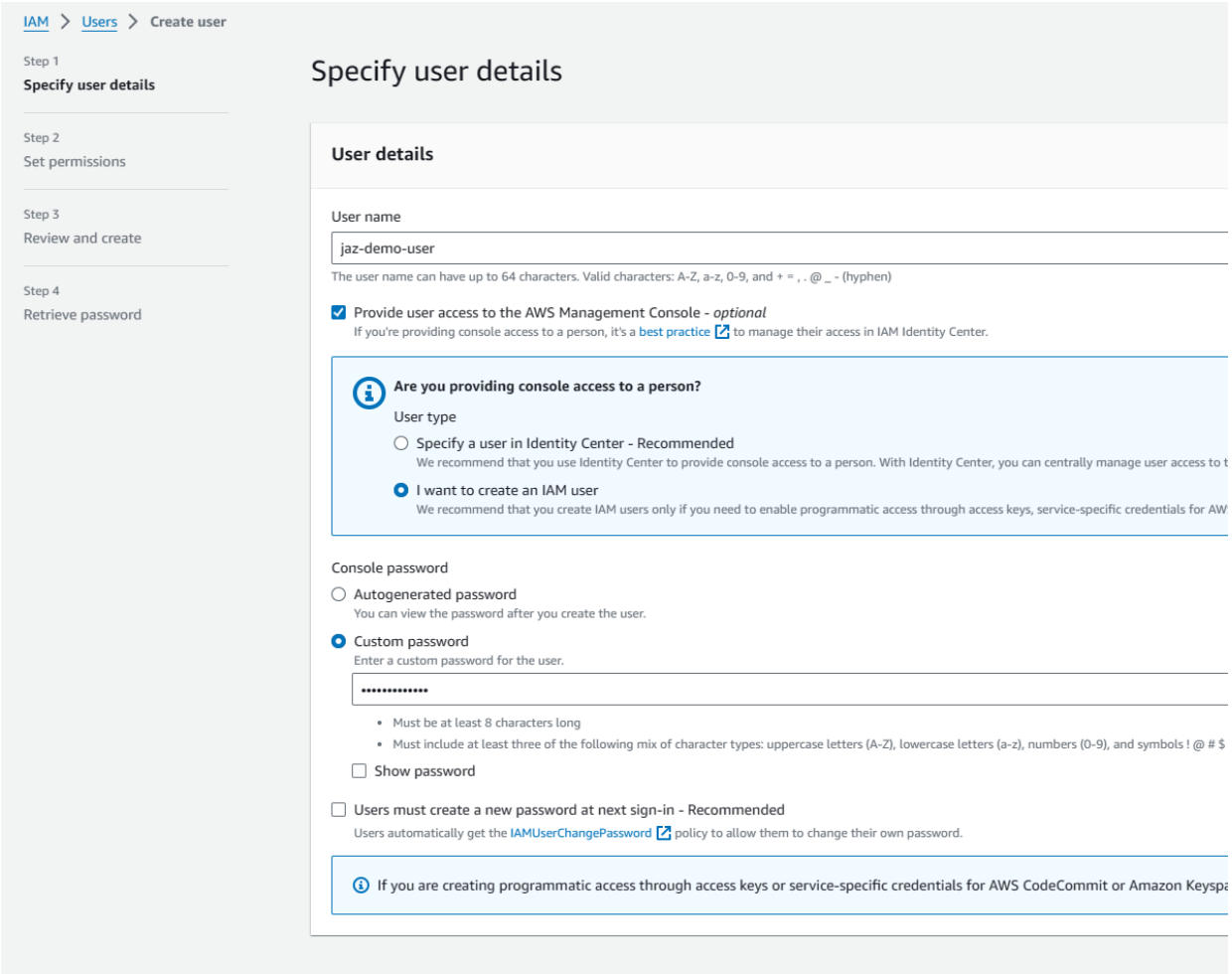
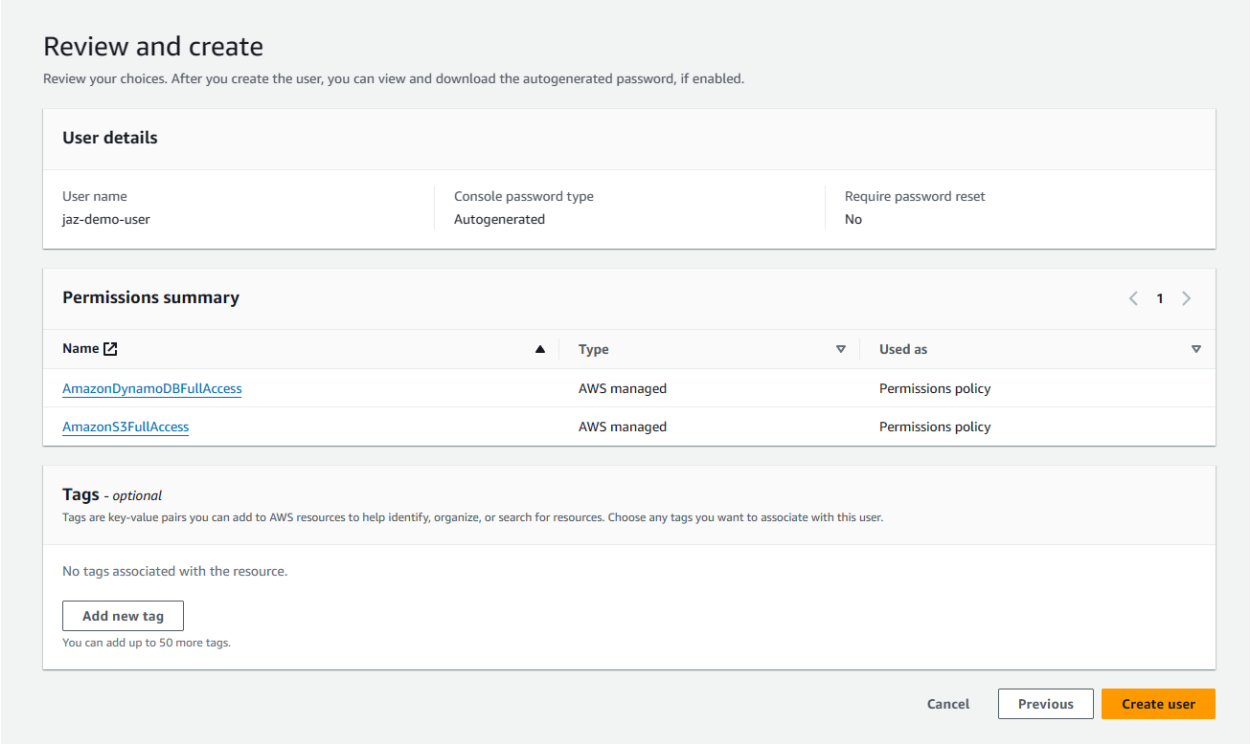
Activity 1: Recap on creating an IAM user and using permission boundaries

