one or more policies to attach to your new role.

Create policy

Filter policies s3

Policy name	Used as
alice-s3-maithely	Permissions policy
AmazonDMSRedshiftS3Role	None
AmazonS3FullAccess	Permissions policy
AmazonS3ReadOnlyAccess	Permissions policy
AWSLambdaS3ExecutionRole-1a3ddce4-a989-4828-88a4-7f5e701af3ef	Permissions policy
AWSLambdaS3ExecutionRole-34f3178e-e3ee-4238-8c64-0e27432978a2	Permissions policy

Cancel Previous Next

Create role

1

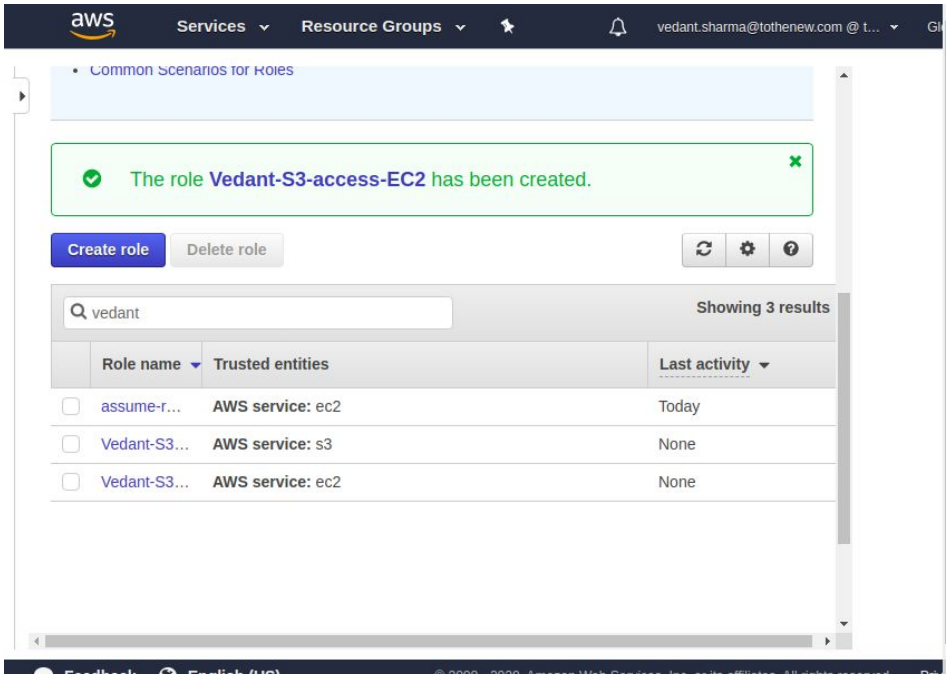
Add tags (optional)

IAM tags are key-value pairs you can add to your role. Tags can include user information, such as an email address, or can be des
title. You can use the tags to organize, track, or control access for this role. Learn more

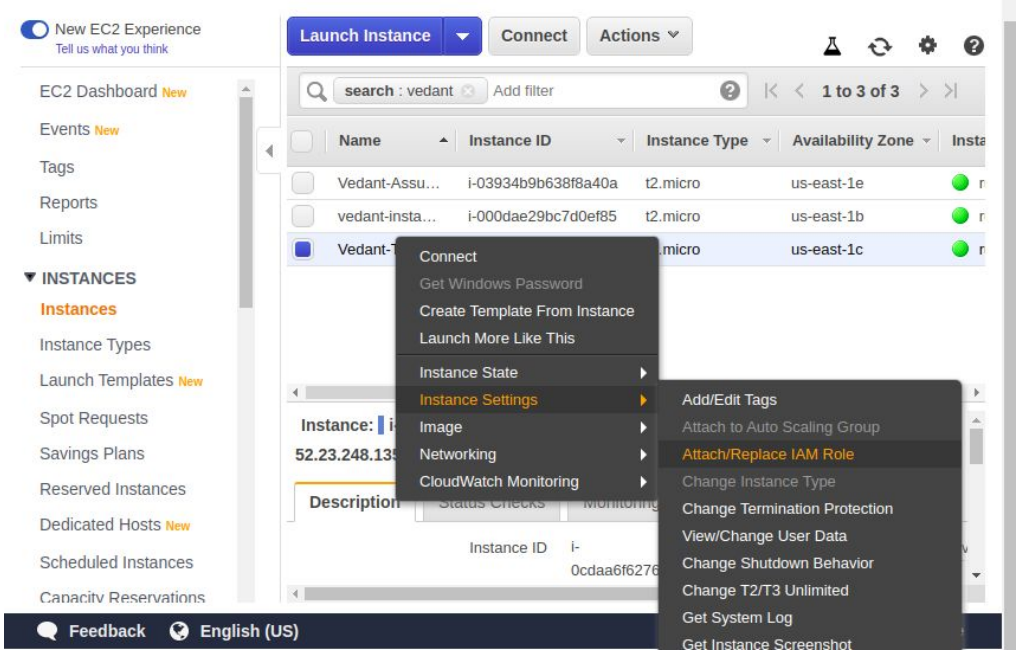
Key	Value (optional)
Name	S3full-access-EC2-Vedant
Add new key	

