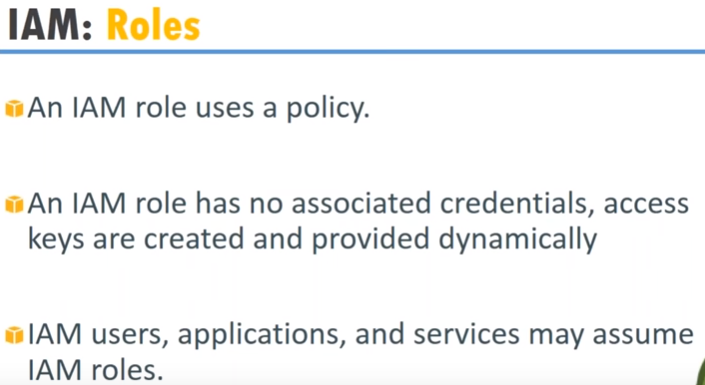
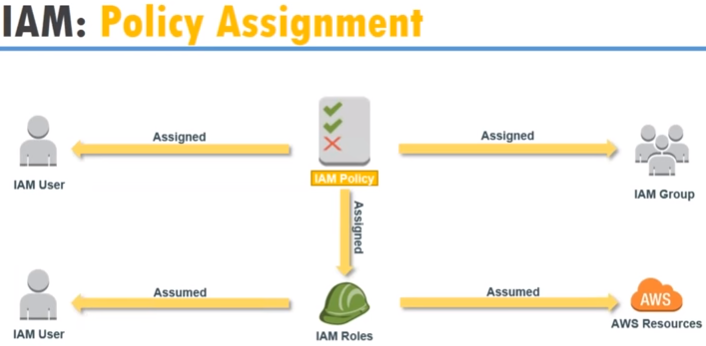
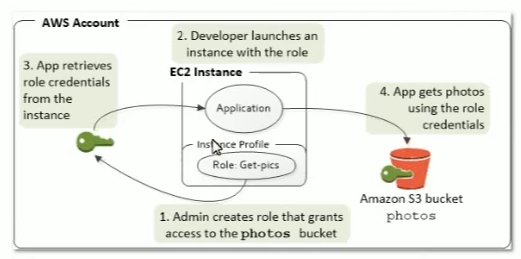
**Roles:**

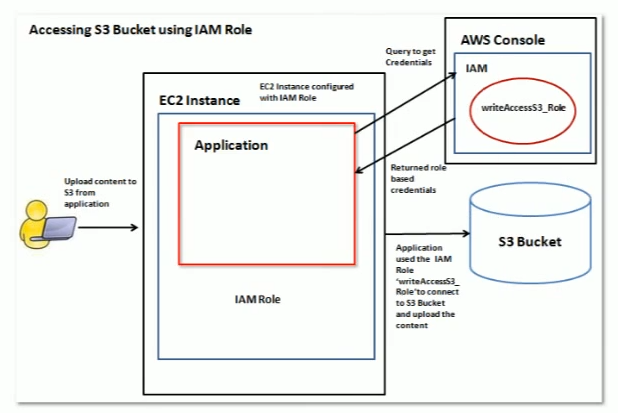
* Roles, we can use to give permissions for particular AWS service
* For example, we have app server and we want to store the files from server to s3
* The developer can use the access key to do the same. But that access key can create a problem due to in security
* So, we create a role as ec2 instance can call S3 API
* We can attach policies to that like s3 full access
* We cannot move the files simply from instance to S3 bucket. We need to have authentication
* We have to place the access key and secret key id into the application which is hosted on ec2 instance and allow s3
* But it is also risky, if someone gets access to their credentials, they can mis utilize
* So, here roles will be useful, we can apply roles to ec2 instance so that the application in instance can also be applied with the role
* Based on that role application will have access to the s3 without exposing access key and secret it



* IAM role doesn’t have any credentials
* It uses temporary keys. Like if ec2 wants to communicate with database, IAM will create temporary keys and will get expired after some time. These things taken care in background