For Step 4 of this activity, feel free to also experiment by attaching other AWS Managed policies to other services as well(As long as it isn't AdministratorAccess or PowerUserAccess)

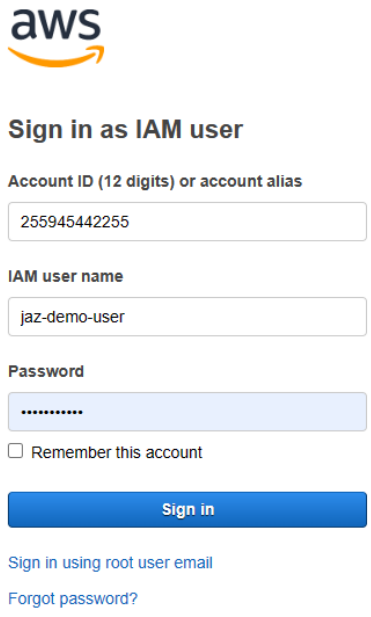
1. Go to the IAM service in AWS
2. On the left, click on the “Users” tab and select “Create User”
3. Create a user with AWS Management console access and use a custom password. Also uncheck the tickbox which says “Users must create a new password at next sign-in - Recommended”. So your page should look something like this below(without the blue box):



4. In the set permissions page, select “Attach policies directly” and attach the following AWS Managed policies to this user:
    a. AmazonS3FullAccess
    b. AmazonDynamoDBFullAccess
5. And then select “Create User”



6. Open a new in-cognito or private browser. Here, we will log into the AWS account with the new user created.Type https://255945442255.signin.aws.amazon.com/console and key in the IAM user name and password that you just created.



7. Once logged in, verify that the user has the relevant access to the services you have granted access to. In this case, S3 and DynamoDB based on the above example.

