CONTINUOUS INTEGRATION

DevOps Basics

What is Continuous Integration?

Continuous Integration (CI) allows the developer and operations teams to roll out the managed builds and deployments efficiently/automatically. The build process can be triggered automatically or manually whenever the developers have made significant changes to the source code. This reduces the effort on both teams and automation also reduces the error rate.

The following components are there in this phase:

- Jenkins
- GitLab (to store code)
- Test Server (to deploy the application/web app)

People involved: Developers and Operation team

What is a Continuous Integration Server?

A continuous integration server is an automation server that coordinates/runs different stages/tools of the DevOps process.

Examples: Jenkins, GoCD

External sources

- What is Continuous Integration? https://www.edureka.co/blog/continuous-integration/
- What is Jenkins? https://www.jenkins.io/
- Jenkins tutorials: https://www.tutorialspoint.com/jenkins/index.htm

Why is it important in DevSecOps?

The Continuous Integration server is responsible for triggering the DevOps pipeline and coordinate the artifact handling. In the case of DevSecops, the CI server triggers the security tools in different phases (e.g. Static Code Analysis, Dynamic Analysis, etc) and stores the logs/reports.

What will you learn in this section?

The user will learn to perform the following tasks

- Building a Java Web App from source code using Jenkins
- Building Nginx Webserver from source code using Jenkins
- Run pre-deployment checks on Django Web App source code using Jenkins

Tools Covered

- Jenkins

Labs

- Java Webapp
  - Jenkins and Gitlab instances are provided. The code is present on Gitlab. The user has to configure the Jenkins server to build the project from source code, package it into JAR archive using Maven.
    Objective: Create a Jenkins job to package the web application into a deployable WAR package!
- Django Webapp

Objective: Create a Jenkins job to migrate and test the web application!

- Nginx Software
  - Jenkins and Gitlab instances are provided. The code is present on Gitlab. The user has to configure the Jenkins server to build the Nginx binary using make.
    Objective: Create a Jenkins job to configure and build the Nginx binary!

Java Webapp

Start

Django Webapp

Start

Nginx Software

Start