IAM Roles Assignment

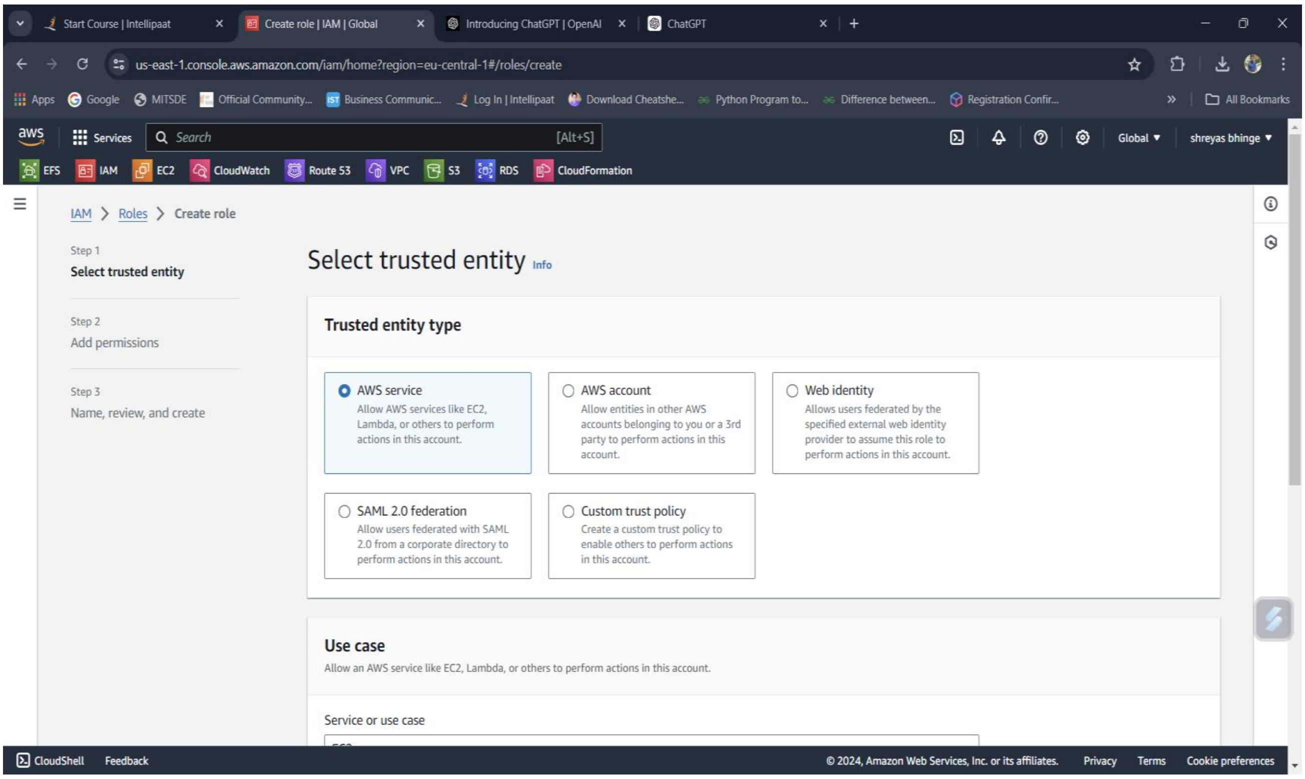
(Shreyas Bhinge)

Tasks To Be Performed:

1. Create a role which only lets user1 and user2 from task 1 to have complete access to VPCs and DynamoDB.

2. Login into user1 and shift to the role to test out the feature.

Step1. Creating a role for Dev1 and Dev2



IAM > Roles > Create role

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Select trusted entity Info

Trusted entity type

● AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

○ Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

○ SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

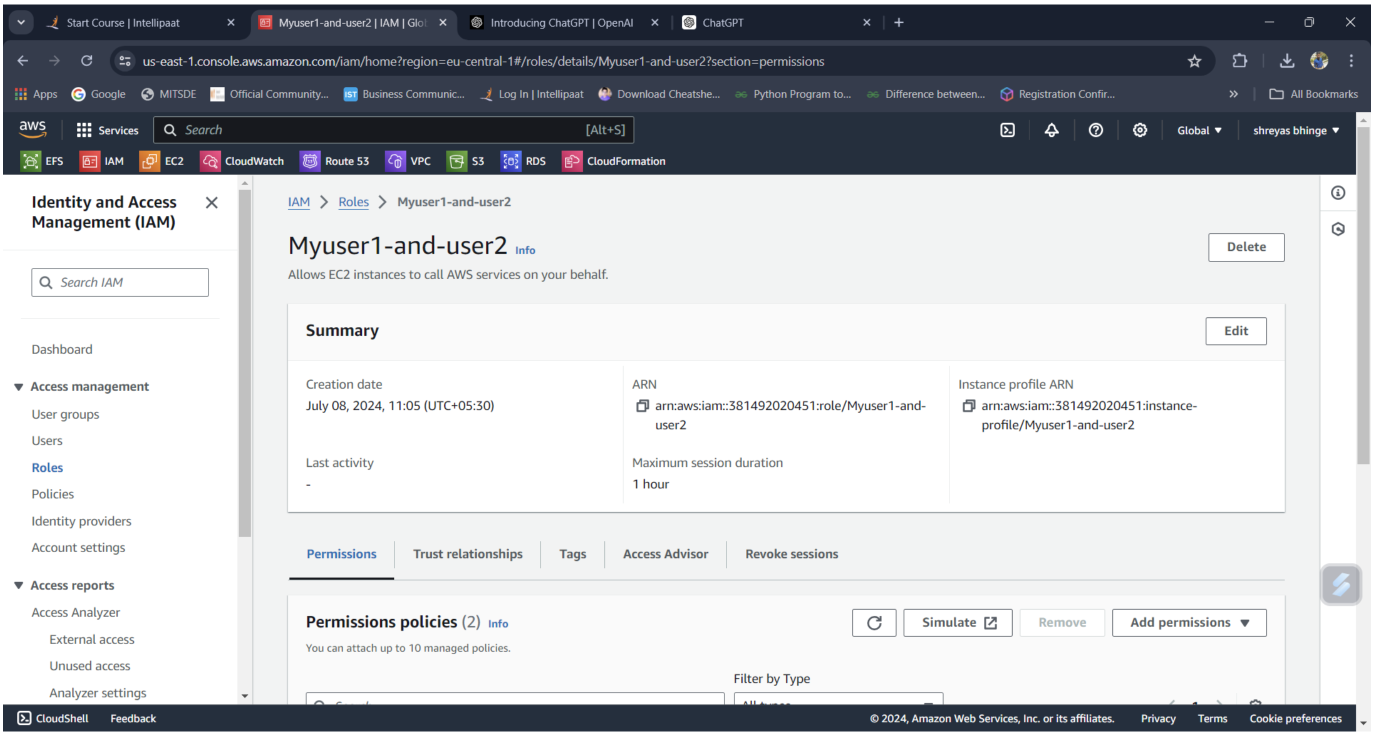
○ Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

Step 2: Attach Policies to the Role



Myuser1-and-user2 Info

Allows EC2 instances to call AWS services on your behalf.

Summary

Creation date: July 08, 2024, 11:05 (UTC+05:30)

ARN: arn:aws:iam::381492020451:role/Myuser1-and-user2

Instance profile ARN: arn:aws:iam::381492020451:instance-profile/Myuser1-and-user2

