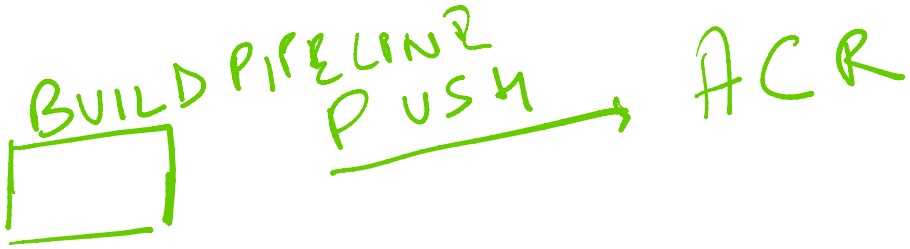
Day 8 AKS + PULL REQUEST + IAAC [Theory]

Topics + Demos

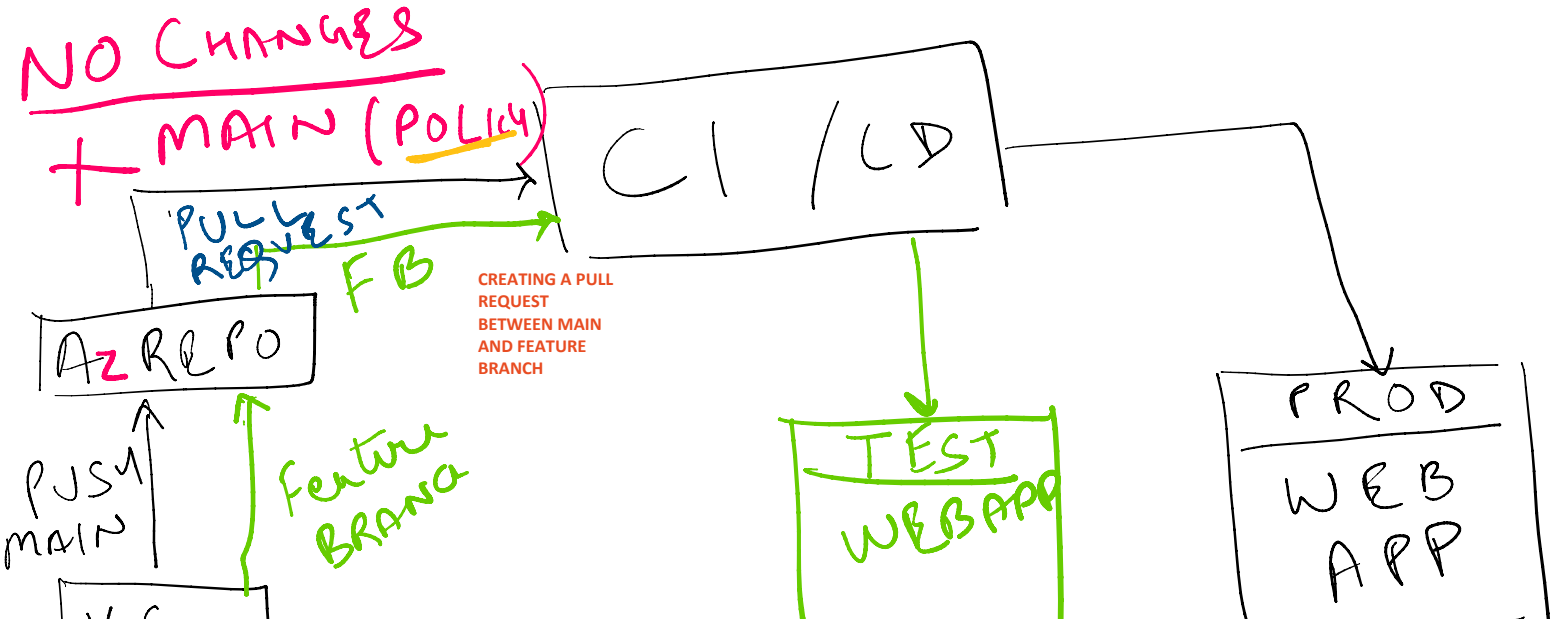
1. **Demo - Manual Deployment Of AKS**
   - I already have a Container Registry -
   - Already have a AKS cluster.
   - Pushed image to the Azure container registry from local machine using Visual Studio.
   - Added the workload manually.
   - Added the load balancer manually.
   - Tested the app in AKS and it worked fine.

2. **Demo - CI/CD AKS**
   - Push the code from visual studio to Azure Repo in a new repository.,
   - Add manifest file to your app code source file.
   - Manifest file is going to be the same YAML file that we saw to automate the deployment of our app and load balancer.
   - Created a build pipeline using the docker build and push task.
   - Tested the build pipeline and worked.
   - We then added the publish pipeline task in the Build pipeline.



Demo - Pull Request + Merging Branches in Enterprise World

1. In real world we do not make changes to the main branch.
2. The main branch is only touched once during the initial deployment and then after we add policies /security to it protect from any abrupt change.
3. We will go back to my .dot8 app.
4. I will also use the azure web apps.



Technical Debt

- There are multiple developers working in orgs.
- All of them keep on making changes/writing new code.
- There is a high chance of exposing your code/app. Getting into vulnerabilities.
- Sonarqube - Tasks to be added to the pipeline
- Sonarcloud - Cloud version

Demo - Sonarcloud/SonarQube

a. Limited

IAAC - Infrastructure As A Code

- Previously I had to create all my resources before hand so that I can deploy my code.
- But is this feasible ?
  - Not feasible.
  - Not Resourceful.
  - Cost gets increased.
- In your release pipeline itself you want to make sure you deploy your environment first.
  - Followed up by building your code on that env.
- There are 4 options we have for IAAC
  - **ARM Template** [Microsoft] - Declare code and define your infrastructure.
    - **Bicep [Newer version of ARM template]**
  - **Terraform -** Define the infra and deploy it
    - Ansible
  - **Azure CLI**
  - **Azure Powershell**

ARM TEMPLATE [BICEP]

- You define your infrastructure.
- You user Azure Resource Manager Template.
- .json fromat.
- You can store your ARM in your source code or any storage of your choice and link it with your release pipeline.