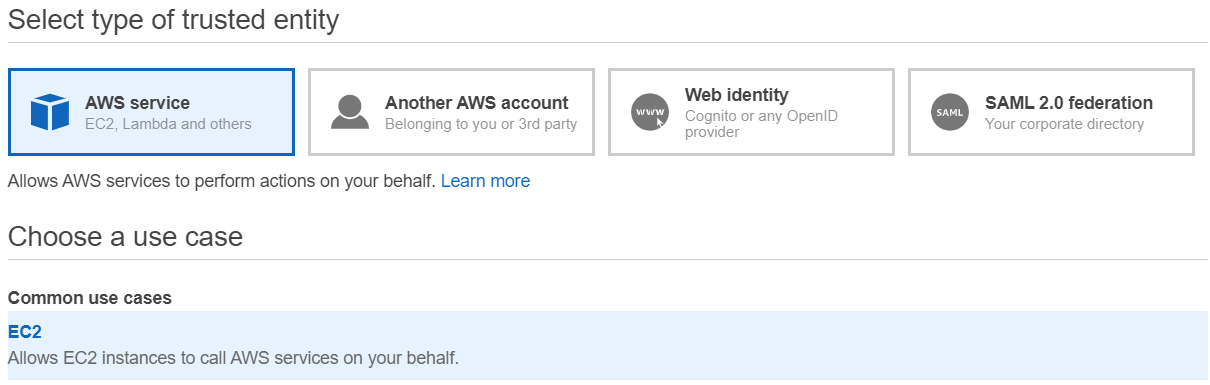




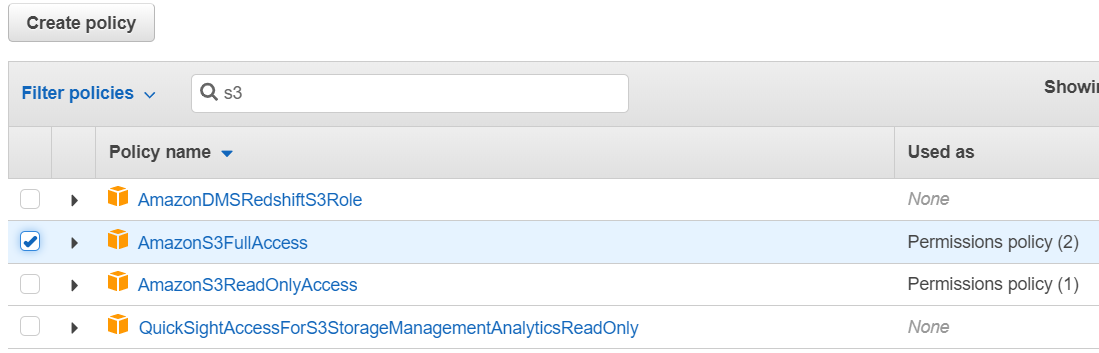
* We have policies to assign on users, groups, roles and resources

**Create roles:**

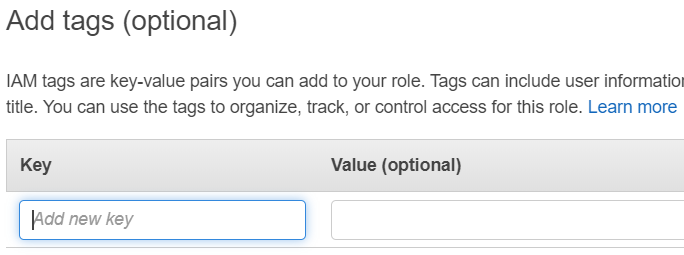
* Create a role by selecting the EC2 service to call on behalf as below