Last activity: -

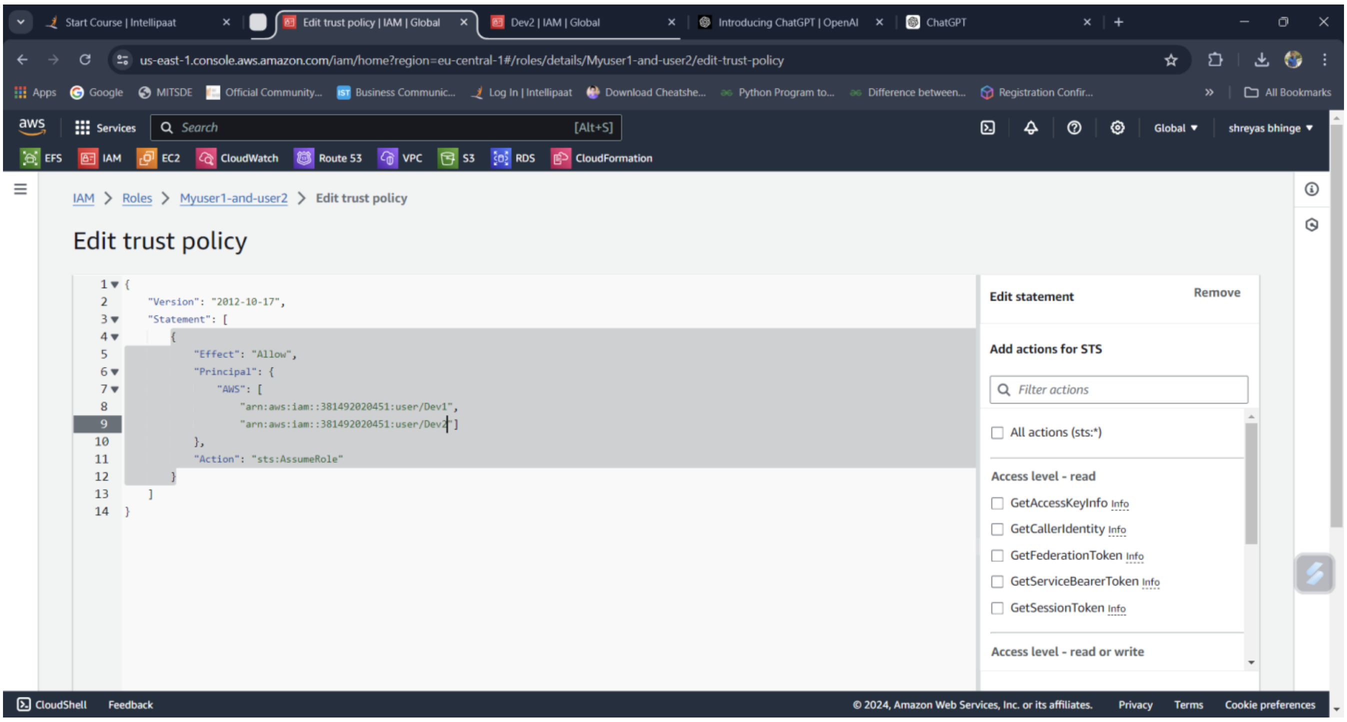
Maximum session duration: 1 hour

Permissions policies (2) Info

You can attach up to 10 managed policies.

Filter by Type

Step 3: Modify the Trust Relationship



Edit trust policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::381492020451:user/Dev1",
                    "arn:aws:iam::381492020451:user/Dev2"]
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

□ All actions (sts:*)