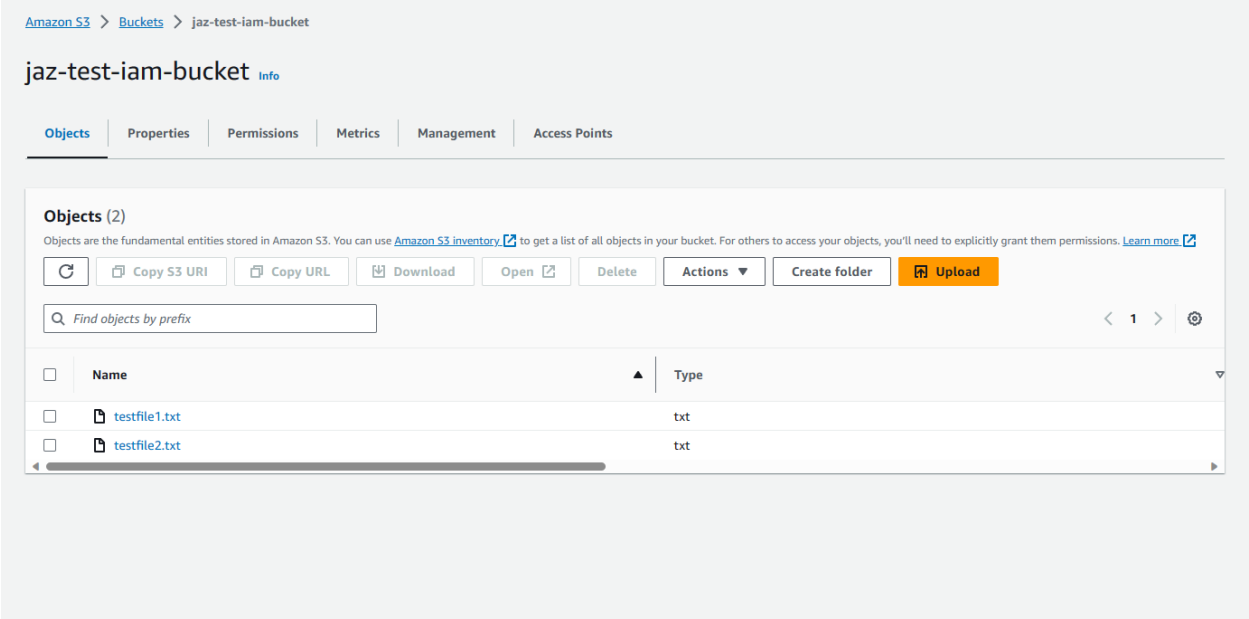
8. Now we can attach a permission boundary to the above created user.

9. Go back to your original window / main aws user, and then attach the following permission boundary to the IAM user:
    a. AmazonS3FullAccess

10. Recap: After attaching the above permission boundary, what would be the resulting permissions of the IAM user? Verify this by going back to your incognito window and logging in into your IAM user.
11. After this activity, remember to delete the IAM user that you created in this activity.

