ACTIVITY 1 – IAM Users & Authentication

Objective

Understand IAM users and authentication mechanisms.

Tasks

1. Create an IAM user named: intern-user1

2. Enable AWS Management Console access

3. Set a custom password

4. Log in using the IAM user and verify:

○ You cannot access S3

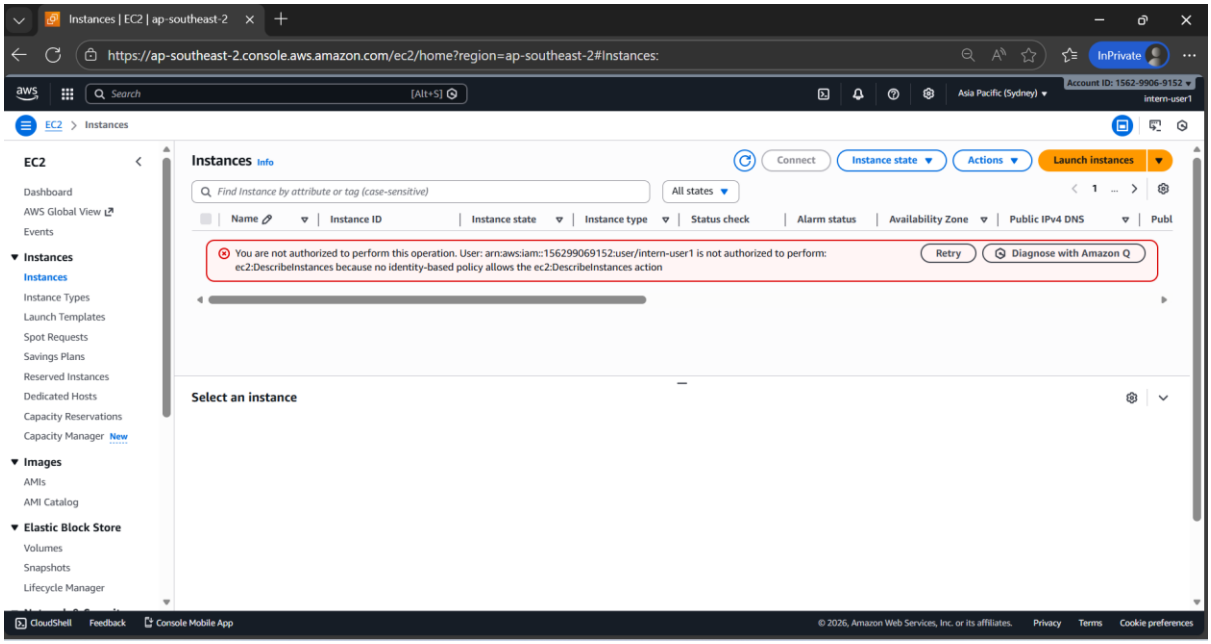
○ You cannot access EC2

Expected Outcome

● User exists

● Login works

● Everything shows Access Denied

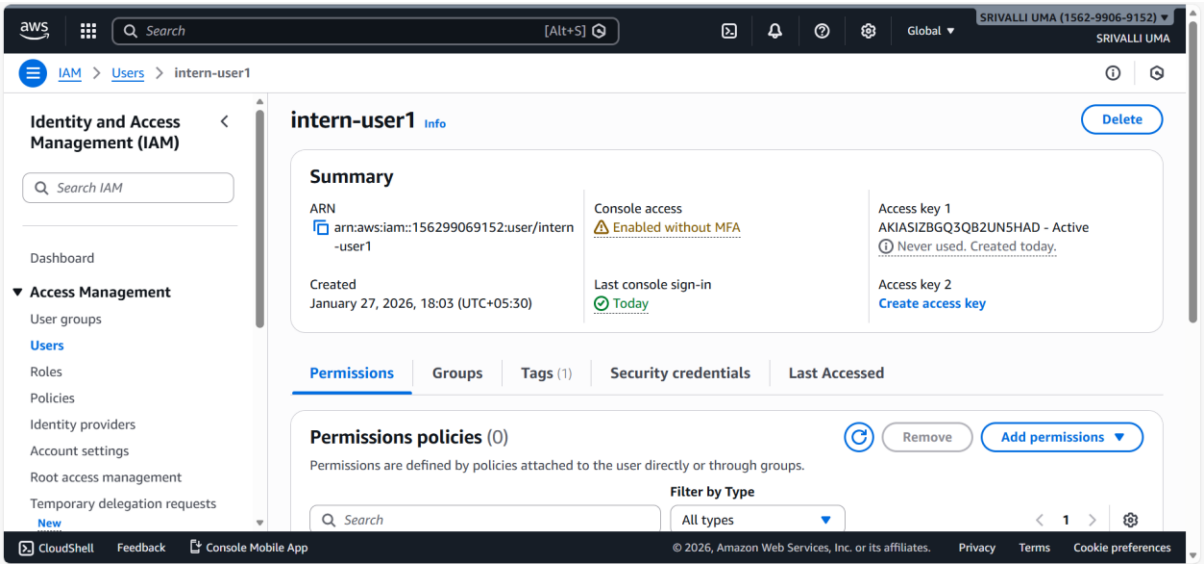



```
C:\Users\260016544>aws configure
AWS Access Key ID [****************5HAD]: AKIASIZBGQ3QB2UN5HAD
AWS Secret Access Key [****************qNCz]: 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
