**Deploy Docker app to AWS using ECS**



* Above diagram is the overall workflow
* Prerequisite: AWS CLI needs to install and create a user in AWS with below permission.

|  |
| --- |
|  |
|  |  |
|  | ECR |
|  | { |
|  | "Version": "2012-10-17", |
|  | "Statement": [ |
|  | { |
|  | "Sid": "VisualEditor0", |
|  | "Effect": "Allow", |
|  | "Action": "ecr:\*", |
|  | "Resource": "\*" |
|  | }, |
|  | { |
|  | "Sid": "VisualEditor01", |
|  | "Effect": "Allow", |
|  | "Action": "ecr:GetAuthorizationToken", |
|  | "Resource": "\*" |
|  | }, |
|  | { |
|  | "Sid": "VisualEditor02", |
|  | "Effect": "Allow", |
|  | "Action": "ecr:InitiateLayerUpload", |
|  | "Resource": "\*" |
|  | } |
|  | ] |
|  | } |

* Create a docker file and build it and push it to ECR
* Steps to create docker file : [https://docker-curriculum.com/](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbEhaZHdHMW5VQ2ZwVUZydG1QR21qWFp4WWkyd3xBQ3Jtc0trdVpzcTJBWlF1YWR2NF85OFJPb25MSTkzX2MyUTEzMWpBM0xaQkRtWjlKTFc1N1RNNFpEakYyWi1mZDdRam1PR2ZnVnZEVzNSUW9nZUxQSTZDRVRib3R4T0tSYXFKTm1VVl82NXQyQUJrRFREd29EVQ&q=https%3A%2F%2Fdocker-curriculum.com%2F&v=zs3tyVgiBQQ)
* Commands for above steps:

|  |  |
| --- | --- |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
| 1.BuildDockerImage |
|  | docker build -t test . | |
|  |  | |
|  | 2. Run container /w image | |
|  | docker run -d --publish 8888:5000 test | |
|  |  | |
|  | 3. Login to ECR | |
|  | aws ecr get-login-password --region REGIONHERE!!!! | docker login --username AWS --password-stdin ACCOUNTIDHERE!!!!.dkr.ecr.REGIONHERE!!!.amazonaws.com | |
|  |  | |
|  | 4. Tag the version | |
|  | docker tag test:latest YOURACCOUNT.dkr.ecr.YOURREGION-1.amazonaws.com/YOURREPO:YOURTAG | |
|  |  | |
|  | 5. Upload | |
|  | Docker push YOURACCOUNT.dkr.ecr.YOURREGION.amazonaws.com/YOURREPO:YOURTAG | |

* Now our container is uploaded into ECR
* Now Go to AWS ECS service
* Create a cluster
* When we create a cluster it automatically create an EC2 instance ,This EC2 instance help to deploy our containers inside it
* Now go to task definition -> Create a new task definition ->Select EC2 ->Configure task definition details -> open add container tab -> Give a container name -> copy and paste ECR repositor ARN(in which we pushed our image) in image section -> Set port mapping with required port details
* Now go to tasks option ->Run new task ->Select launch type as EC2 ->Select task detention we created in previous step -> click run task
* Now the image which is available in AWS ECR is deployed to the EC2 instance we configured
* Now we can access our application from the browser using EC2\_Public\_IPV4\_DNS:Port\_number .