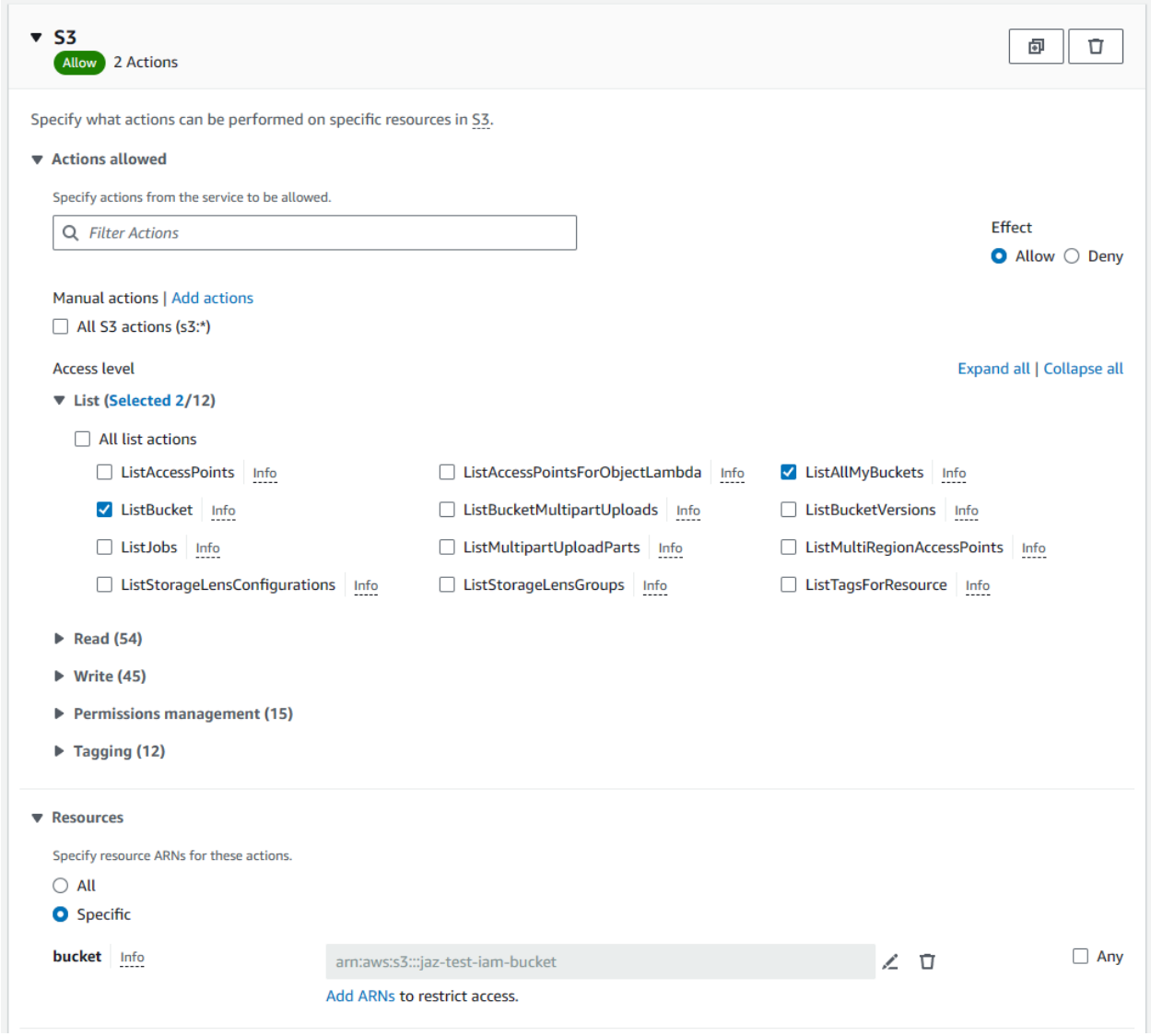
Activity 2: Creating an IAM Role for EC2 to only be able to list objects in a specific S3 bucket