Access level - read

□ GetAccessKeyInfo Info

□ GetCallerIdentity Info

□ GetFederationToken Info

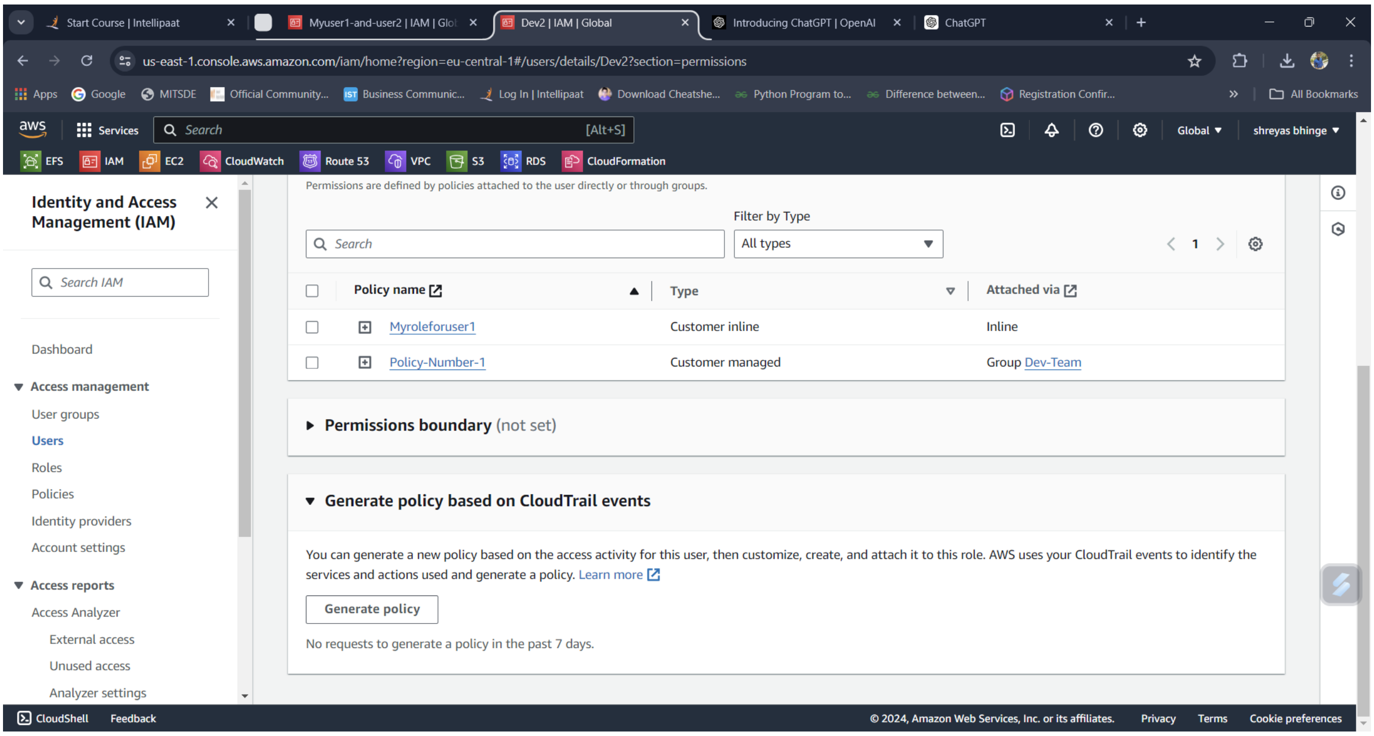
□ GetServiceBearerToken Info

□ GetSessionToken Info

Access level - read or write

Step 4: Assign Role to Users

A) Assign Role for Dev1 as Myroleforuser1



Permissions are defined by policies attached to the user directly or through groups.

Filter by Type: All types

Policy name	Type	Attached via
Myroleforuser1	Customer inline	Inline
Policy-Number-1	Customer managed	Group Dev-Team

Permissions boundary (not set)

Generate policy based on CloudTrail events

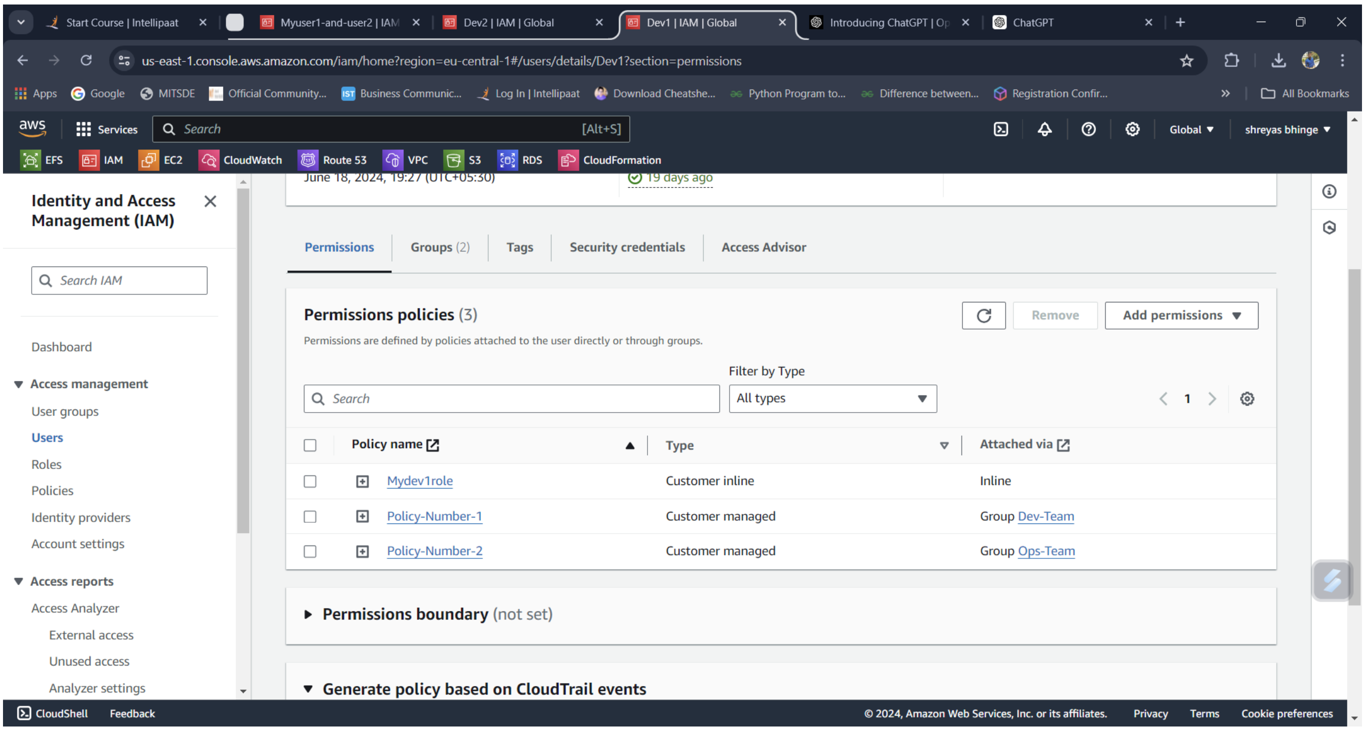
You can generate a new policy based on the access activity for this user, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more

Generate policy

No requests to generate a policy in the past 7 days.

