

**AUTOMATION PROJECT USING JENKINS**

**Jenkins** is an open-source automation server that helps automate various aspects of software development, including building, testing, and deploying software. It is a widely used tool in the field of continuous integration and continuous delivery (CI/CD).

With Jenkins, you can create automated workflows called "jobs" or "builds" that define the steps and actions required to build and deploy your software projects. These jobs can be triggered manually or automatically based on certain events, such as code changes in a version control system.

Some key features of Jenkins include:

1. Continuous Integration (CI): Jenkins allows developers to integrate their code changes frequently, ensuring that the software builds successfully and that any issues are detected early in the development process.
2. Extensibility: Jenkins has a large number of plugins available, which allow you to extend its functionality and integrate with various tools and technologies. These plugins provide support for things like version control systems, testing frameworks, deployment tools, and more.
3. Distributed Builds: Jenkins can distribute the build and test workloads across multiple machines, allowing for faster and more efficient execution of jobs.
4. Build Pipelines: Jenkins supports the creation of complex build pipelines, where multiple jobs can be connected together to form a sequence of steps. This enables advanced workflows and allows for more sophisticated deployment strategies.
5. Community and Ecosystem: Jenkins has a vibrant community of users and contributors, which means there is a wealth of resources, documentation, and support available. You can find a wide range of plugins and integrations developed by the community.

Jenkins can be installed on a server or run as a Docker container, and it provides a web-based interface for managing and configuring jobs. It supports various operating systems and integrates with popular version control systems like Git

Overall, Jenkins is a powerful tool for automating software development processes and improving the efficiency and quality of your software projects.

Jenkins port number is 8080.

Jenkins was introduced to overcome water fall model and agile methodologies (draw backs / manual effort) because of its key feature CI / CD.

we will integrate Jenkins with all devops tools with Jenkins and deploy the applications through environment (dev / Qa / uat / prod)

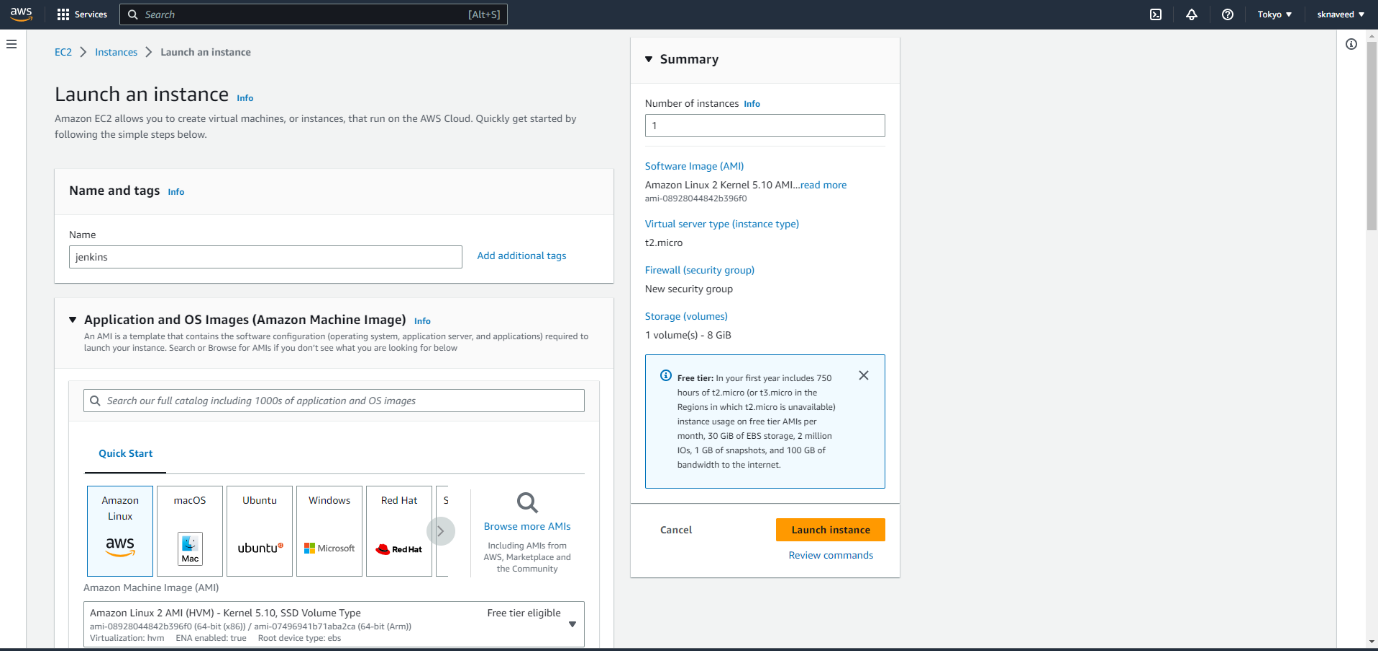
In Jenkins we create jobs to integrate and deploy the applications, there are 2 types of jobs.