1. Create a new S3 bucket with default configurations and upload some random files into it

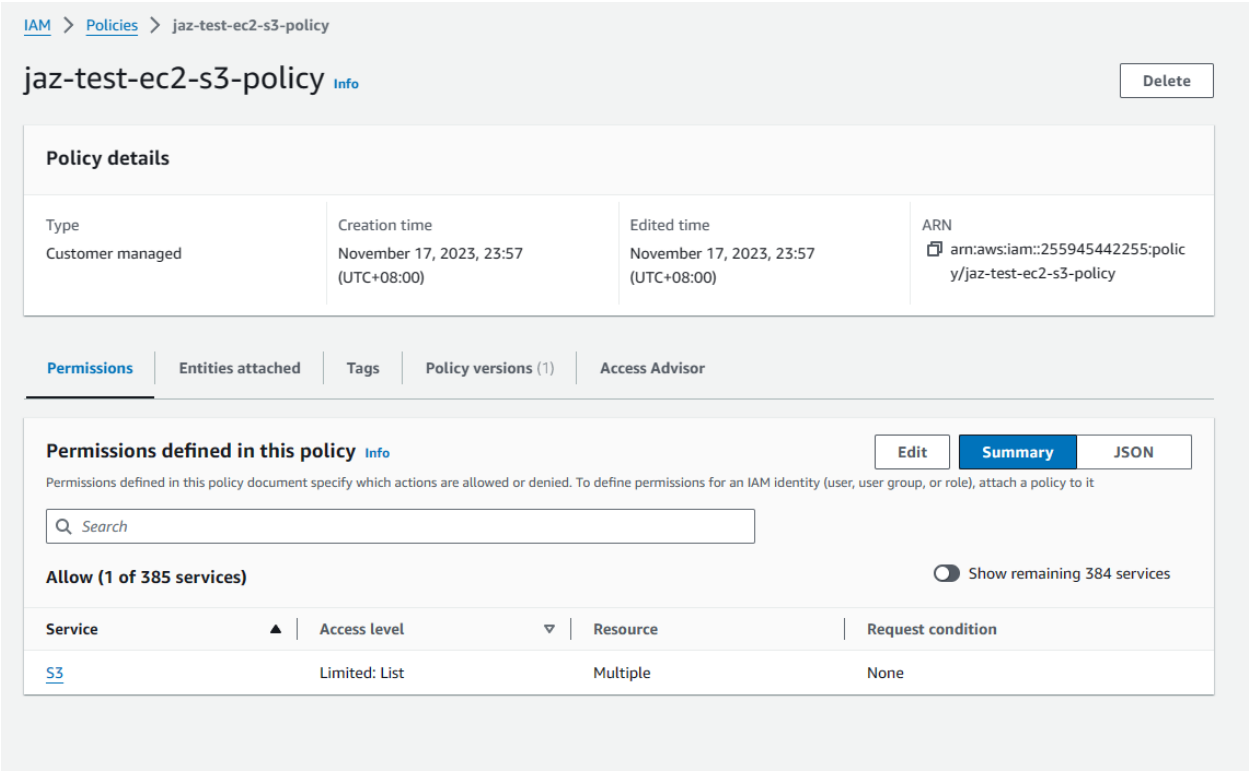


2. Create EC2 with the following configurations:
   a. Amazon Linux 2023 AMI
   b. In any of the 2 public subnets in c4_sandbox_vpc
   c. Attach “allow-ssh” security group
   d. Select Advanced Details -> Metadata Version -> Set to V1 and V2(token optional)
   e. 

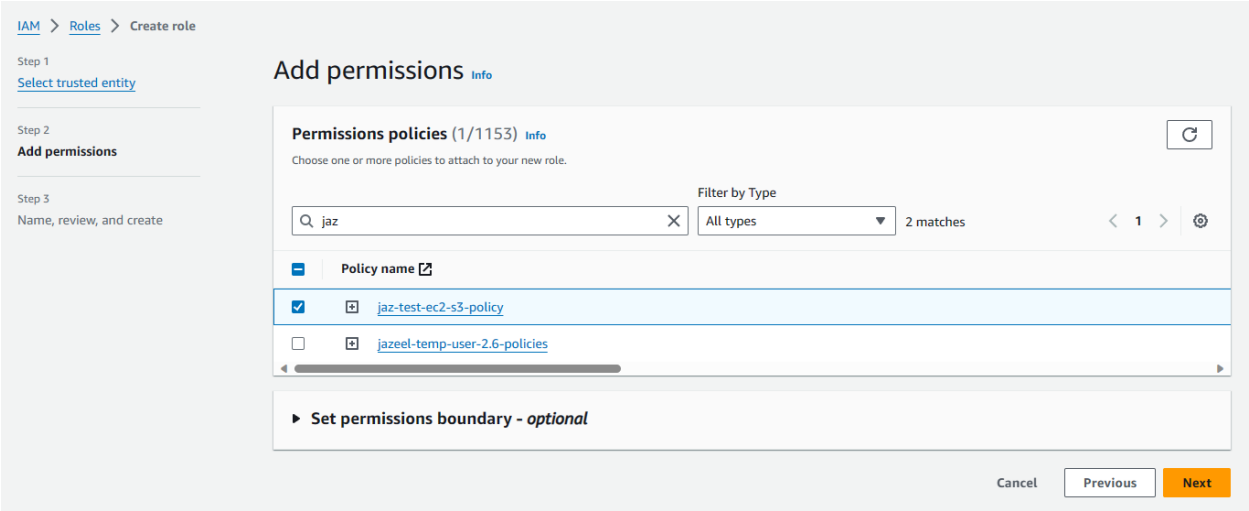
3. In this case, since we want to restrict our EC2 only to a specific bucket, we cant use AWS Managed policies anymore. We have to create our own custom policy to achieve this.
4. Go to IAM service -> Click on policies on the left and select “Create Policy”
5. For the services, Select S3 and check “ListAllMyBuckets” and “ListBucket”. When you select “ListBucket”, it would automatically prompt you to select “All” or “Specific.” In this case, we would select “Specific” and then select “Add Arns”. Then we would input the name of the S3 bucket we created in step 1. So should look something like this:



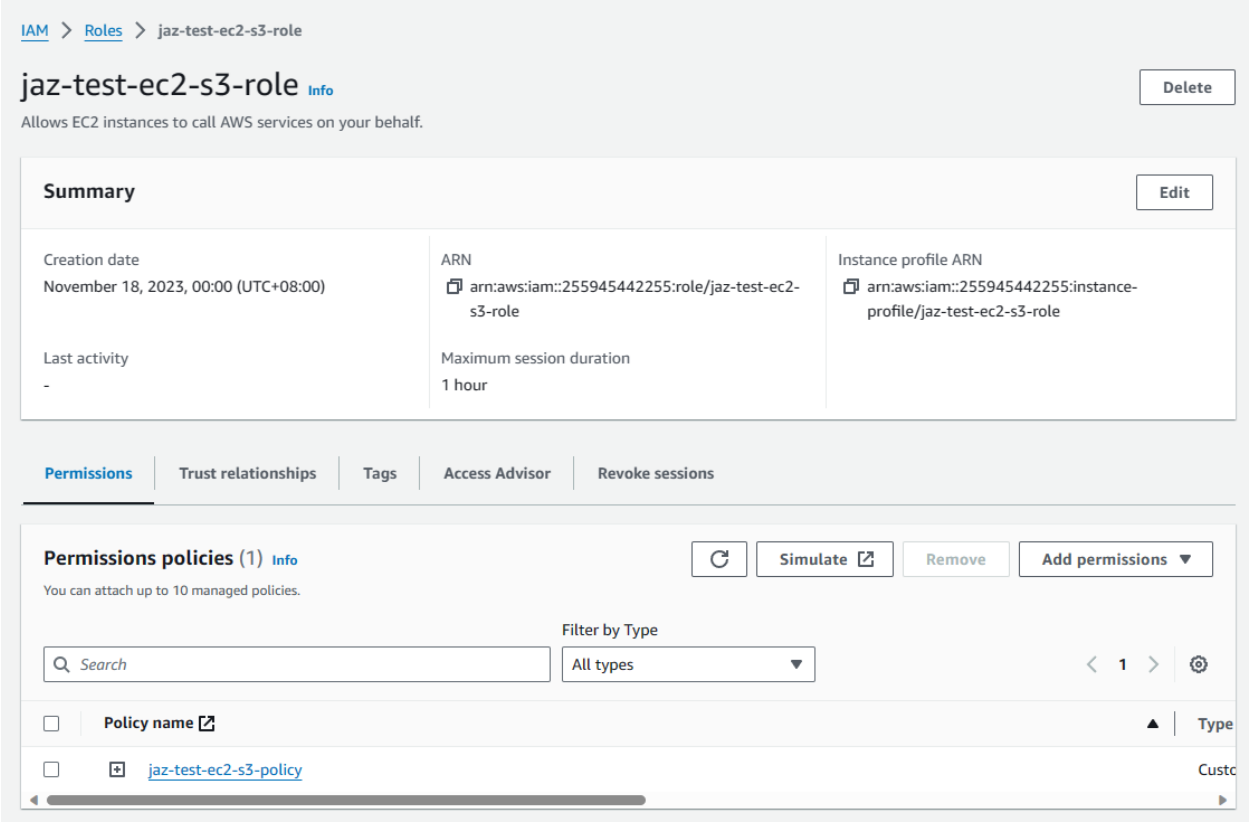
6. Click on next and give your IAM policy a meaningful name and then click on create policy. Example of my policy below:



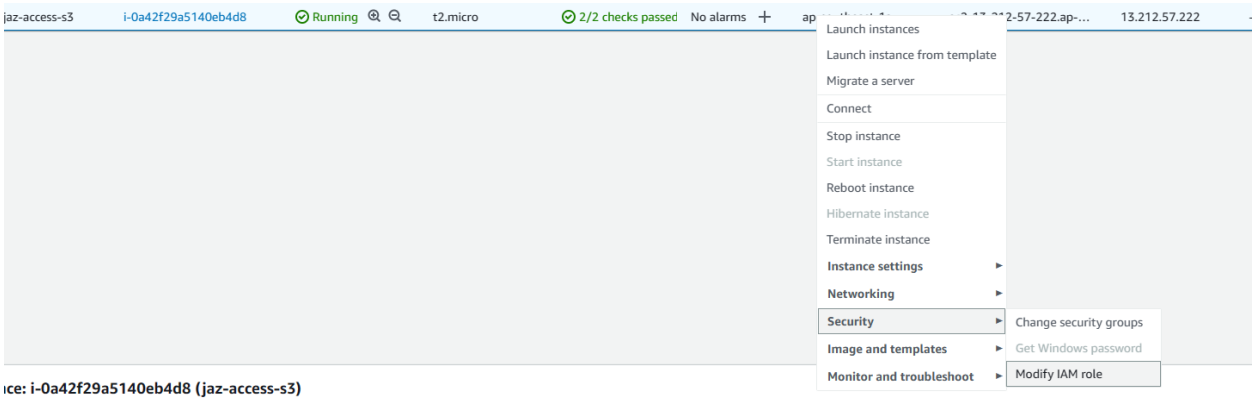
7. Click on Roles on the left and select “Create Role”
8. For service or use case, Select “EC2” and then click next
9. Look for the policy you created in the search tab provided, and then select your policy



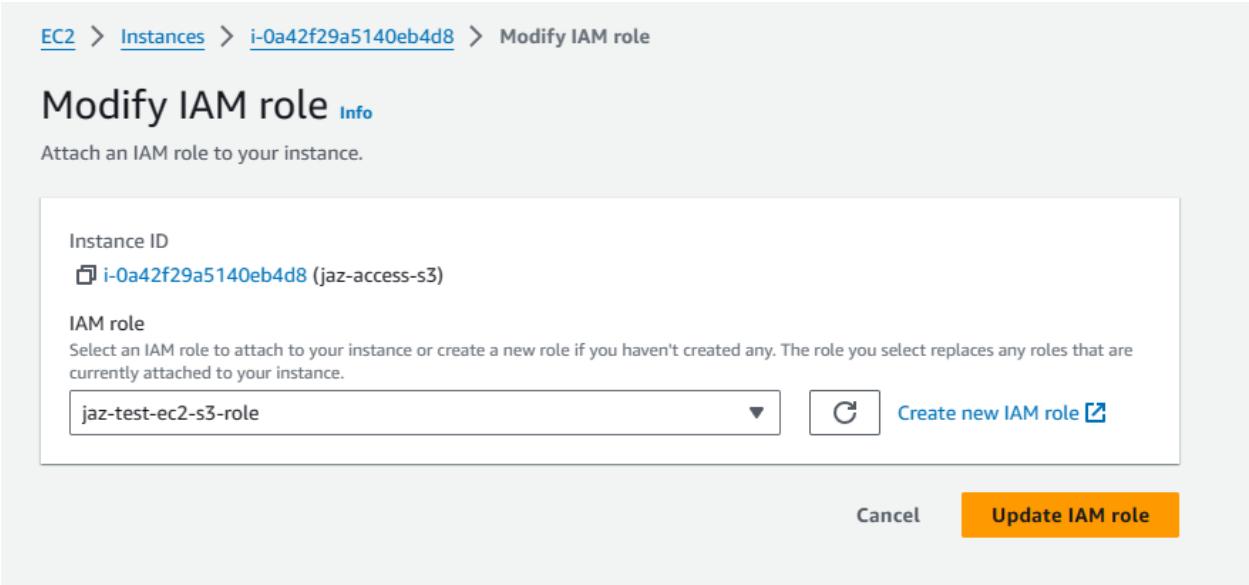
10. Click on next, then give your role a meaningful and unique name as well. You should have something like this:



11. Now we just need to attach this IAM role to the EC2 you created
12. Look for the EC2 you created and right click on it. Select Security -> Modify IAM Role



13. Look for the IAM role that you created and then select “Update IAM role”



14. Now we just need to verify the permissions are right.
15. Connect to your EC2 instance to run the following commands:
    a. aws s3 ls
    b. aws s3 ls s3://name-of-the-bucket-you-created-in-step-1
    c. aws s3 ls s3://other-s3-buckets-in-the-account

```
[ec2-user@ip-10-0-101-183 ~]$ aws s3 ls s3://jaz-test-iam-bucket
2023-11-17 15:07:41          0 testfile1.txt
2023-11-17 15:07:41          0 testfile2.txt
[ec2-user@ip-10-0-101-183 ~]$ aws s3 ls s3://hangytong

An error occurred (AccessDenied) when calling the ListObjectsV2 operation: Access Denied
[ec2-user@ip-10-0-101-183 ~]$
```