© 2024, Amazon Web Services, Inc. or its affiliates.

B) Assign Role for Dev2 as Mydev1role



June 18, 2024, 19:27 (UTC+05:30) — 19 days ago

Permissions policies (3)

Permissions are defined by policies attached to the user directly or through groups.

Filter by Type: All types

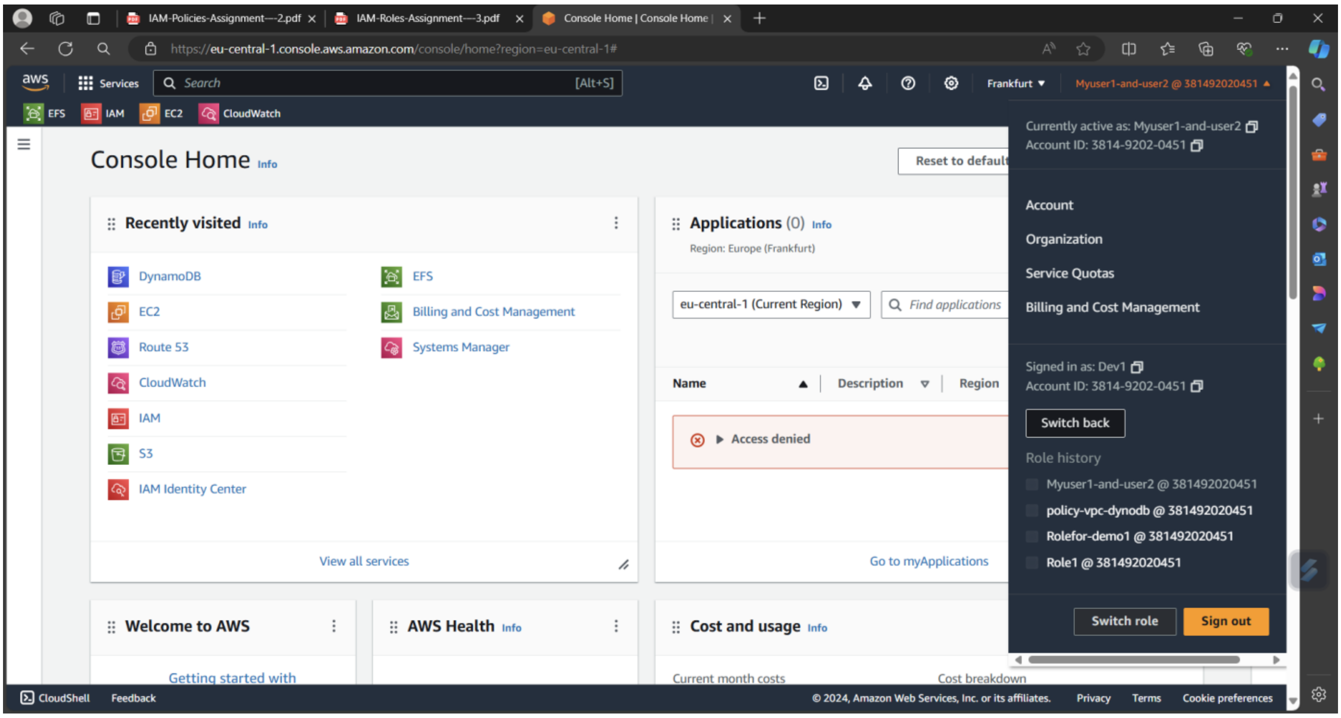
Policy name	Type	Attached via
Mydev1role	Customer inline	Inline
Policy-Number-1	Customer managed	Group Dev-Team
Policy-Number-2	Customer managed	Group Ops-Team

Permissions boundary (not set)

Generate policy based on CloudTrail events

© 2024, Amazon Web Services, Inc. or its affiliates.

Step 5. Switched the role to Myuser1-and-user2



https://eu-central-1.console.aws.amazon.com/console/home?region=eu-central-1#

Currently active as: Myuser1-and-user2
Account ID: 3814-9202-0451

Signed in as: Dev1
Account ID: 3814-9202-0451

Role history
- Myuser1-and-user2 @ 381492020451
- policy-vpc-dynodb @ 381492020451
- Rolefor-demo1 @ 381492020451
- Role1 @ 381492020451