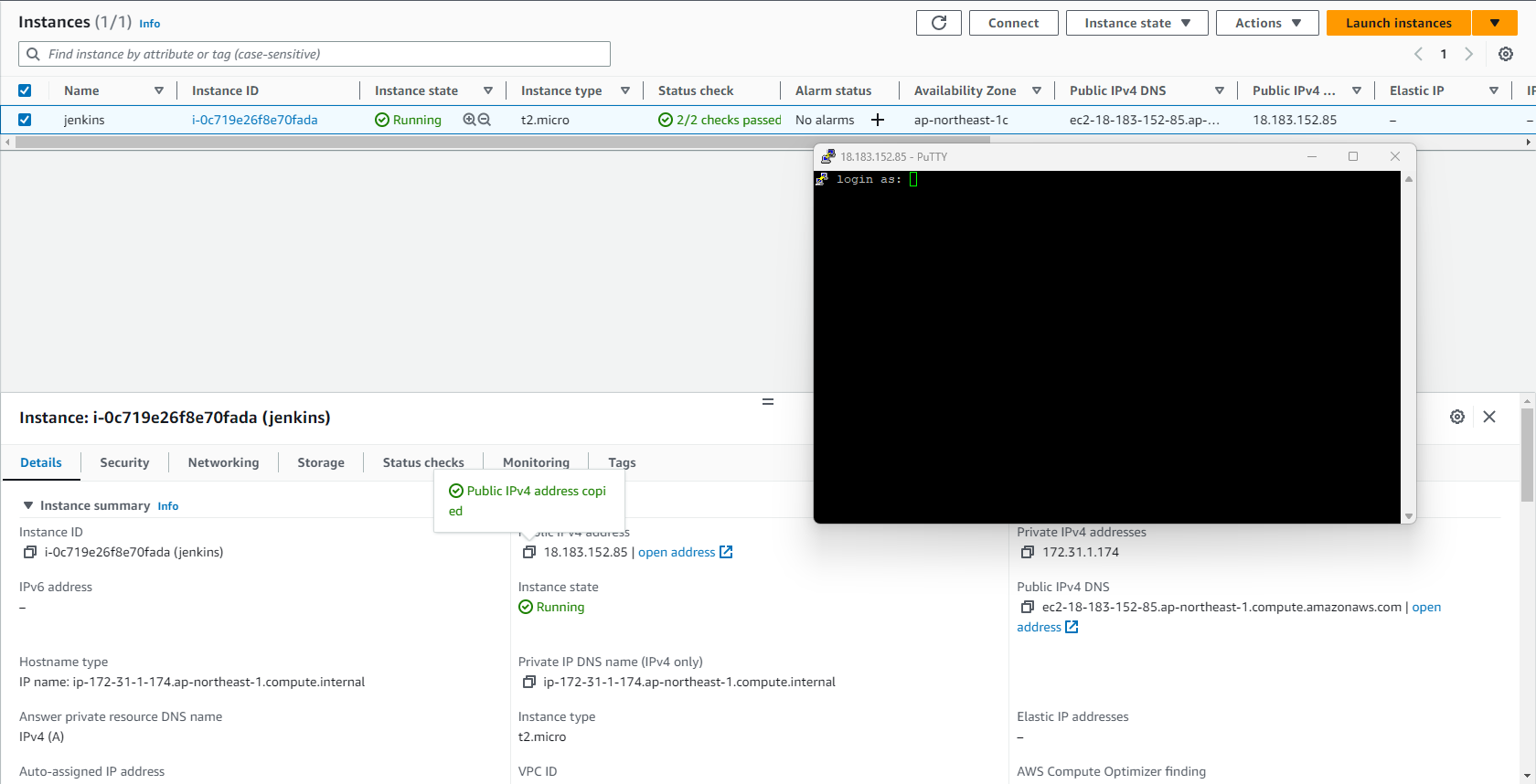
1. free style jobs (In Jenkins dash board GUI mode)

