Jenkins-Day-1

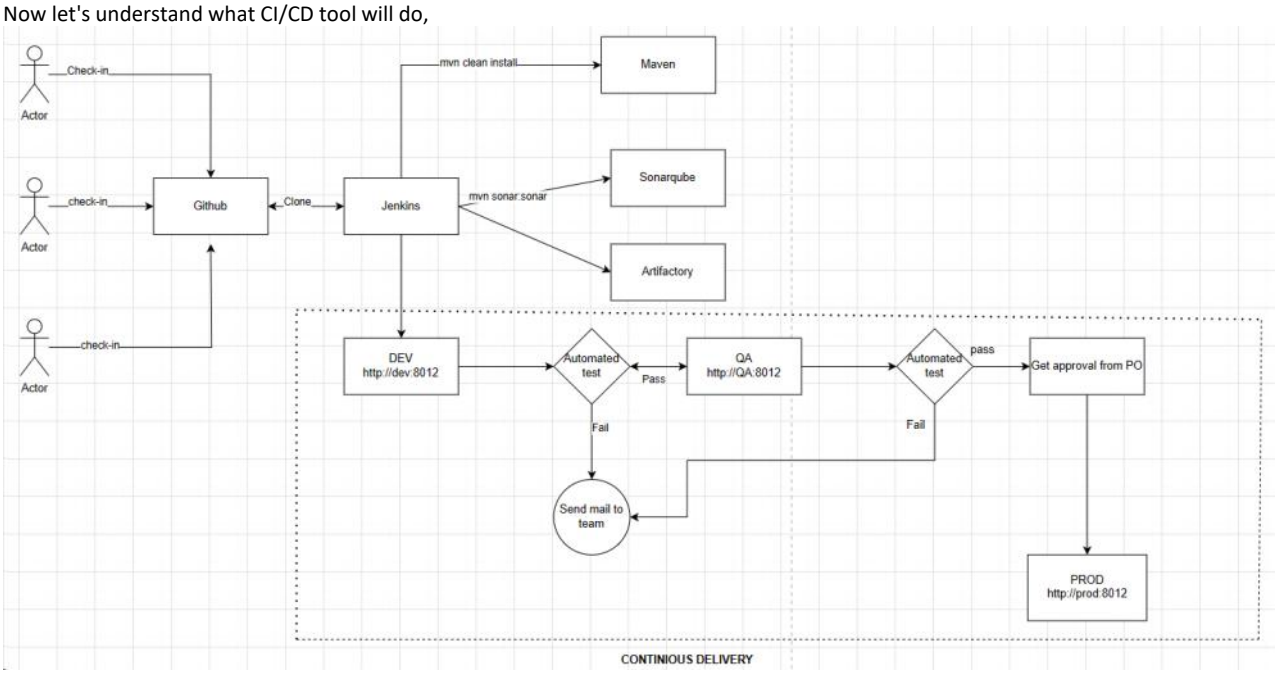
01 December 2023 18:39

- Today onwards we are going to start discussion on the Jenkins tool.

- Before we are going to understand
  - **what is Jenkins tool?**
  - **Why do we need Jenkins tool?**
  - **How to install Jenkins on AWS Linux EC2 instance**

- Jenkins is a **CI/CD** tool**.**
  - **CI** - Continuous Integration
  - **CD** - Continuous Delivery (or) Continuous Deployment

- Now let's understand what CI/CD tool will do,



  - Assume there are 3 developers are working in a project.
    These three developers normally will develop the code for the project in their laptops, correct or not? - Yes
    After code development completed these developers will keep the code in common place, correct?
    What is that common place? GitHub

  - Now assume this is a Jenkins.

  - What are all other tools we have covered so far?
    I have discussed
    - build tool as a Maven
    - Static code analysis tool as a sonarqube

  - Once application development code completed & you generated packages like .jar/.war/.ear file, will you deploy it to client e nvironment directly? - No
    Why? There will be a chances for issues/bugs in that code & so whenever we deploy that .jar/.war/.ear there will be a chances of application will not work correctly.

  - So before we deploy application into client environment first we do test in DEV & QA environment, after the regression test c ompleted successfully we do deployments in PROD environments.

