Jenkins is an automation tool written in Java with built-in plugins for continuous integration tasks. It is used to continuously build and test projects making it easier to integrate the changing codes to it.

Jenkins Originally Developed by Sun Micro Systems in 2004 under name **Hudson.**

The Project was later named Jenkins when oracle bought Microsystems

Jenkins Installation:

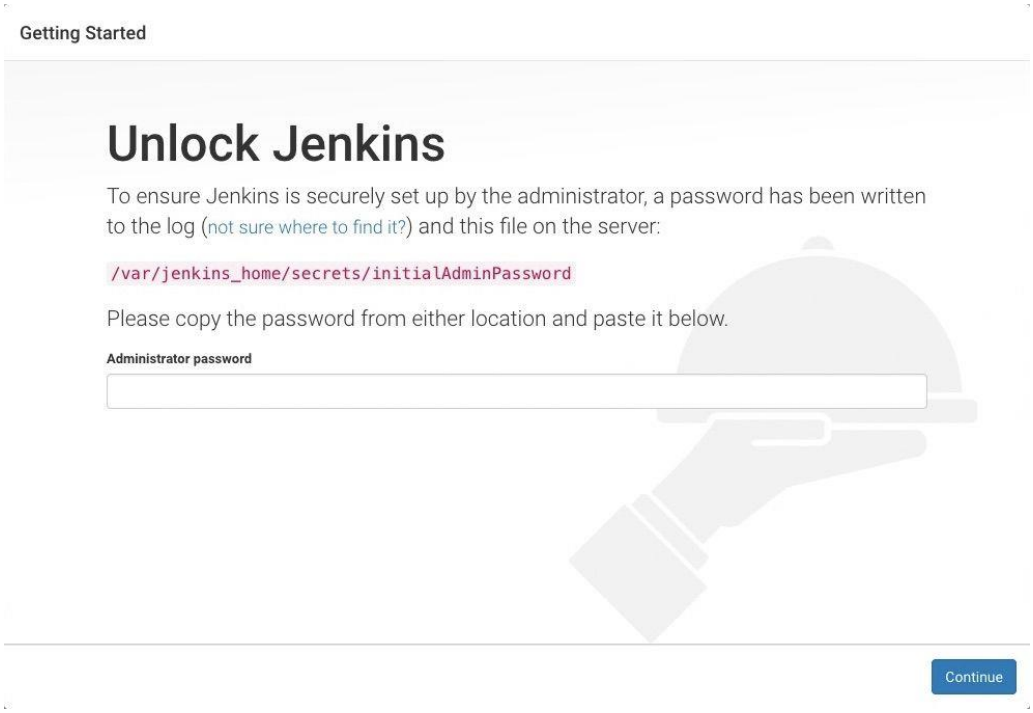
- Install Jdk 17
- sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
- sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
- yum install Jenkins

- You can enable the Jenkins service to start at boot with the command
- sudo systemctl enable Jenkins

- You can start the Jenkins service with the command:
- sudo systemctl start Jenkins

- You can check the status of the Jenkins service using the command:
- sudo systemctl status jenkins
-

Browse **to http://localhost:8080** (or whichever port you configured for Jenkins when installing it) and wait until the **Unlock Jenkins** page appears.

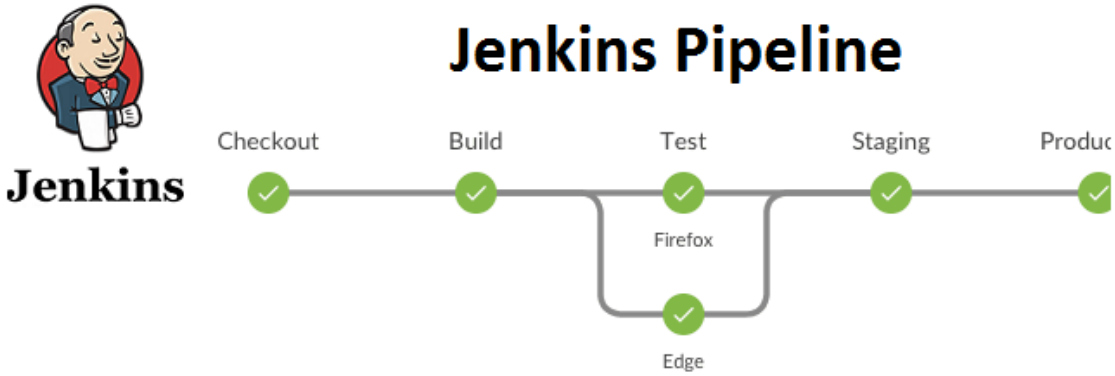


- The command**: sudo cat /var/lib/jenkins/secrets/initialAdminPassword** will print the password at console.

Other Platforms than Jenkins:

- GitLab CI/CD
- Travis CI
- CircleCI
- Bamboo

Jenkins WorkFlow:



Continuous Delivery

A software development discipline where software is built so that it CAN be released to production at any time.

Continuous Integration

A software development practice where contributors are integrating their work very frequently. This results in multiple daily integrations to a mainline. Automated testing (post-commit promotion) is commonly used.

PipeLine:

Jenkins Pipeline is a suite of plugins which supports implementing and integrating continuous delivery pipelines into Jenkins. Pipeline provides an extensible set of tools for modeling simple-tocomplex delivery pipelines "as code"

CI/CD Pipeline

A typical CI/CD pipeline includes the following stages:

1. **Source Stage**: Code is committed to the version control system (e.g., Git).
2. **Build Stage**: Code is compiled or packaged into executable files.
3. **Test Stage**: Automated tests are run to verify the functionality and correctness of the code.
4. **Deploy Stage**: The code is deployed to a staging or production environment.
5. **Monitor Stage**: The deployed application is monitored for performance and errors.