2. Pipeline jobs. (Jenkins dash board ===>>> Groovy script and GitHub ==>> Jenkins file)

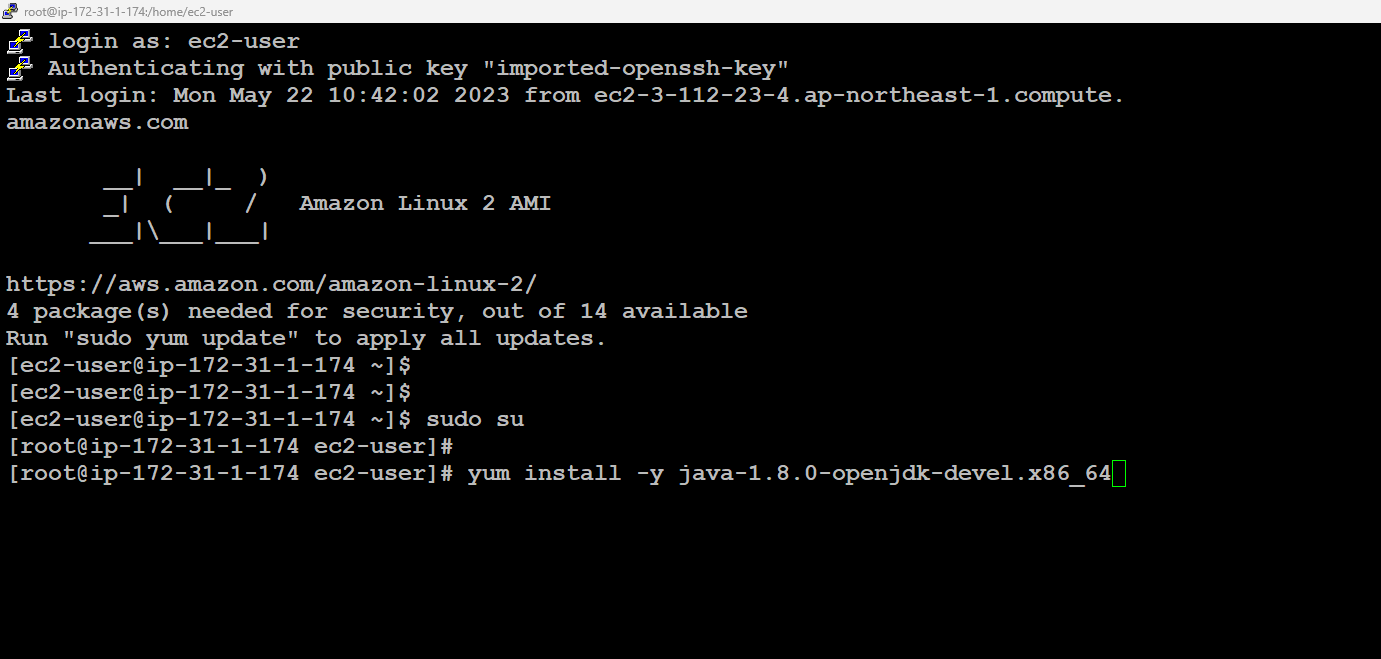
**Jenkins installation**

1. we need to take one Ec2 instance and login into that Ec2 instance.

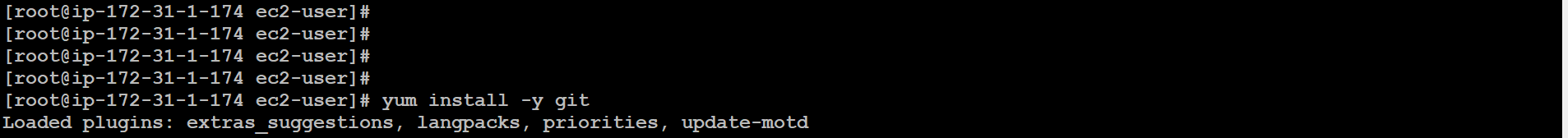




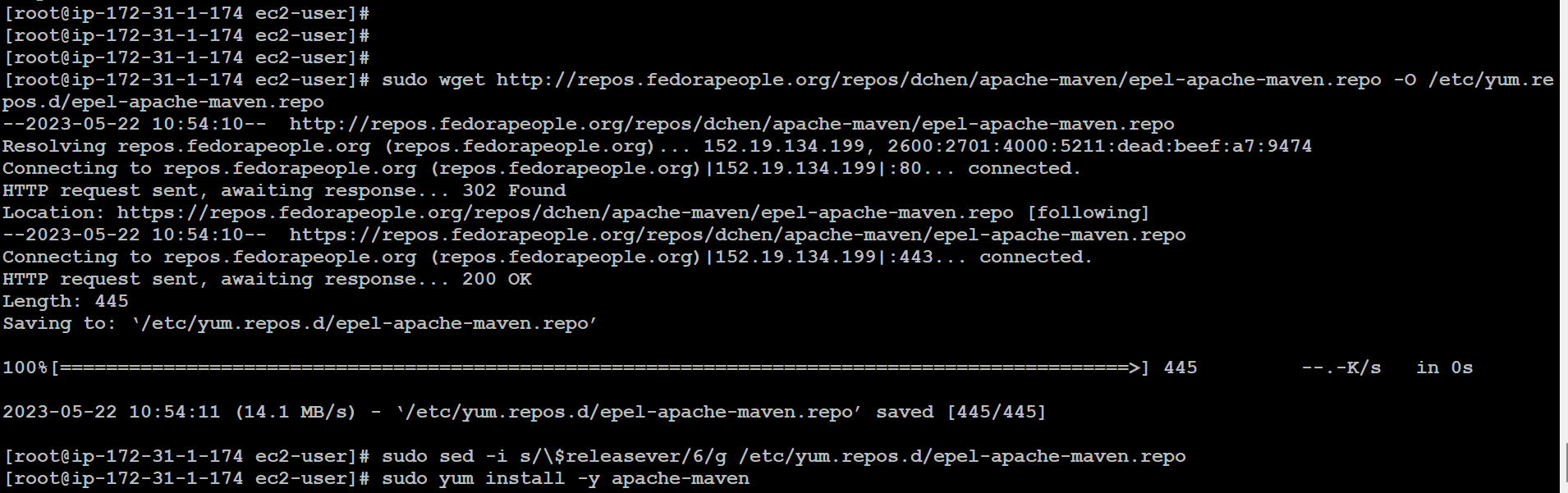
1. Now install java



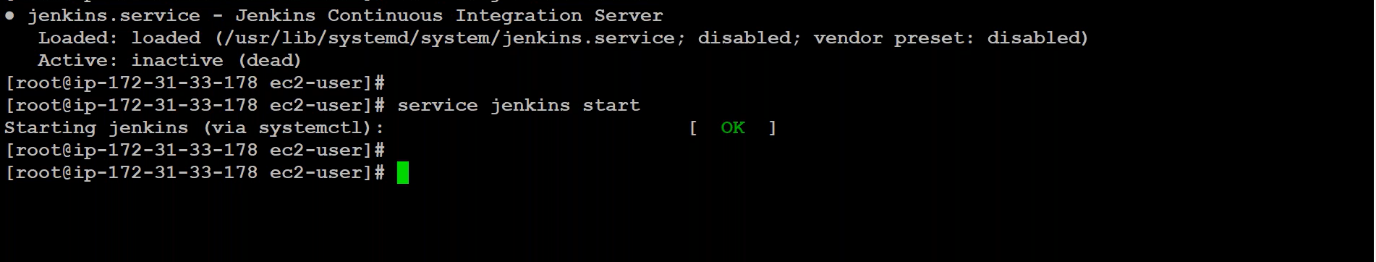
1. After that install Git



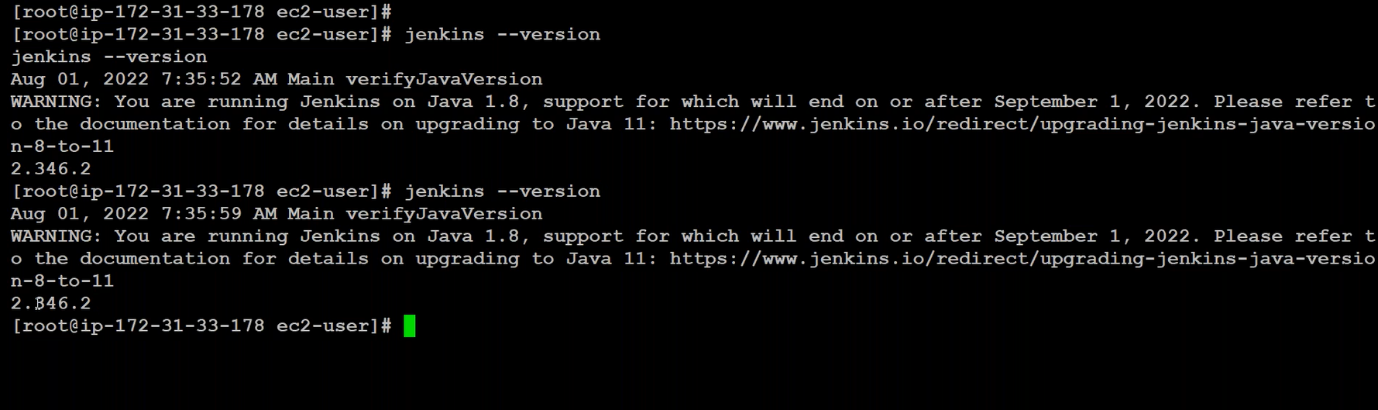
1. Install maven for build



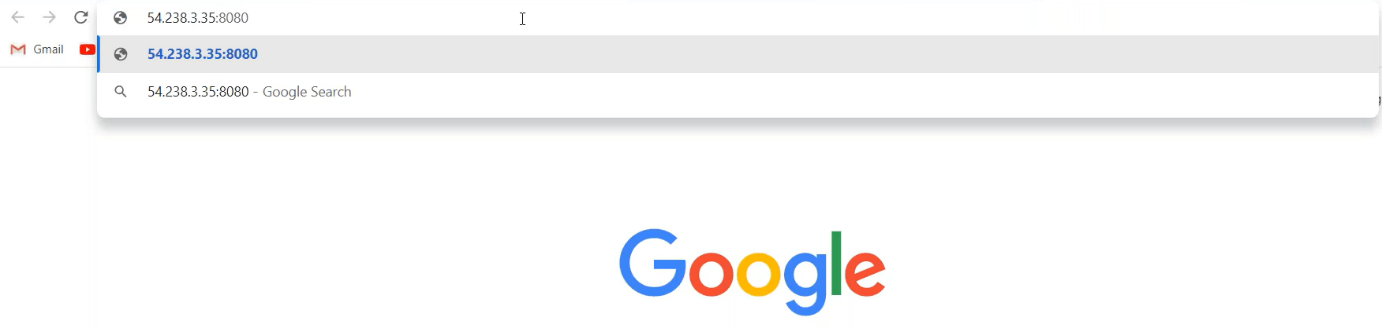
1. Install Jenkins in the same Ec2 instance



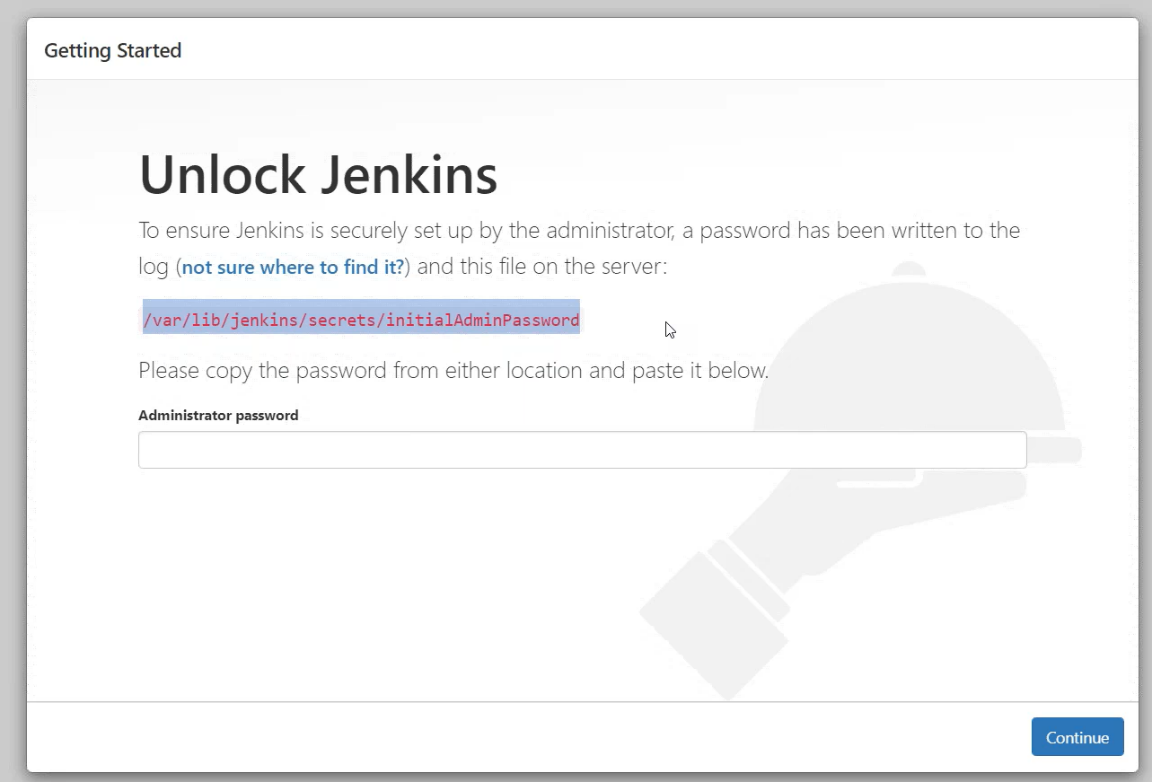
1. start the Jenkins service by typing ‘service Jenkins star’



1. In Jenkins ec2instance select the public IP and paste it in browser and type ‘:8080’ enter. The Jenkins dashboard will load.



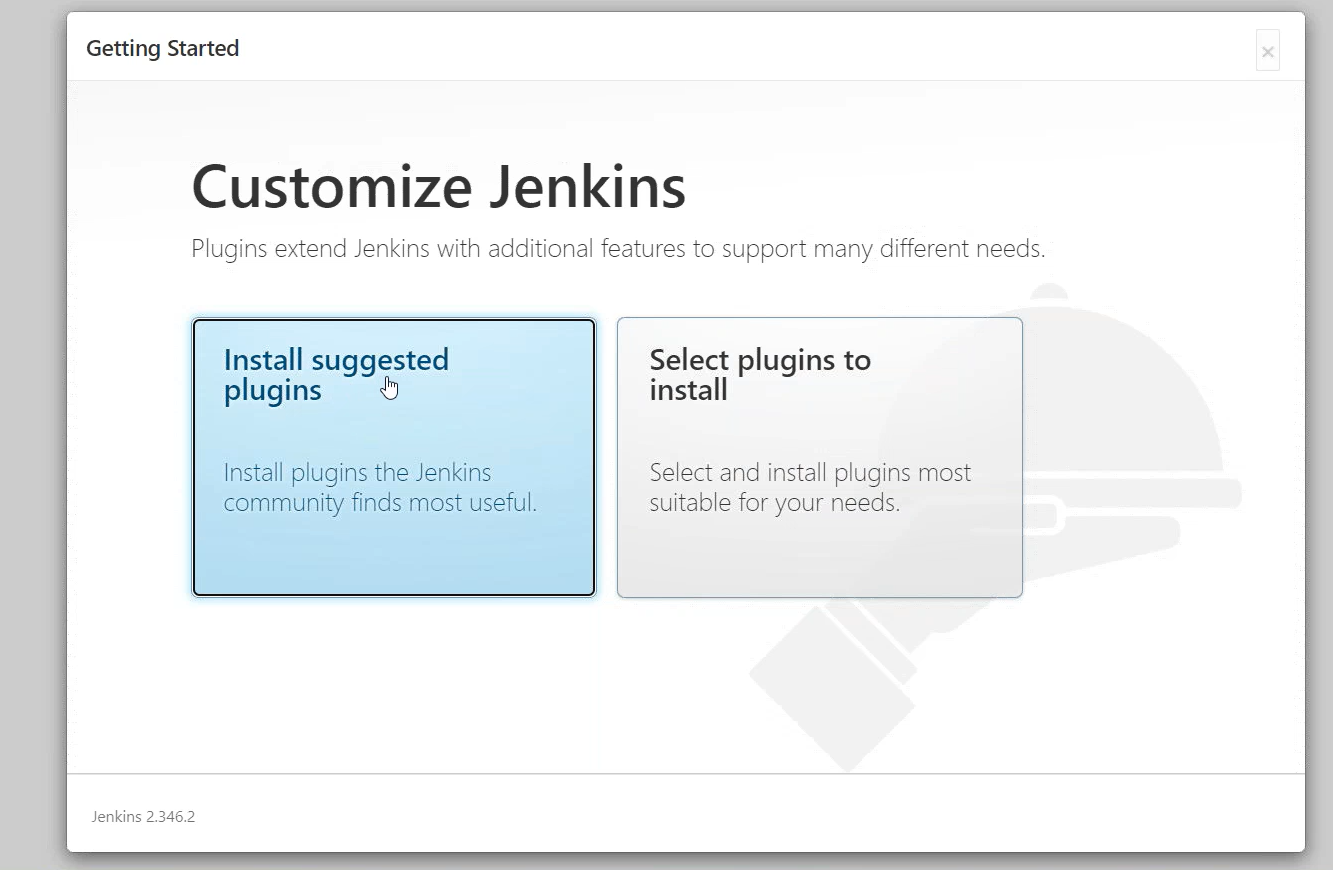
1. Jenkins dash board ==>> display ==>> path /var/lib/Jenkins/secrets/initialadminpasswd ===>>> copy this path.

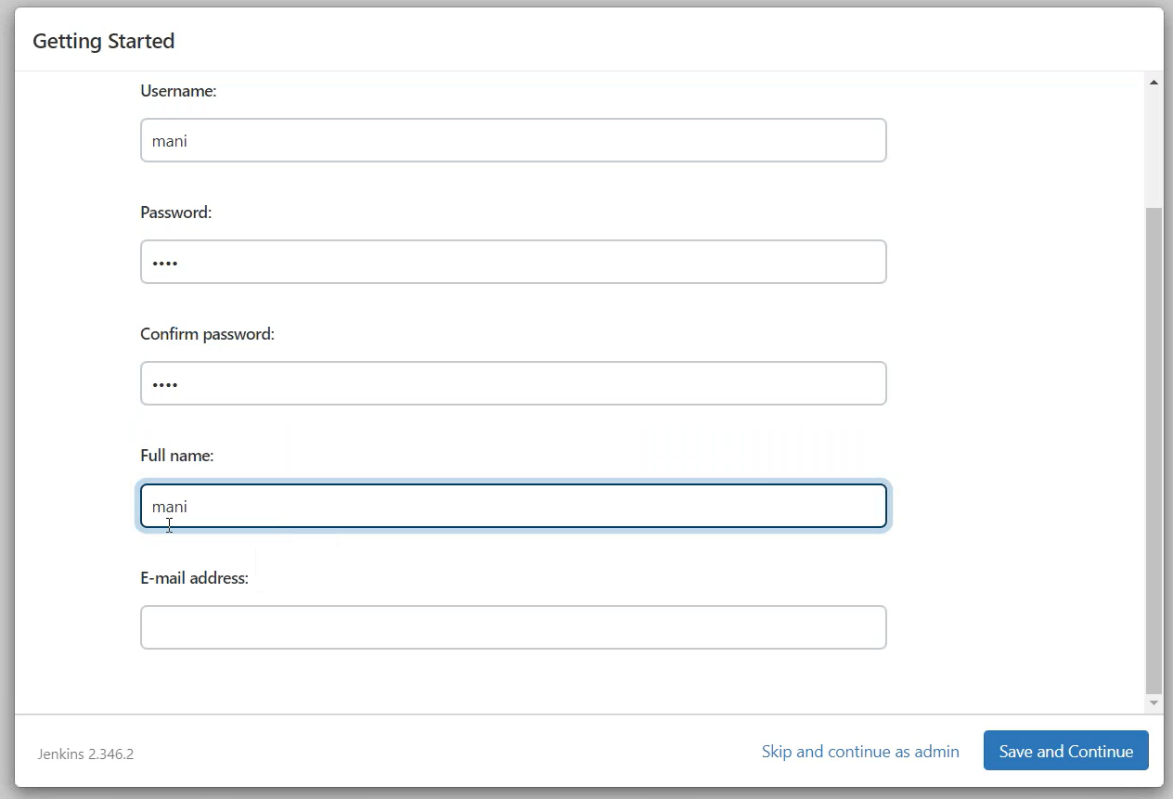
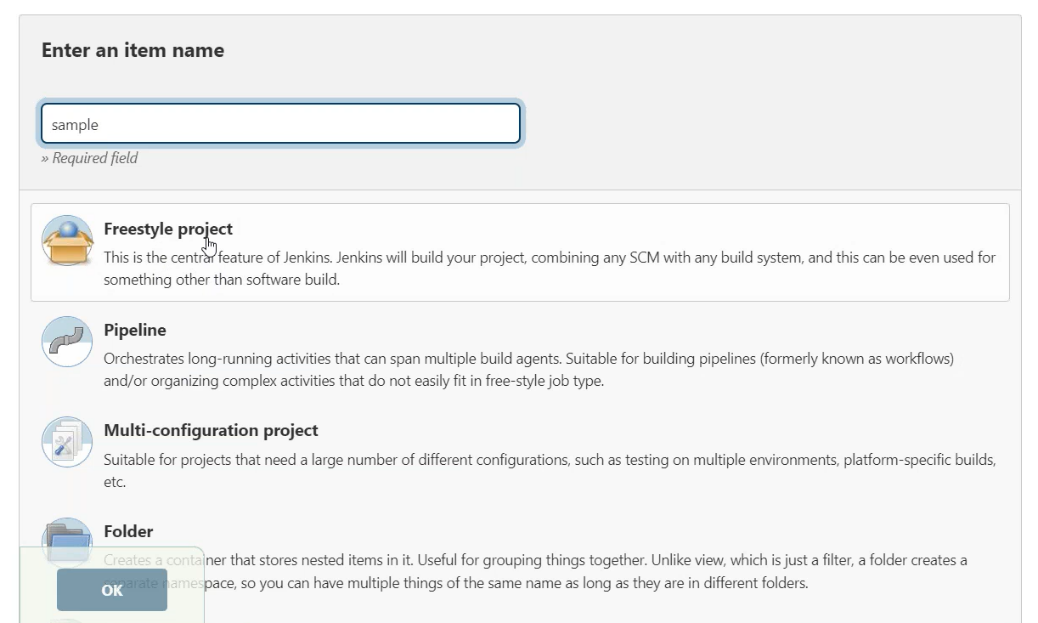


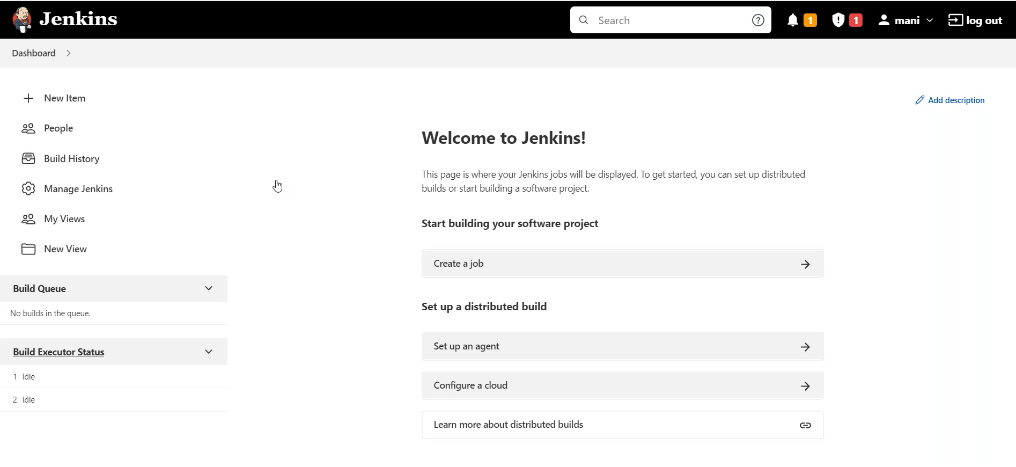
1. go to Jenkins ec2 instance ==>>> cat /var/lib/Jenkins/secrets/initialadminpasswd ==>> enter ===>> encrypted password ==>> copy paste it into Jenkins dashboard box.

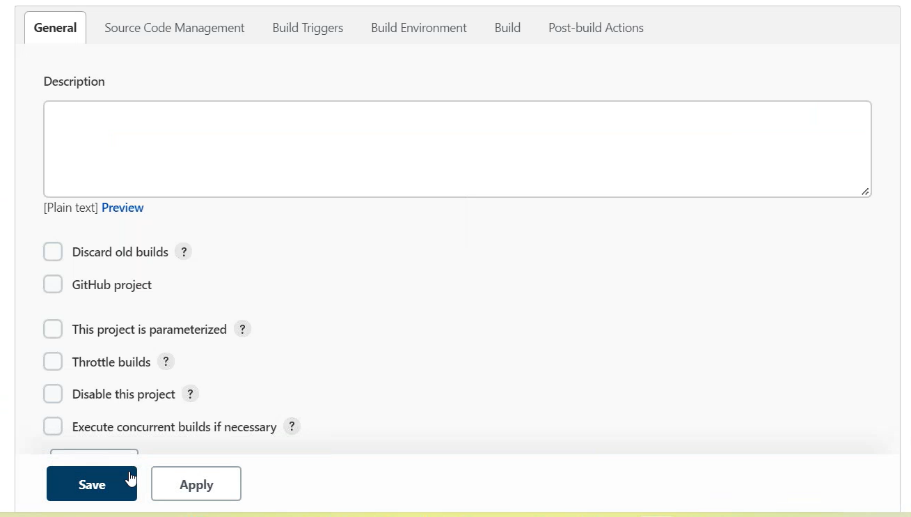


1. select suggested plugins and enter



1. Give the username, password, conform password, email and continue towards your Jenkins dashboard. 
2. Here is the Jenkins dashboard





DACWQAFUHUWEI WENLWEVNK WJENJKWN