    Let's assume this is
    - DEV environment & It's automated testing setup
    - QA environment & It's automated testing setup
    - PROD environment & prod-checkouts

  - Now all these tools are integrated with Jenkins
    - GitHub
    - Maven
    - Sonarqube
    - DEV/QA/PROD servers

- So whenever there is a
  - Code change in a GitHub
  - Jenkins will download that code and inform maven to run build command mvn clean install & will generate packages .jar/.war/.ear
  - Once the packages are generated next Jenkins will inform sonarqube to run static analysis & publish generated report into son arqube dashboard.
  - Now packages are deployed into DEV servers automatically with help of Jenkins.
    - For each environment there will be a endpoint URL like **https:// <dev-server-ip>:8012**
    - Next QA engineers will do end to end testing in DEV environment.
    - Once the testing completed successfully in DEV environment QA will green signal to proceed to deploy on QA environment.
  - Now Jenkins will do deployment to QA servers,
    - Now QA environment will have one more endpoint URL like **https:// <qa-server-ip>:8012**
    - Again QA engineer will do end to end test manually in QA environment.
    - Now we see there is no issues & ready to take prod & we have to take approval from product-owner to deploy application in to PROD.
  - After product-owner approval application will get deployed into PROD environment with help of Jenkins.

► **Continuous Integration** ==> Once developer check-in code into GitHub & on that code
  - Automated build
  - Automated unit-test
  - Automate code analysis
  - Artifact will stored

► **Continuous Delivery** ==> It's a process to automate deployment in DEV & QA environments & automate testing as well for those two environments(non-prod), but we need approval of application owner to do production deployment.

► **Continuous Deployment** ==> It's a process to automate deployment in DEV & QA environments & automate testing as well, & no need approval from application owner to do production deployment.

==========

► **How to install Jenkins in EC2**

==========

Go to the official site: https://jenkins.io

► **Pre-requisites**
- EC2 instance: Amazon-Linux
- Allow port 8080
- Install Java-11
  yum install java-11-openjdk-devel -y

► **Installation steps**
- **Add the Jenkins repo to download the packages:**
  sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
- **Import key file to authenticate the Jenkins repo in-order to install the Jenkins package**
  sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
- Install Jenkins
  yum install jenkins -y
- Start jenkins server
  systemctl start jenkins
- To auto restart jenkins on server reboot
  systemctl enable jenkins

► **Configure Jenkins**
- By default jenkins runs on port 8080
- We can access jenkins using URL http://<ip_address>:8080
- Default user name for jenkins is **admin**
- Password for admin stored by default at **/var/lib/jenkins/secrets/initialAdminPassword**
- Do the installation of recommended plugins

- Change password of admin user

► Next I will show you how to create jenkins job, configure it & execute it.

- Click on new-item to create job
- We have enter the name of the job in the text box.
- There are different kind of job types are present to create.
  - Free-style
  - Pipeline
  - Folder
  - Maven
    - **Freestyle:**
      - This is the most common & basic job type in Jenkins.
      - Using this job type we can process for build, test & deploy processes with various configurations & settings options in GUI way.
    - **Pipeline:**
      - Using this option we define the process for build, test & deploy with code either in scripted type/declarative type.
    - **Folder:**
      - This option used to organize the jobs, I mean that if you have 50 jenkins-jobs, out of these 25-related to Team-A & other 25-related to Team-B. So we can create two folders like Team-A & Team-b, we can move the Team-A related jobs to Team-A folder & Team-B related jobs to Team-B folder.
    - **Maven:**
      - This option is mainly designed for build, test & deploy process for Java based applications.
- Let's choose the freestyle job option & continue.
  - Here what are the different sections to define process for build, test deploy.
    - General
    - Source code management
    - Build Trigger
    - Build
    - Post-build
  - First we go with basic configurations,
    - In General section under **description** we can write **the purpose of the job.(Like: This is my first jenkins freestyle-job)**
    - Under the build section choose execute sh script & enter command to print message **"This is my first jenkins job"**
      - **echo "This is my first jenkins job"**
      - Why did I entered echo command? because jenkins present running on Linux node.
      - **Wherever we created the jobs in jenkins, that node we can refer as master node.**
    - **Save:** If you click on save button you remain in the same page & configuration are applied.
    - **Apply:** When we click on apply if any configurations those will get applied & will go back jenkins main page.
  - Now Execute the job --> See the output of job execution --> **"This is my first jenkins job" got printed**
  - This job triggered for only one time so far based on job history.
  - When you click on execute one more time & we can find the job execution number.
- Hope now you got idea how to create, configure & execute the job.

==========

► **Home-work**

==========

- **Create freestyle Jenkins job to print the date**

- **Job name:** date-job
- **Description:** To print the date of Jenkins EC2 server
- **Execute shell:** echo “The date & time is: ” `date`