**Services we are going to use**

CodeCommit

Codedeploy

IAM

GIT

ROLES

CodePipeline

**Pre- Requisitre**

1. Git installed
2. IAM user – with permission user name – user1

AWScodecommitfull access && AWScodepipelinefullaccess

Create security credentials for GIT - generate

### User – security credentials - HTTPS Git credentials for AWS CodeCommit

1. Service Role - 2 roles need to be created

Create IAM role for codedeploy

Role – codedeploy – select codedeploy again –permission – select codedeploy access- done

Name -democodedeployrole

Create Role for EC2 to access S3

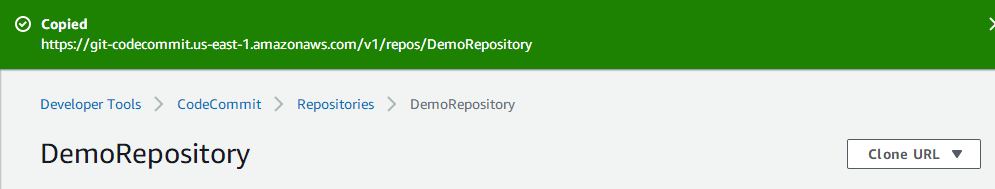
Role – Ec2 – EC2 – S3Readonly permission

Name –DemoEC2role

1. CodeCommit

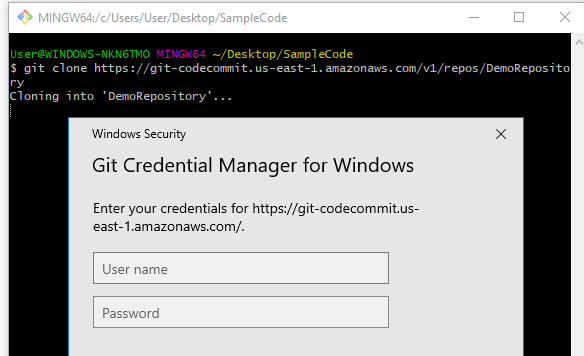
Create one repository – name it Demorepository

Now clone this repository to local machine



Copy the above url , go to local machine and open gitbash

Below it will ask for user ID and password, here you need to add gitcredentials ( generated during IAM user creation )



Download code from github

Next to update this code to codecommit

Once you have code uploaded to codecommit – go to next step

Launch EC2 Instance

Linux – security group – make sure to open port 80

Attach EC2 role ( created above )

Install codedeploy agent on linux machine

sudo yum update

sudo yum install ruby

sudo yum install wget

cd /home/ec2-user

wget https://bucket-name.s3.amazonaws.com/latest/install

(<https://docs.aws.amazon.com/codedeploy/latest/userguide/resource-kit.html#resource-kit-bucket-names> )

chmod +x ./install

sudo ./install auto

sudo service codedeploy-agent status

sudo service codedeploy-agent start

sudo service codedeploy-agent status

Now go to codedeploy

Services – codedeploy – Create application – name your application – select platform as EC2 – Create application

Next to create – deployment group – name your deployment group – choose service role – select inplacedeployment – select EC2 – select machine – disabled loadbalancer – save

Next step is to create CodePipeline

Go to services – codepipeline –

Create pipeline – name your pipeline – select new service role –select allow aws codepipeline to create a service – default S3 location –

Next – select provider – codecommit – select repository name – select amazon cloudwatch Events – next

Skip build stage

Next deploy provider – select aws codedeploy – select region – select application – select deploymentgroup – next – review and create the pipeline

Once pipeline is created – go to EC2 webpage and check for deployment

Now modify your code and check for pipeline

Go to source code – make some changes – upload that changes to codecommit repository