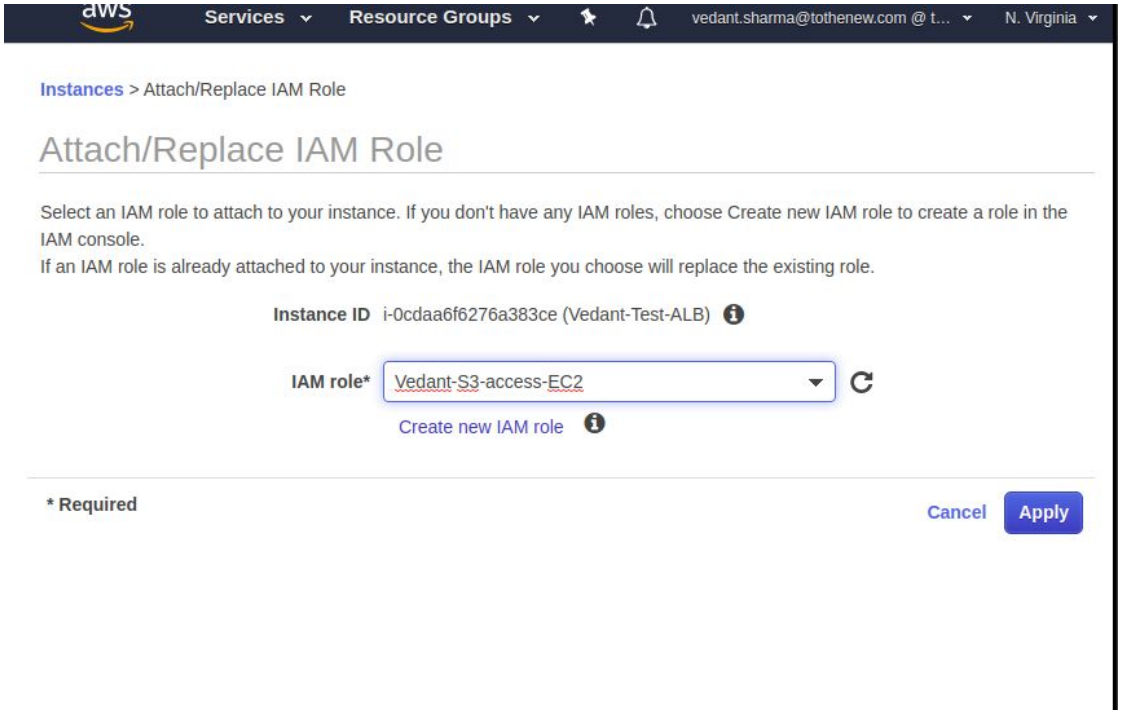
You can add 49 more tags.

Cancel Previous Next: R



Attaching role to EC2





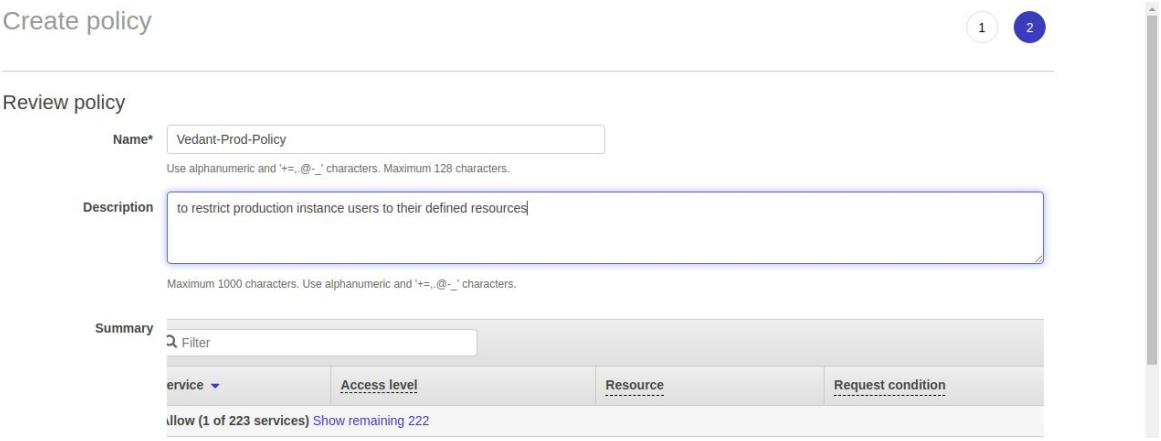
Accessing buckets from the given instance:

Q9. You have both production and development based instances running on your VPC. It is required to ensure that people responsible for the development instances do not have access to work on production instances for better security. Define the tags on the test and production servers and add a condition to the IAMPolicy which allows access to specific tags.

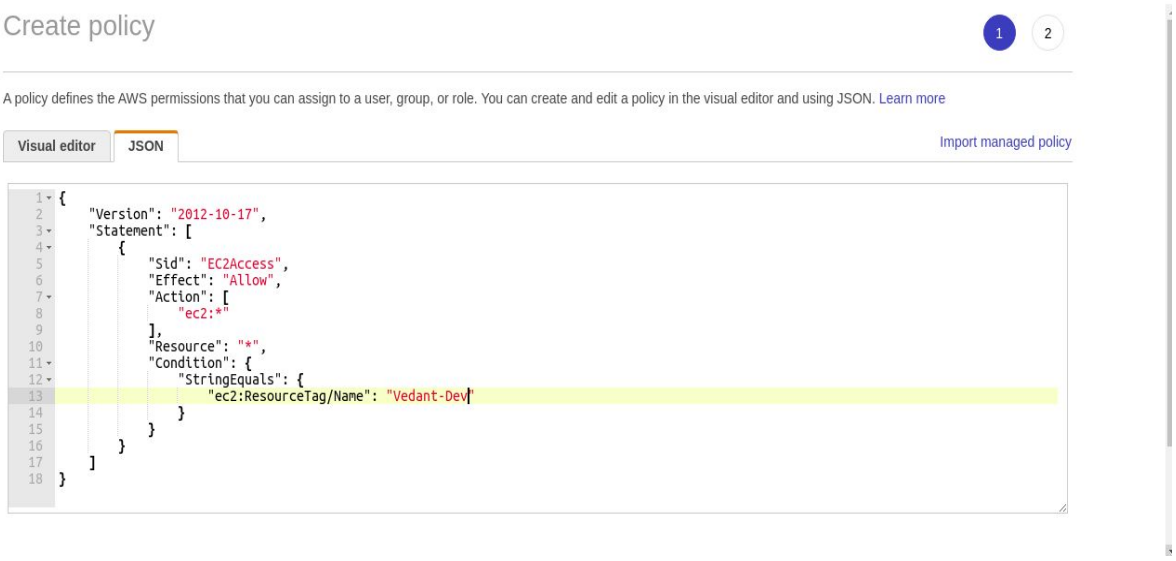
```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": [
```

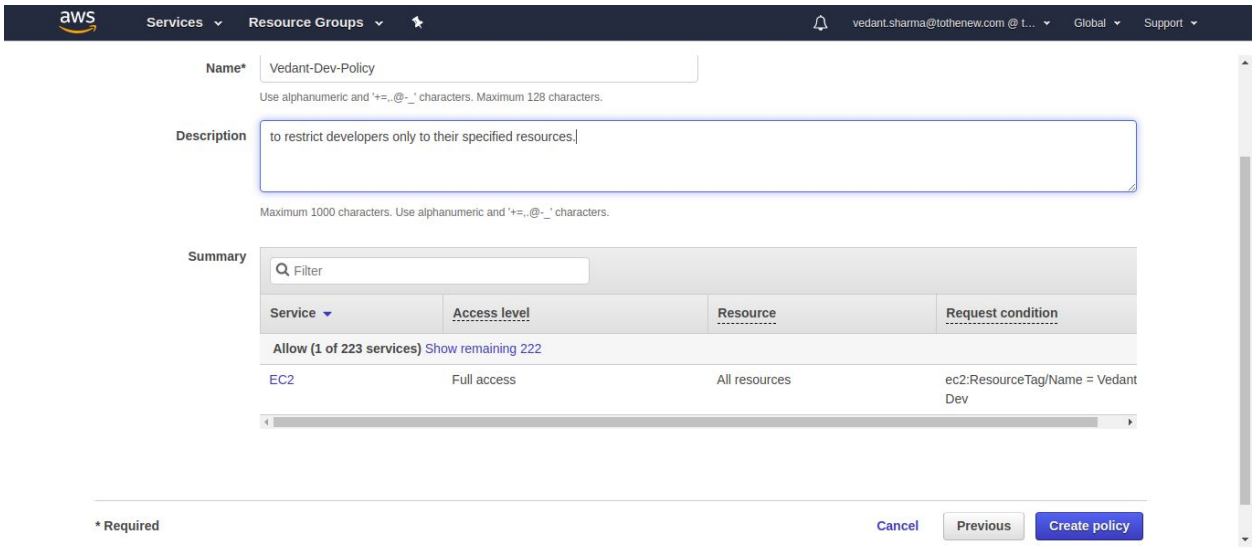
```
        "ec2 *"
      ],
      "Resource": "*",
      "Condition": {"StringEquals":
        {"Condition": {"StringLike": {"iam:ResourceTag/Name": "Vedant-Prod"}}
    }
}
}
}
```

Creating a Policy and adding the policy



Creating policy for developers:





Attaching the created policies to the respective groups:

aws Services Resource Groups vedant.sharma@tothenew.com @ t... Global Support

Identity and Access Management (IAM)

- Dashboard
- Access management
  - Groups
  - Users
  - Roles
  - Policies
  - Identity providers
  - Account settings
- Access reports
  - Access analyzer
    - Archive rules
    - Analyzer details
  - Credential report
  - Organization activity

IAM > Groups > **Vedant-production-Group**

Summary

Group ARN:	arn:aws:iam::187632318301:group/Vedant-production-Group
Users (in this group):	1
Path:	/
Creation Time:	2020-02-28 13:23 UTC+0530

Users | Permissions | Access Advisor

Managed Policies

The following managed policies are attached to this group. You can attach up to 10 managed policies.

Attach Policy

Policy Name	Actions
Alice-vedant-Policy	Show Policy | Detach Policy | Simulate Policy

Inline Policies

aws Services Resource Groups vedant.sharma@tothenew.com @ t... Global Support

Attach Policy

Attach Policy

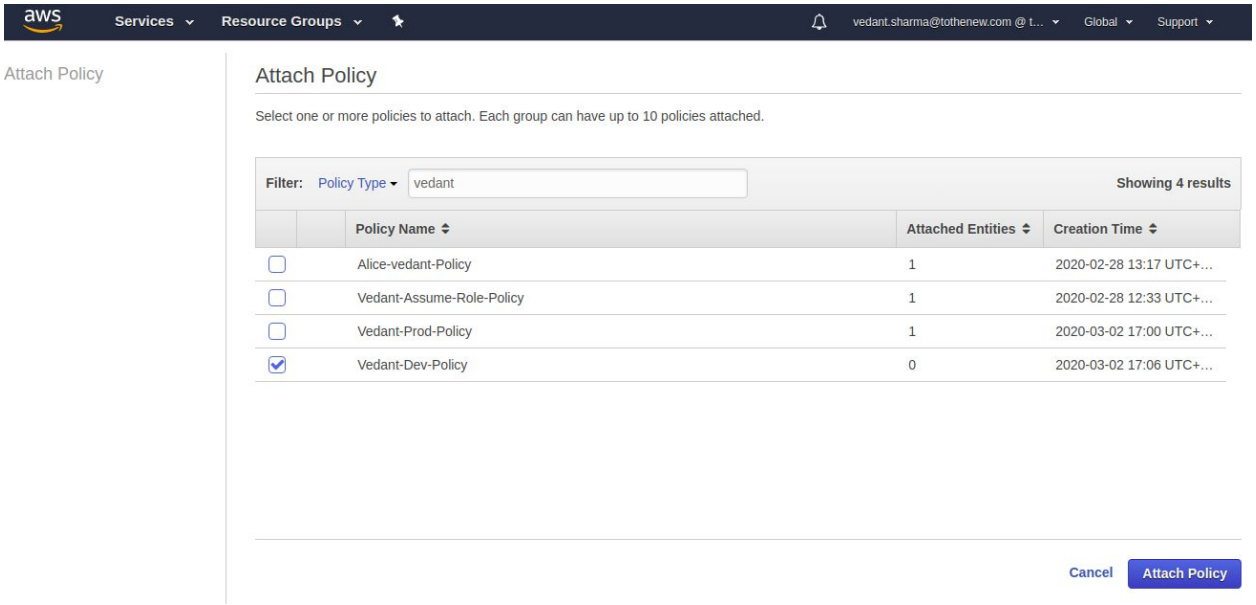
Select one or more policies to attach. Each group can have up to 10 policies attached.

Filter: Policy Type vedant — Showing 4 results

	Policy Name	Attached Entities	Creation Time
☐	bob-vedant-ec2-access	1	2020-02-28 15:35 UTC+...
☐	Vedant-Assume-Role-Policy	1	2020-02-28 12:33 UTC+...
☐	Vedant-Dev-Policy	0	2020-03-02 17:06 UTC+...
☑	Vedant-Prod-Policy	0	2020-03-02 17:00 UTC+...

Cancel | Attach Policy

Attaching policy to the developer group:



Q10. Create a policy for allowing users to set or rotate their credentials, such as their console password, their programmatic access keys, and their MFA devices.

Creating a policy and associating with user