Default region name [ap-southeast-2]: ap-southeast-2
Default output format [JSON]: JSON

C:\Users\260016544>aws s3 ls

An error occurred (AccessDenied) when calling the ListBuckets operation: User: arn:aws:iam::156299069152:user/intern-user1 is not authorized to perform: s3:ListAllMyBuckets
 because no identity-based policy allows the s3:ListAllMyBuckets action

C:\Users\260016544>
```



ACTIVITY 2 – IAM Groups & Permission Inheritance

Objective

Understand why groups exist and how permissions flow.

Tasks

1. Create two IAM groups:

BackendTeam

DatabaseTeam

2. Add intern-user1 to BackendTeam

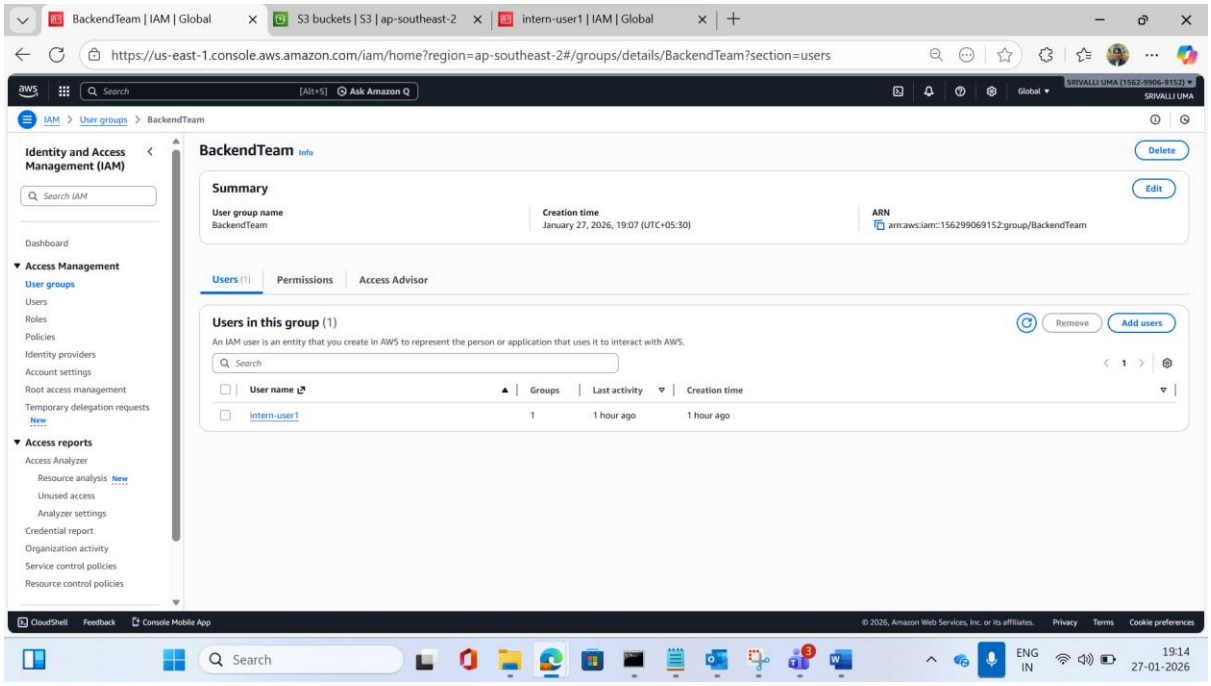
3. Attach AWS managed policy:

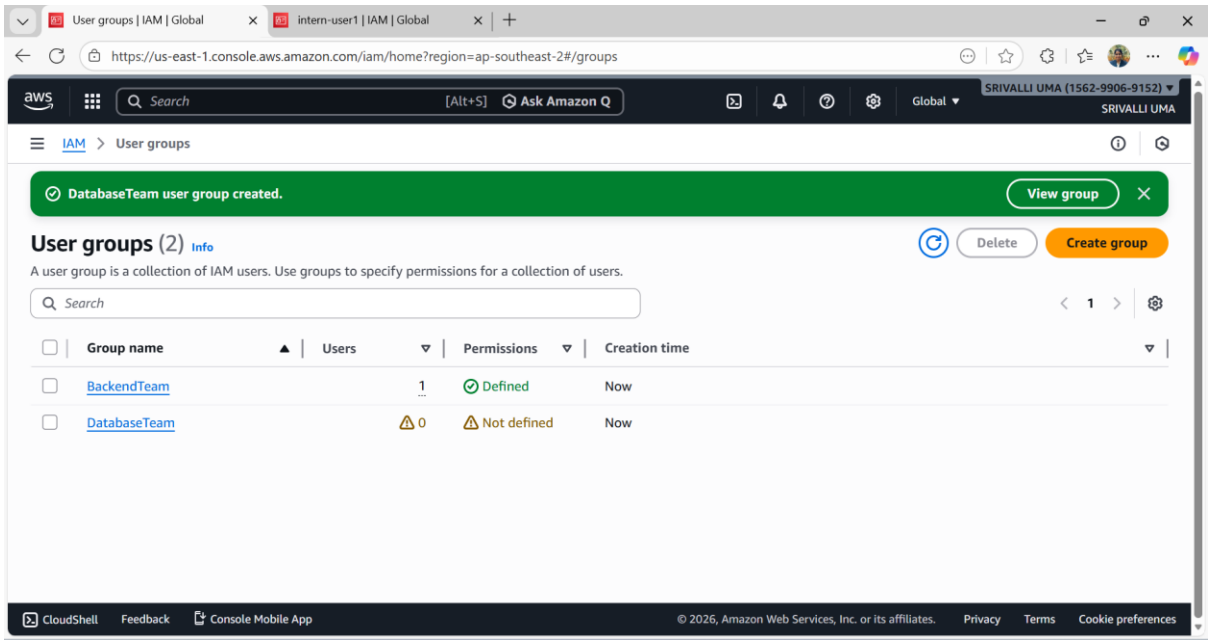
○ AmazonS3ReadOnlyAccess to BackendTeam

4. Login as intern-user1 and:

○ List S3 buckets (should work)

○ Upload file (should fail)






```
C:\Users\260016544>aws s3 ls
2026-01-27 19:12:03 uma-backend-s3
```

```
C:\Users\260016544>echo "Hello, world!!" > file1.txt

C:\Users\260016544>aws s3 cp file1.txt s3://uma-backend-s3/
upload failed: .\file1.txt to s3://uma-backend-s3/file1.txt An error occurred (AccessDenied) when calling the PutObject operation: User: arn:aws:iam::156299069152:user/inte
rn-user1 is not authorized to perform: s3:PutObject on resource: "arn:aws:s3:::uma-backend-s3/file1.txt" because no identity-based policy allows the s3:PutObject action

C:\Users\260016544>
```

Questions to Answer

● Why upload is denied?

The AmazonS3ReadOnlyAccess policy only grants read-only permissions for S3 resources. But it does not include write permissions. So when the user intern-user1 tries to upload then the IAM denies the request because the permissions required is missing. To enable the upload we need to add PutObject.

● Where did the permission come from?