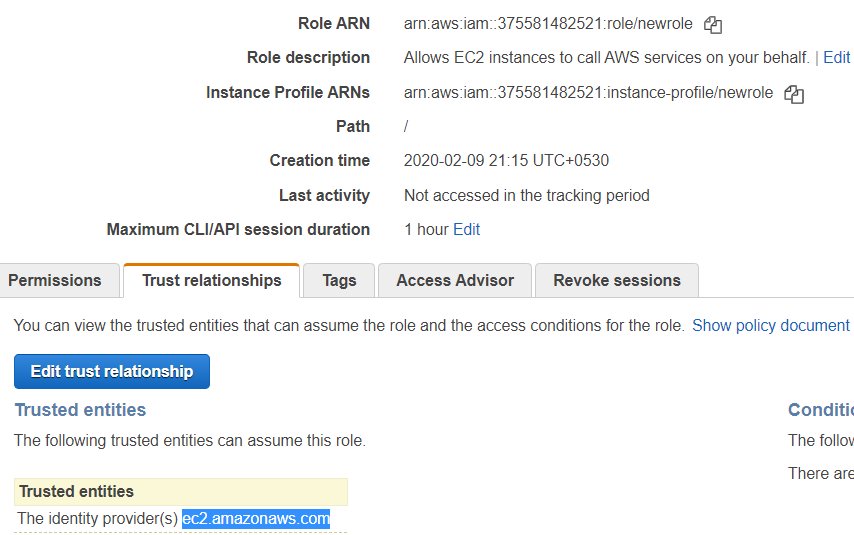
* Select the policy in next step



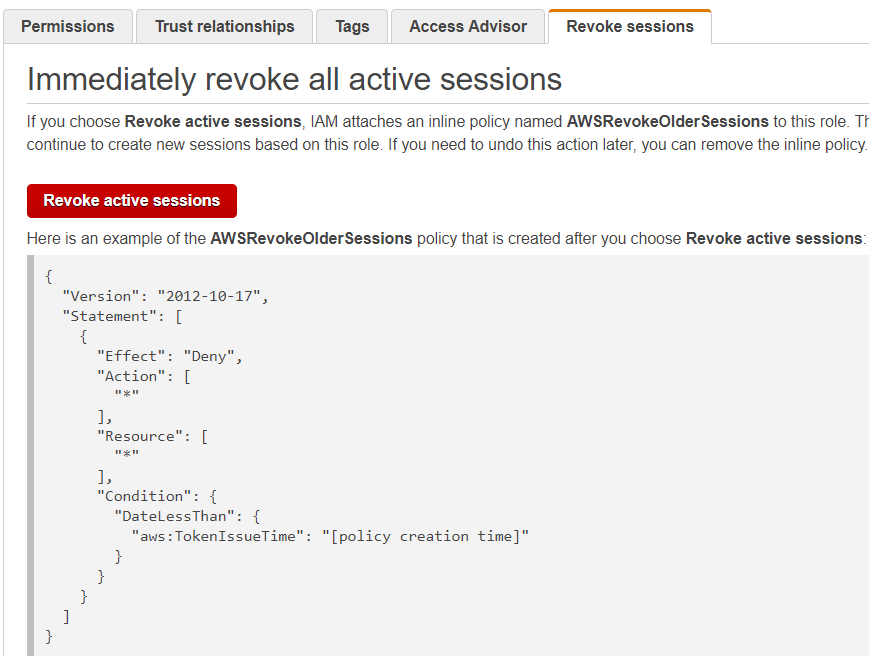
We can add any tags in step if we want to



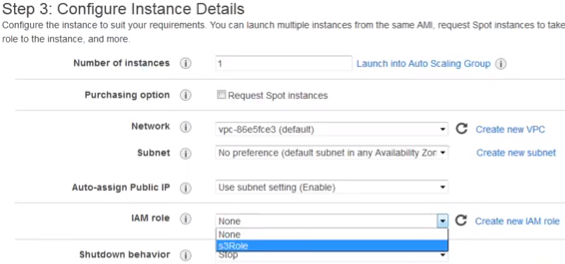
Once the role is created, we can see the trust relationships of the role as below



* We can revoke the active sessions if needed
* We can attach or detach the policies to the roles



**Attach a role:**



* We can attach the role while creating the ec2 instance as above
* We cannot attach later on

Now create an ec2 server, login to the server and then run the below command to check whether ec2 can access s3 or not. This we can do without using aws configure command and use keys

* **Aws s3 ls 🡪 to check the buckets in s3**
* **Aws s3 ls s3://bucketname 🡪 to list the objects in bucket**
* **Aws s3 cp s3://bucketname/objectname <path> 🡪 to copy the files from s3 to server**
* **Aws s3 mb s3://bucketname 🡪 to make the bucket**
* **Aws s3 rb s3://bucketname 🡪 to remove the bucket**

We can also Connect as programming user inside the server with the same region of s3. Then use s3 without roles

* Before executing the above commands, we need to make sure we have installed aws CLI.
* amazon Linux comes with aws cli by default. For others we need to install manually