We attached the policy to the BackendTeam group and intern-user1 was a part of that group so the permissions for ReadOnlyAccess was enabled. So, the list buckets permission came from the AmazonS3ReadOnlyAccess policy attached to BackendTeam.

ACTIVITY 3 – Custom Policy Using Visual Editor

Objective

Create a least-privilege custom policy

Scenario

Backend team needs:

● Read & write objects in one specific S3 bucket

● No delete permission

Tasks

1. Create a policy using Visual Editor

2. Service: S3

3. Allow:

○ GetObject

○ PutObject

○ ListBucket

4. Restrict access to:

my-backend-bucket

5. Attach policy to BackendTeam

6. Test:

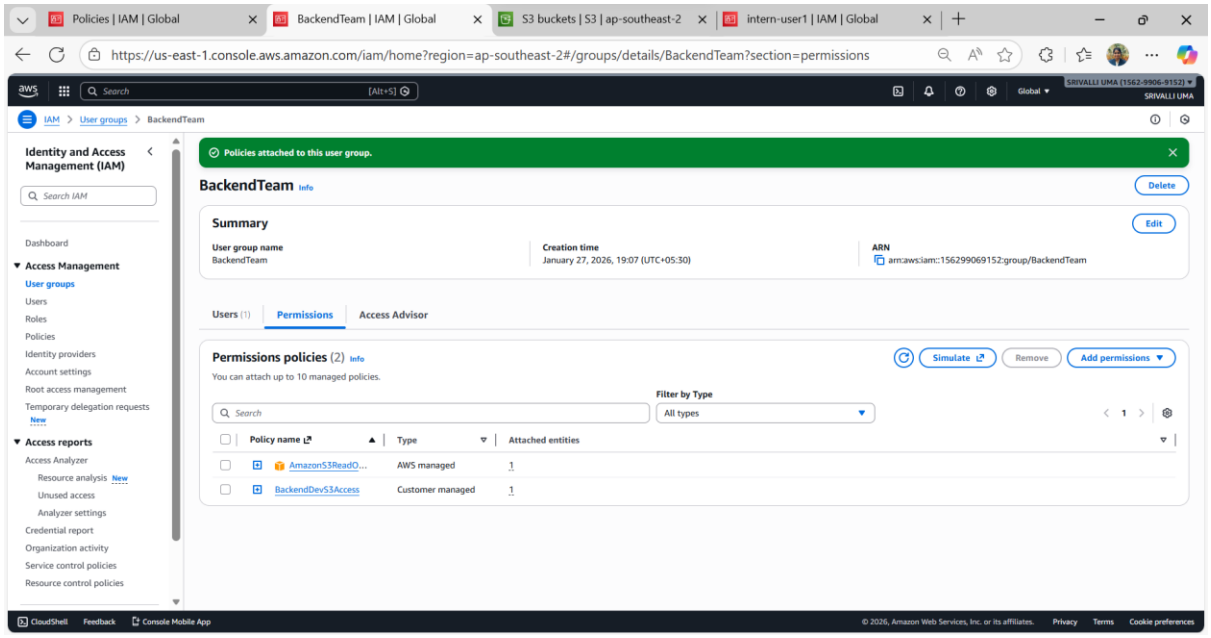
○ Upload file → allowed

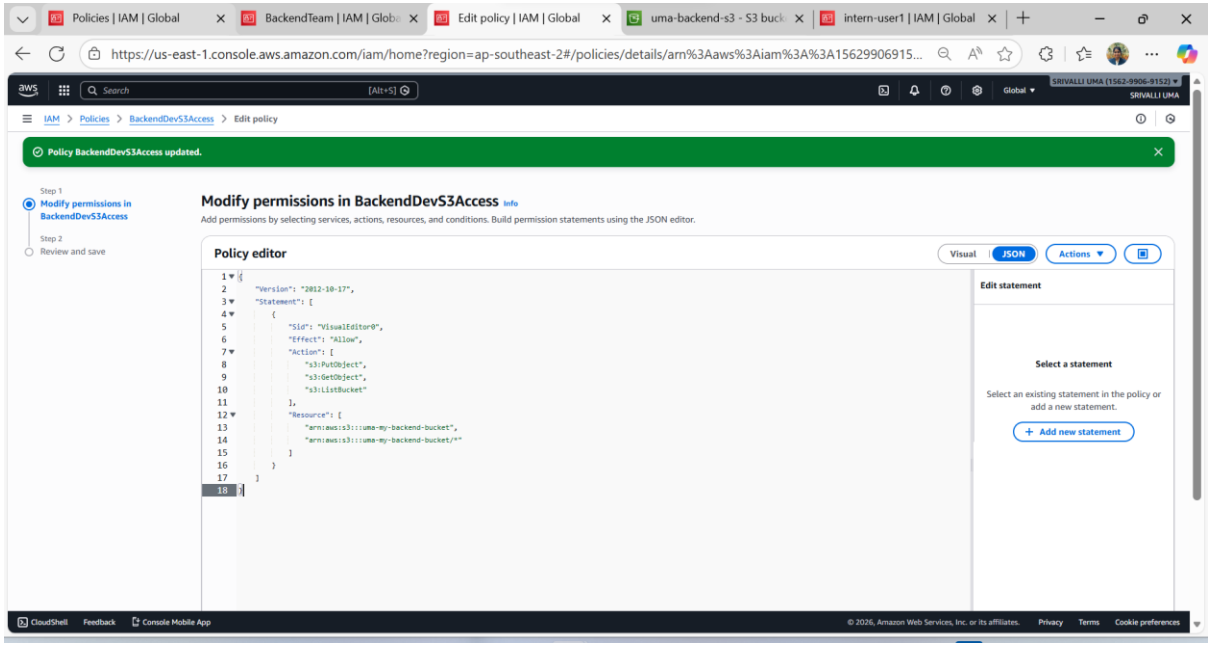
○ Delete file → denied

Expected Outcome

● Fine-grained access

● No admin permissions






```
C:\Users\260016544>aws s3 cp fileA.txt s3://uma-my-backend-bucket/
upload: .\fileA.txt to s3://uma-my-backend-bucket/fileA.txt

C:\Users\260016544>aws s3 rm s3://uma-my-backend-bucket/fileA.txt
delete failed: s3://uma-my-backend-bucket/fileA.txt An error occurred (AccessDenied) when calling the DeleteObject operation: User: arn:aws:iam::156299069152:user/intern-us
er1 is not authorized to perform: s3:DeleteObject on resource: "arn:aws:s3:::uma-my-backend-bucket/fileA.txt" because no identity-based policy allows the s3:DeleteObject ac
tion

C:\Users\260016544>
```

ACTIVITY 4 – Custom Policy Using JSON Editor

Objective

Understand IAM policy JSON structure.

Scenario

Frontend team can:

- Start/Stop only one EC2 instance
- Cannot access other instances

Tasks

1. Create a policy using JSON editor
2. Allow actions:
   - ec2:StartInstances
   - ec2:StopInstances
   - ec2:DescribeInstances
3. Restrict resource to:

One specific EC2 instance ARN

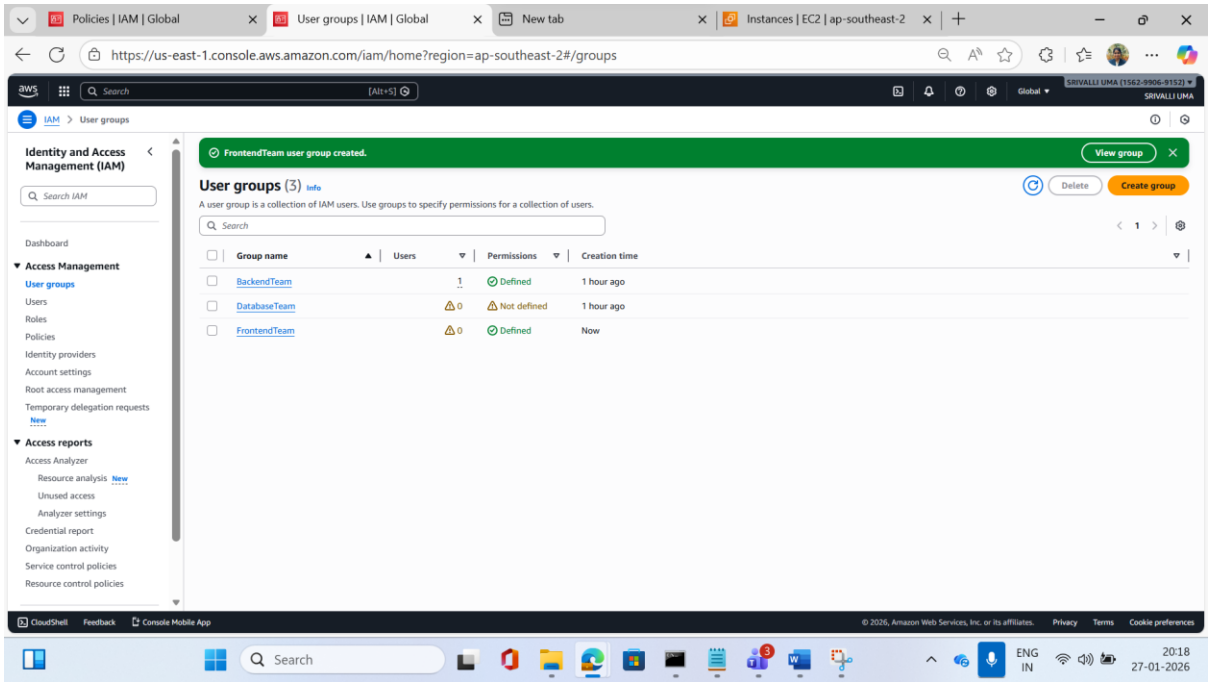
4. Create group:

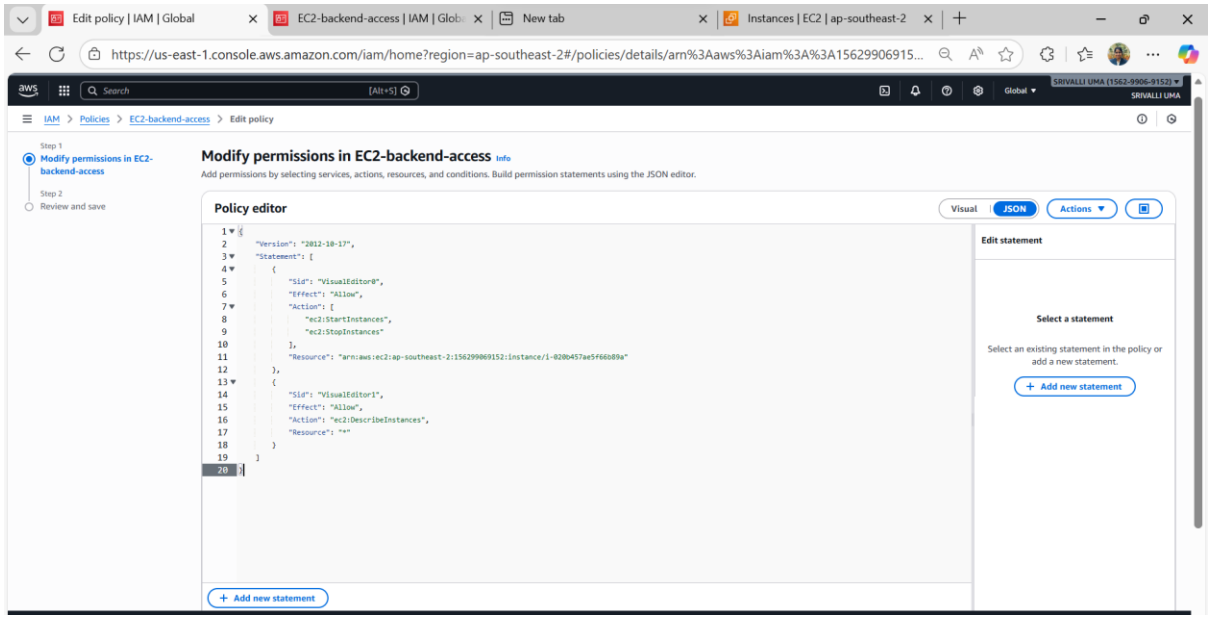
FrontendTeam

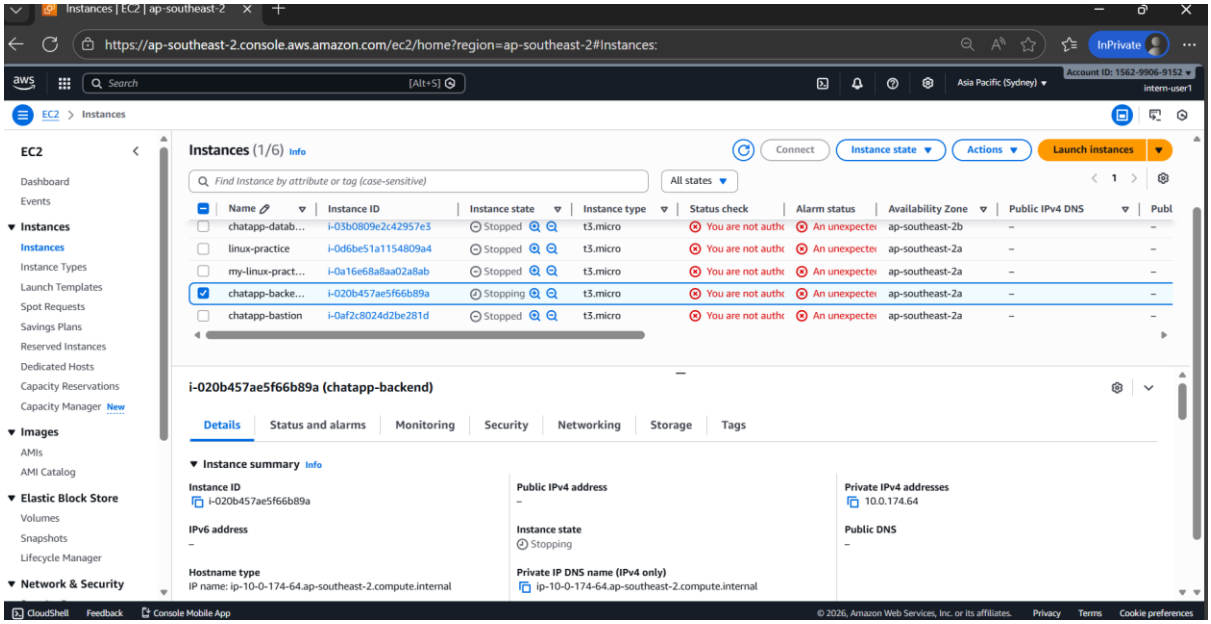
5. Attach policy
6. Test by logging in as frontend user

Expected Outcome

● Only one EC2 visible & controllable






Instances | EC2 | ap-southeast-2
https://ap-southeast-2.console.aws.amazon.com/ec2/home?region=ap-southeast-2#Instances:
InPrivate
aws
Search
[Alt+S]
Asia Pacific (Sydney)
Account ID: 1562-9906-9152
intern-user1
EC2 > Instances
EC2
Dashboard
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Capacity Manager New
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Instances (1/6) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4 DNS
Publ
chatapp-datab...
i-03b0809e2c42957e3
Stopped
t3.micro
You are not authc
An unexpecte
ap-southeast-2b
linux-practice
i-0d6be51a1154809a4
Stopped
t3.micro
You are not authc
An unexpecte
ap-southeast-2a
my-linux-pract...
i-0a16e68a8aa02a8ab
Stopped
t3.micro
You are not authc
An unexpecte
ap-southeast-2a
chatapp-backe...
i-020b457ae5f66b89a
Stopping
t3.micro
You are not authc
An unexpecte
ap-southeast-2a
chatapp-bastion
i-0af2c8024d2be281d
Stopped
t3.micro
You are not authc
An unexpecte
ap-southeast-2a
i-020b457ae5f66b89a (chatapp-backend)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-020b457ae5f66b89a
IPv6 address
–
Hostname type
IP name: ip-10-0-174-64.ap-southeast-2.compute.internal
Public IPv4 address
–
Instance state
Stopping
Private IP DNS name (IPv4 only)
ip-10-0-174-64.ap-southeast-2.compute.internal
Private IPv4 addresses
10.0.174.64
Public DNS
–
CloudShell
Feedback
Console Mobile App
© 2026, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences