**Enterprise java CI/CD with Jenkins,GitOps and kuberneted observability**

**Techonology used:**

**1. Jenkins:**

* Acts as the automation server.
* Pulls the code from GitHub.
* Runs Maven build and test phases.
* Connects with SonarQube to perform static analysis.
* Builds Docker image and pushes to Docker Hub.
* Updates K8s manifests.

**2. SonarQube:**

* Analyzes code quality and security vulnerabilities.
* Integrates with Jenkins using the withSonarQubeEnv and waitForQualityGate.

**3. Docker:**

* Packages the Spring Boot app into a container image.
* Image pushed to Docker Hub.

**4. Minikube:**

* Runs a local single-node Kubernetes cluster for deployment testing.
* Used to simulate real-world Kubernetes environments for learning and testing.

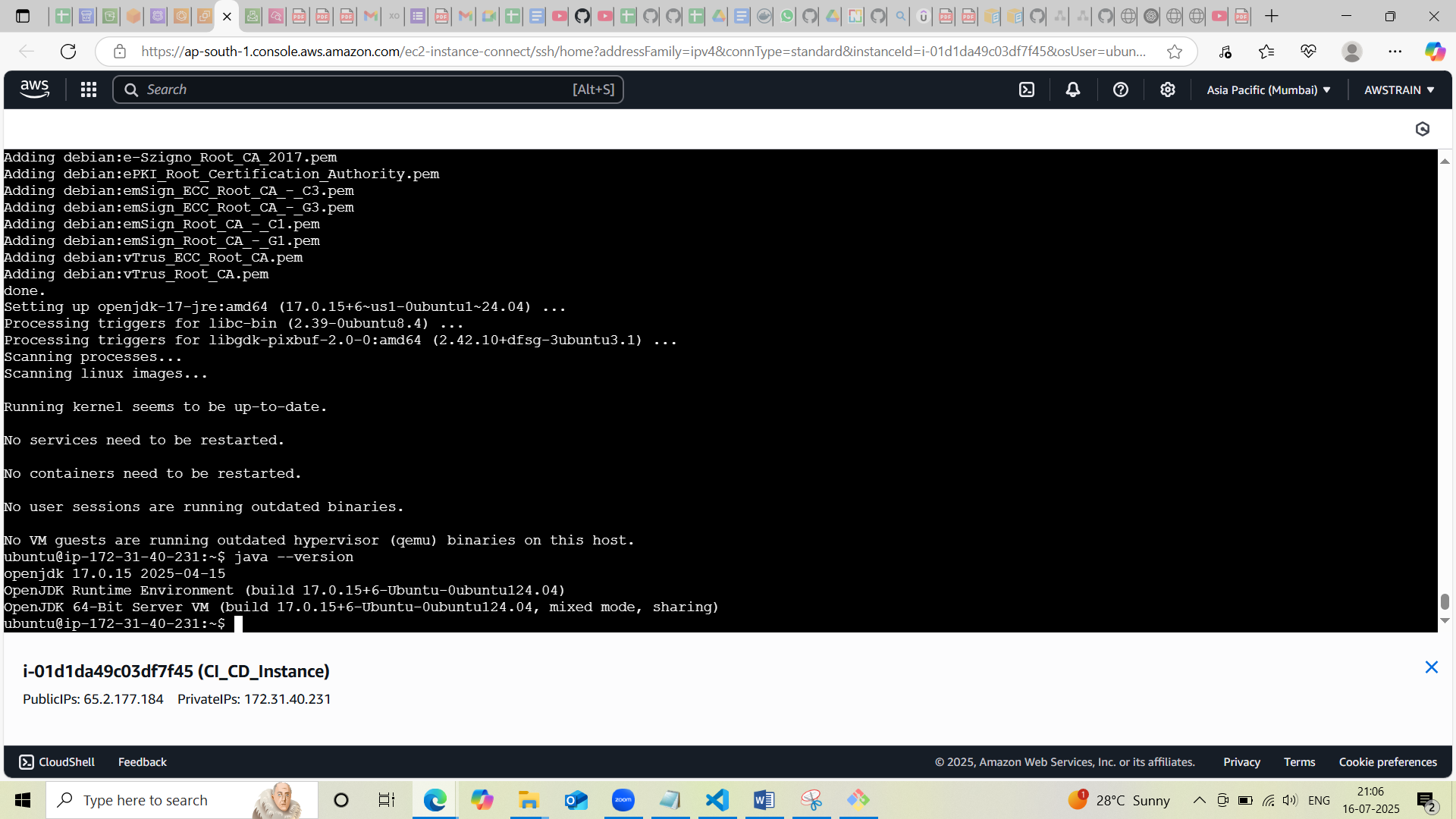
**5. Kubernetes:**

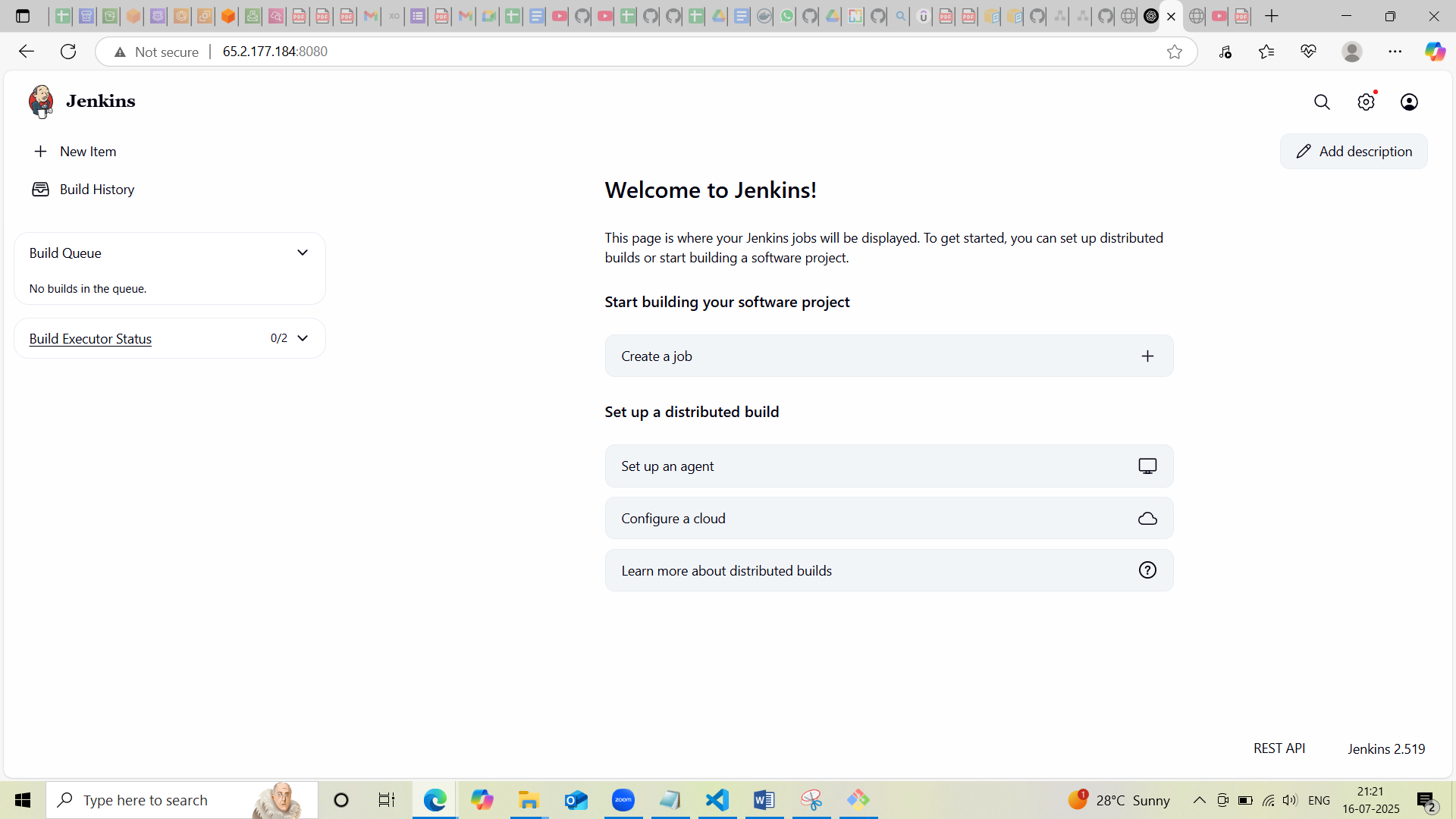
* Manages the container lifecycle.
* Uses deployment.yaml and service.yaml for deploying and exposing the application.

**6. Argo CD:**

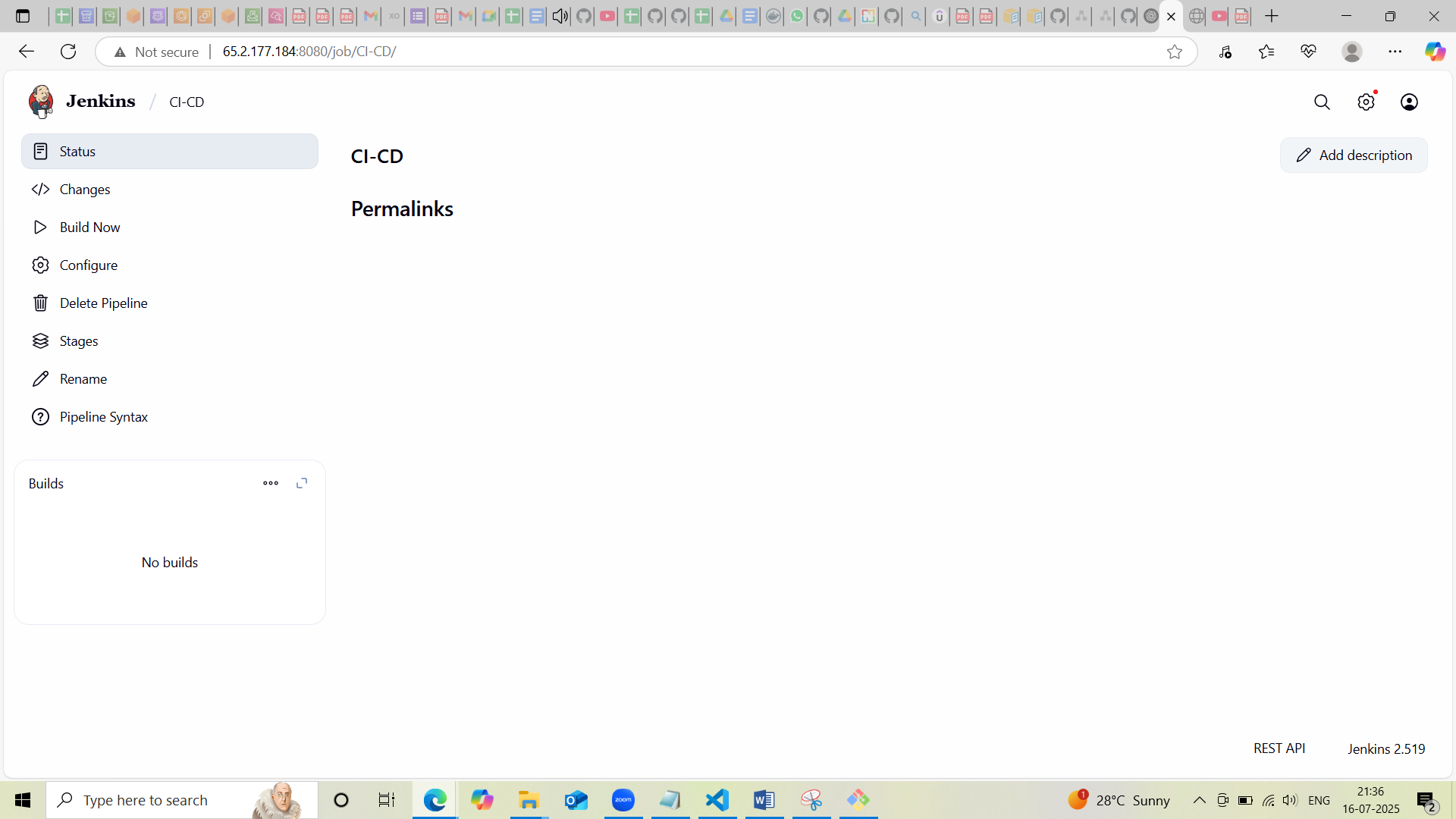
* Continuously watches the Git repo for manifest changes.
* Automatically syncs the new deployment to the Minikube cluster.
* Provides GitOps-based deployment and rollback capabilities.

**Installing Java and Jenkins:**

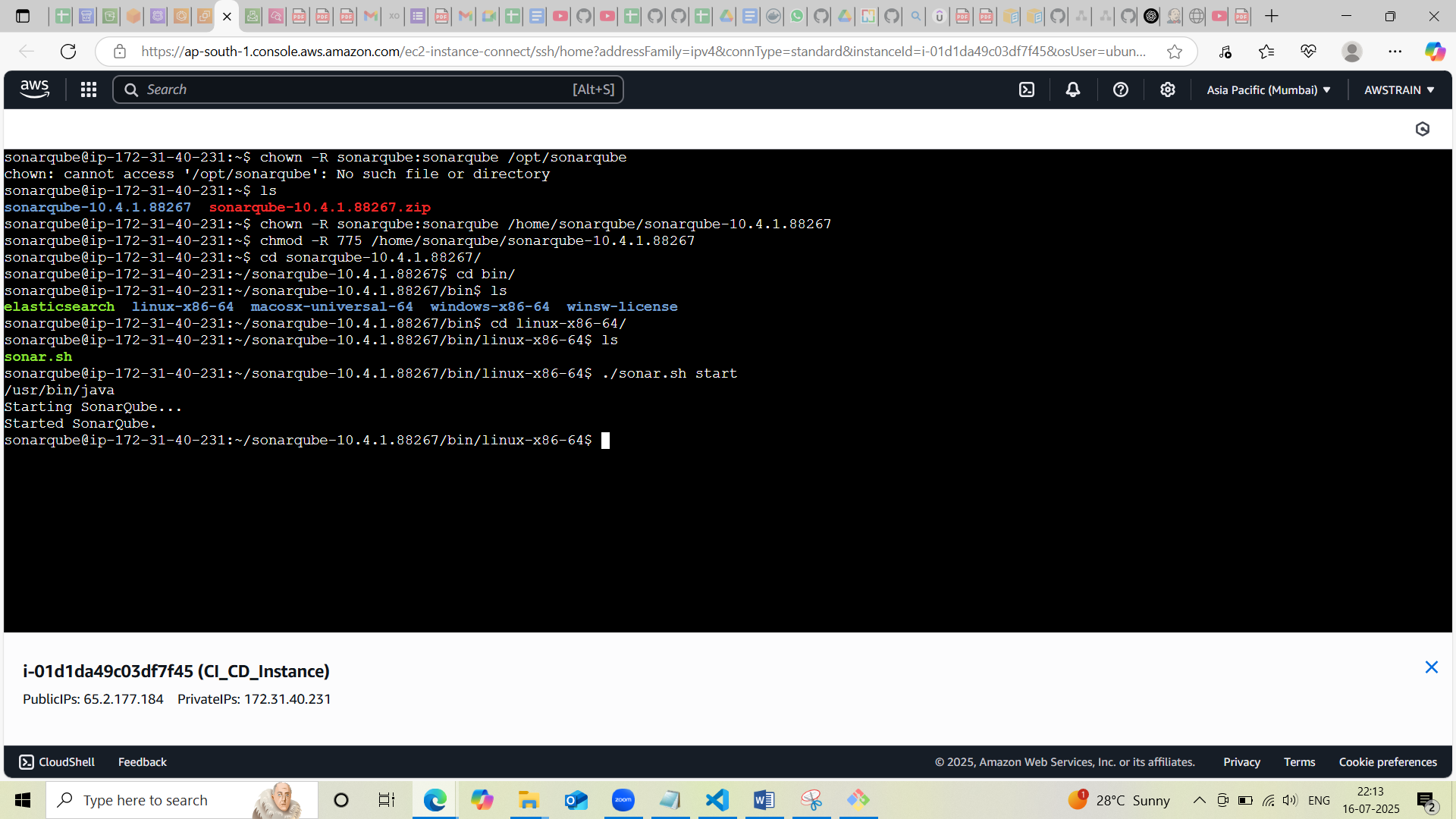
****

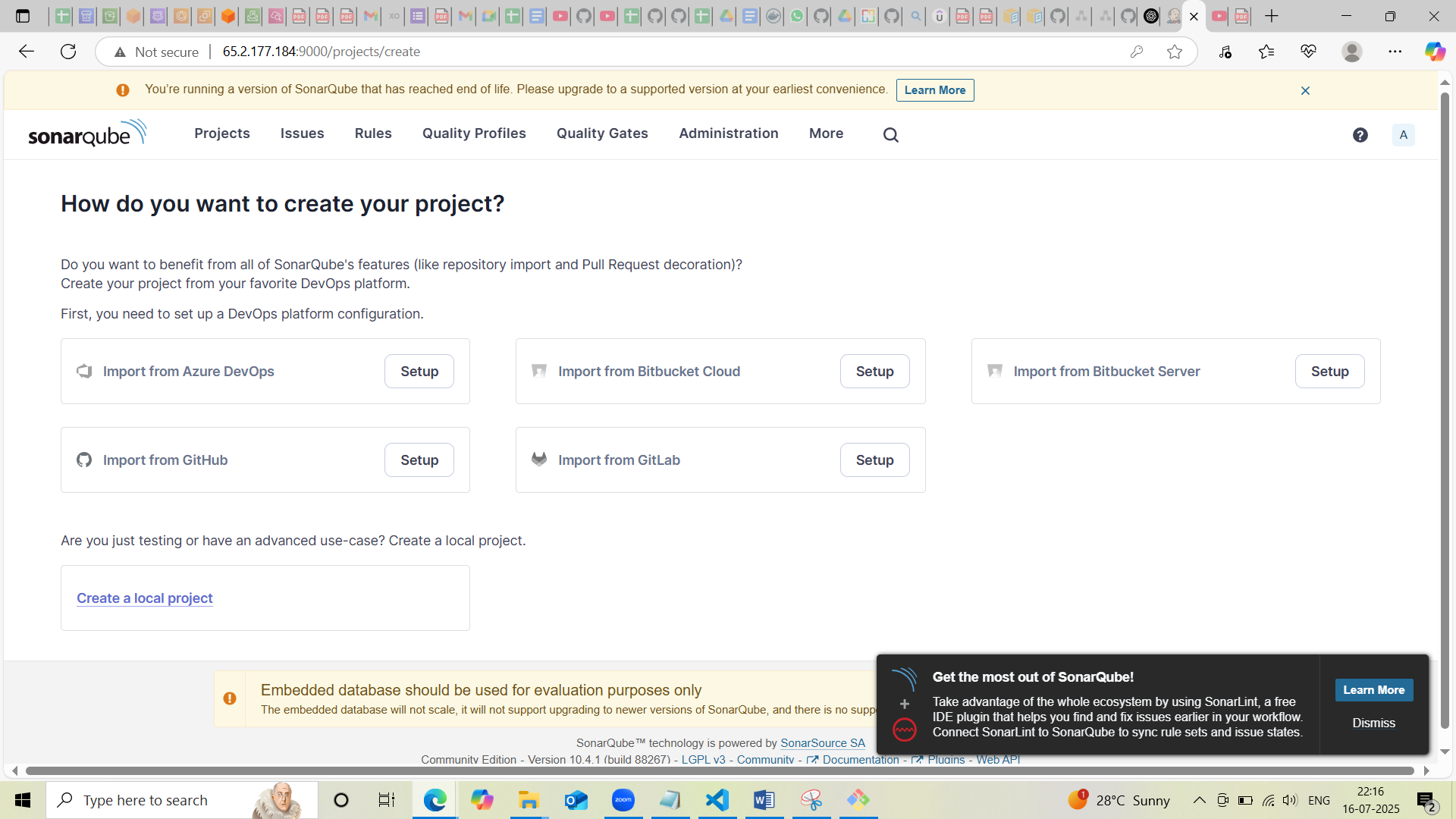
****

**Creation of Jenkins pipeline and installed nessessary plugins such as docker and sonar qube**

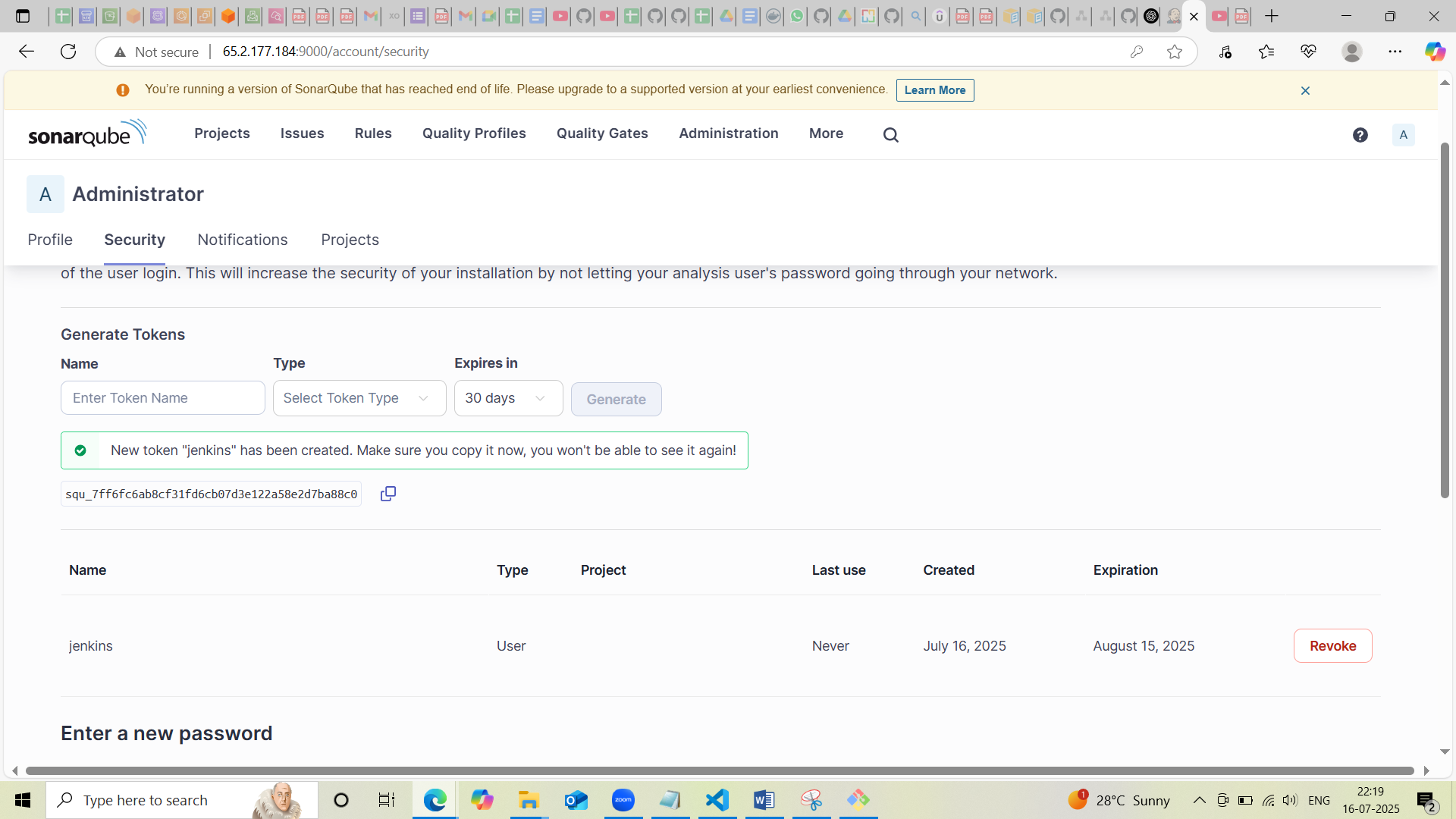
****

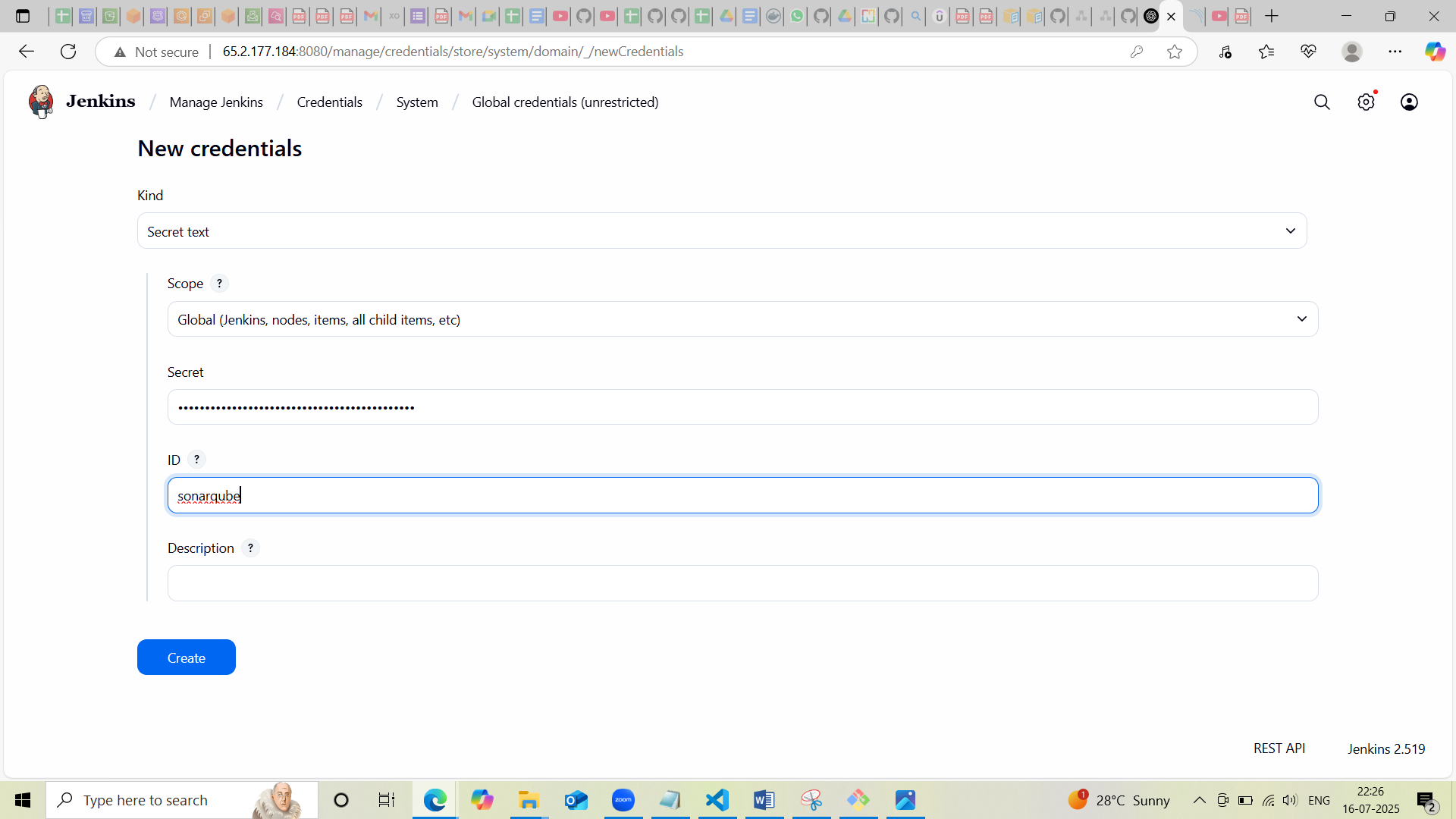
**Installing Sonar qube**

****

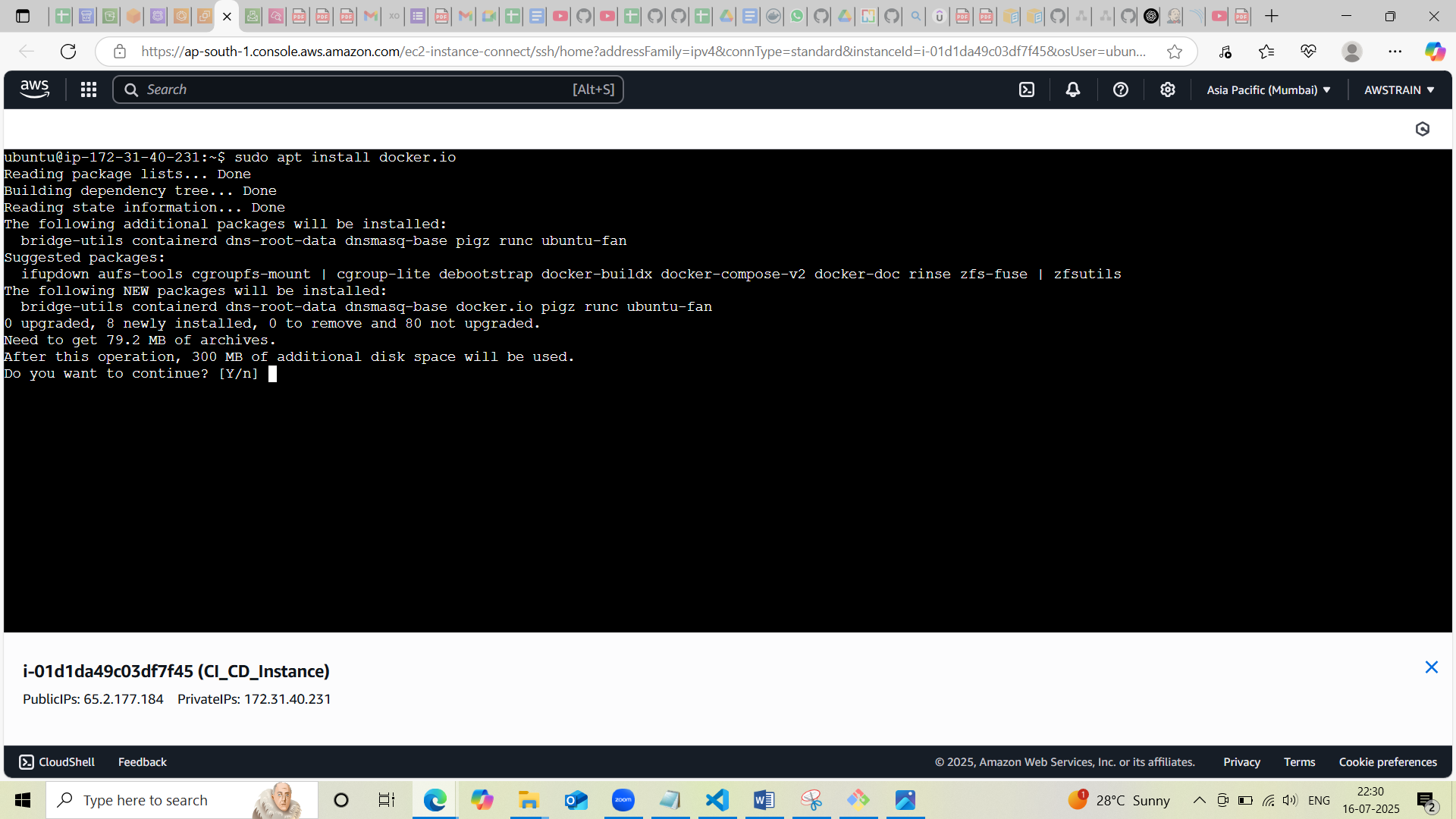
****

**Integrating sonar qube with Jenkins**

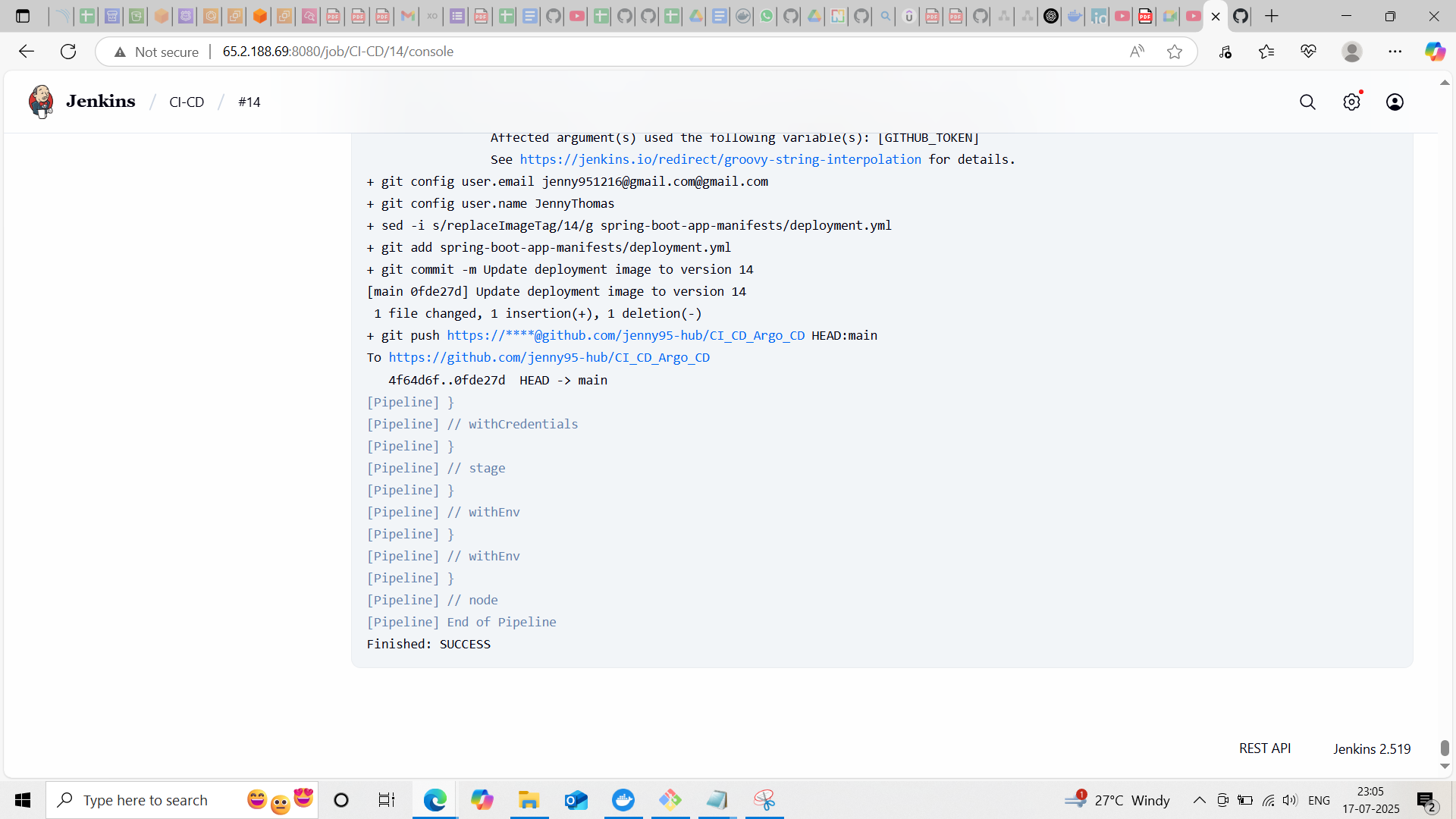
****

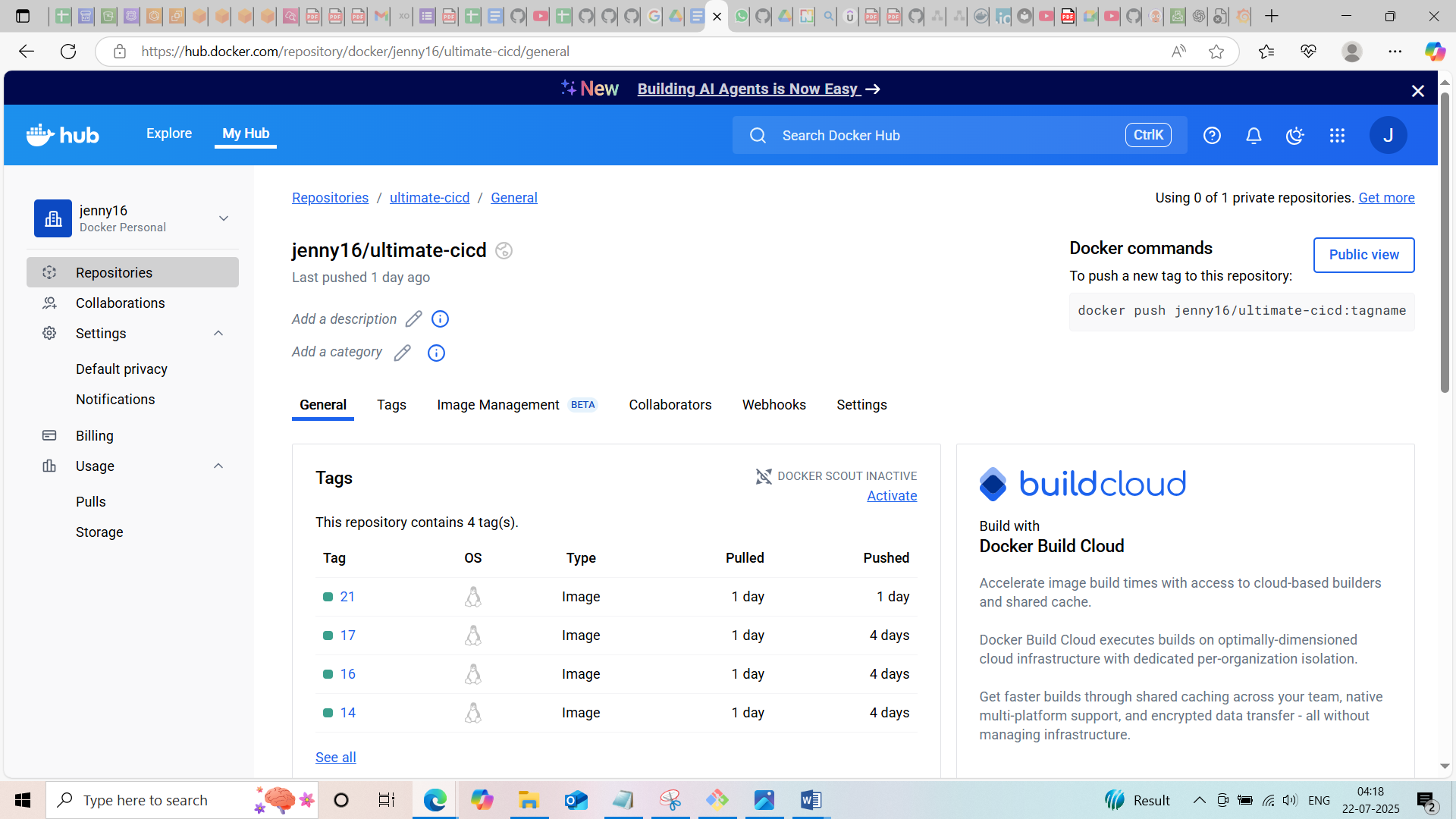
****

**Installation of docker**

****

**Once the successful build of CI part docker image created and pushed to docker hub**

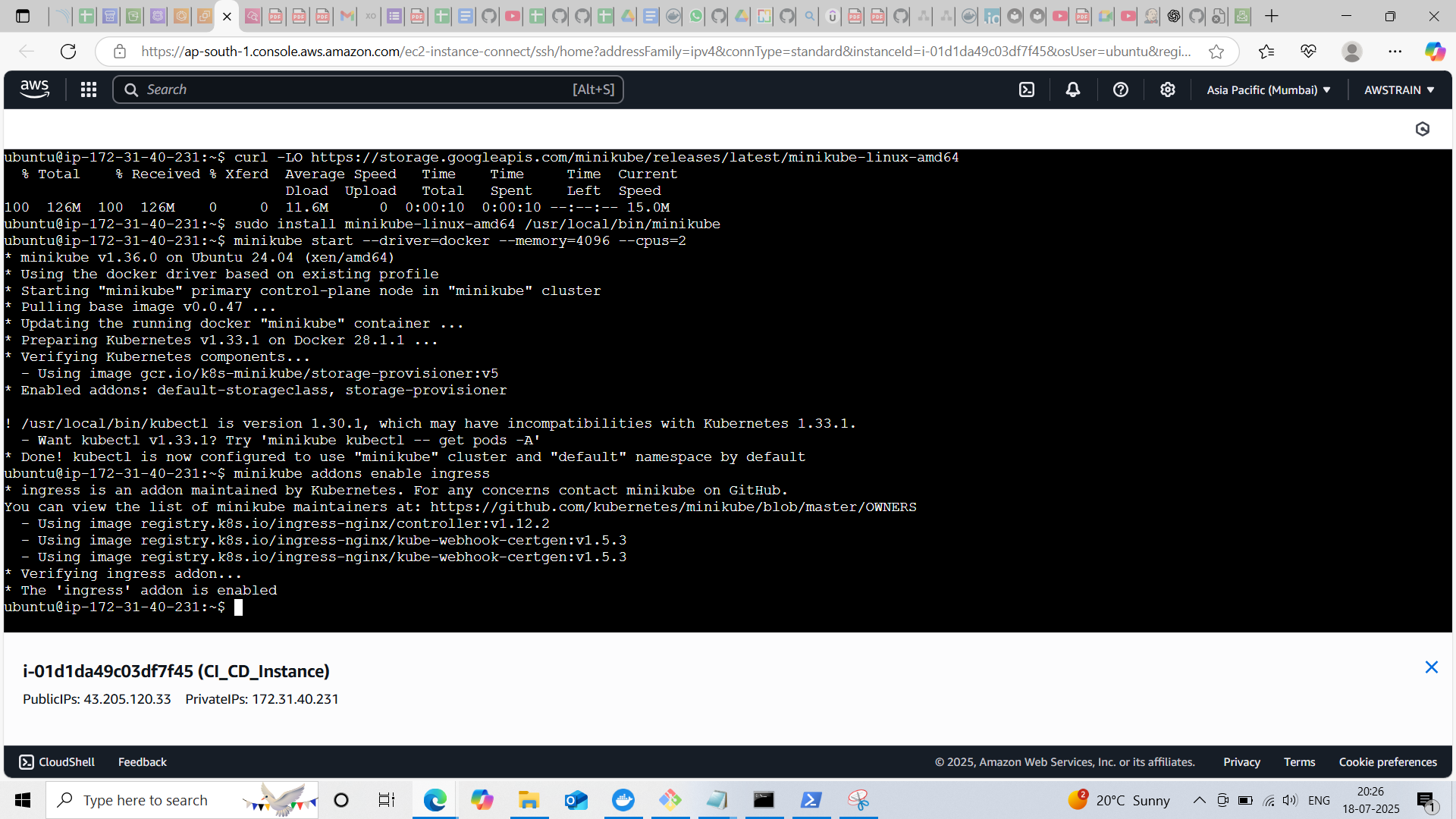
****

****

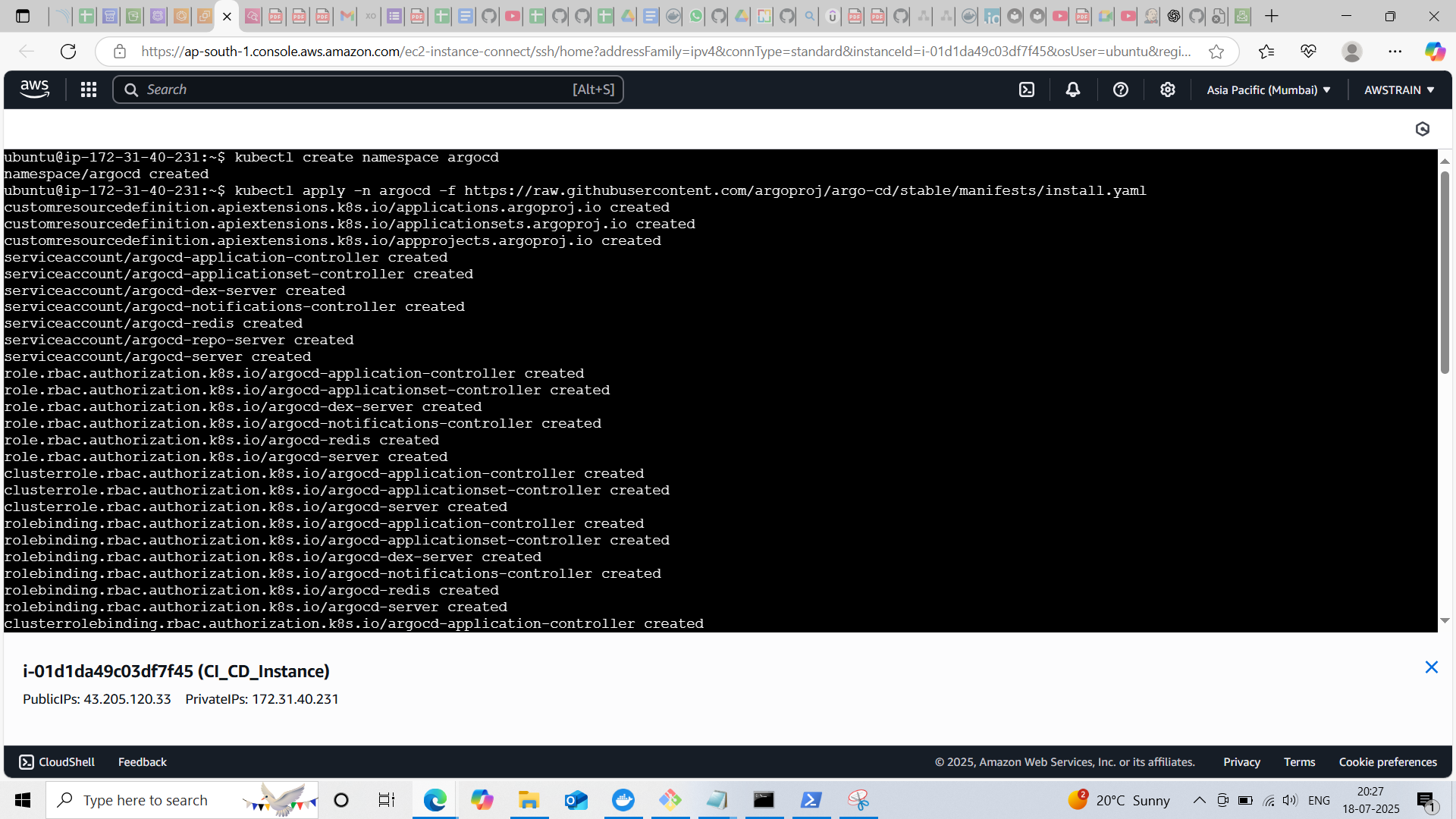
**Sonar qube test also passed**

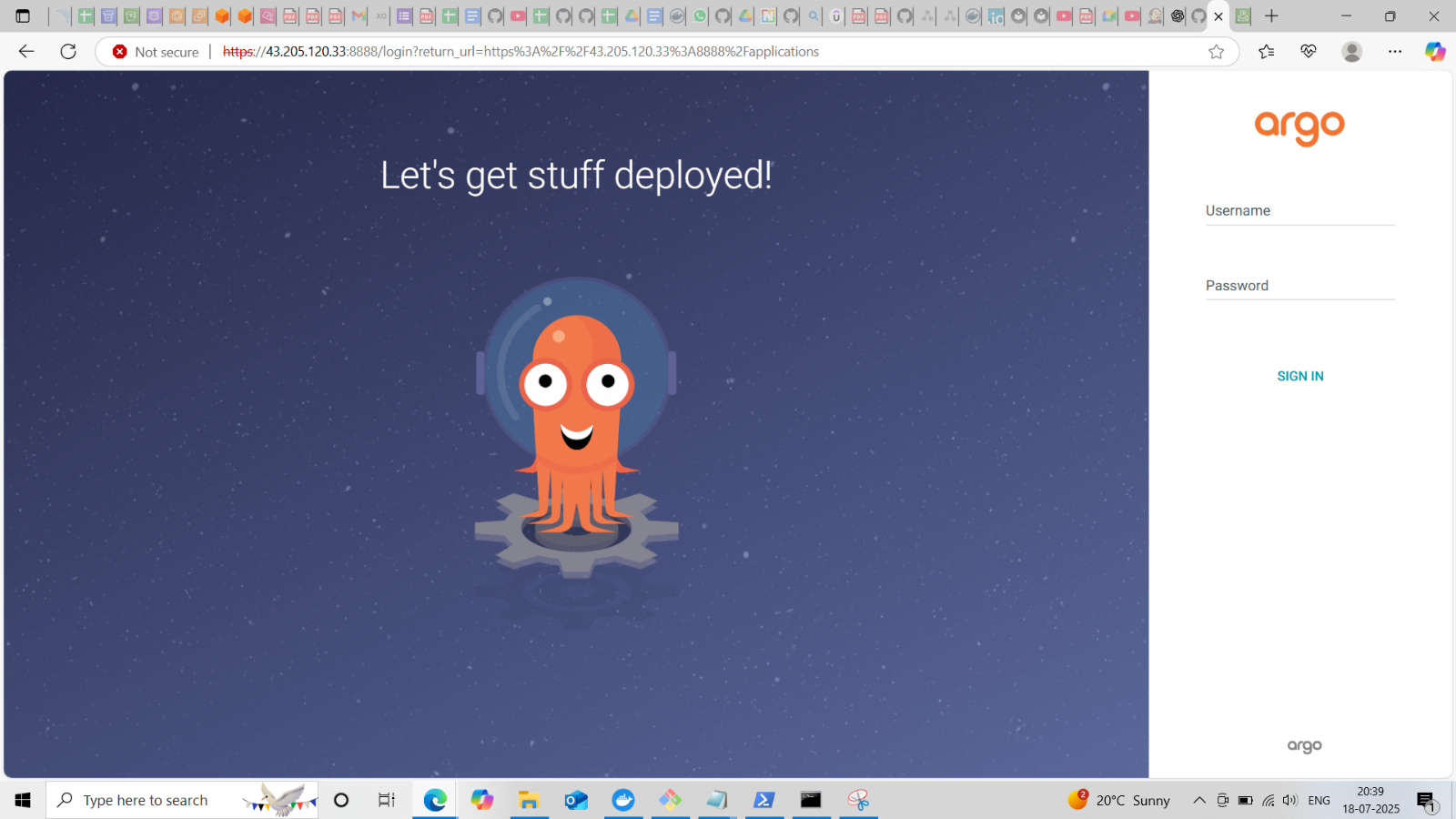
****

**Installing minikube**

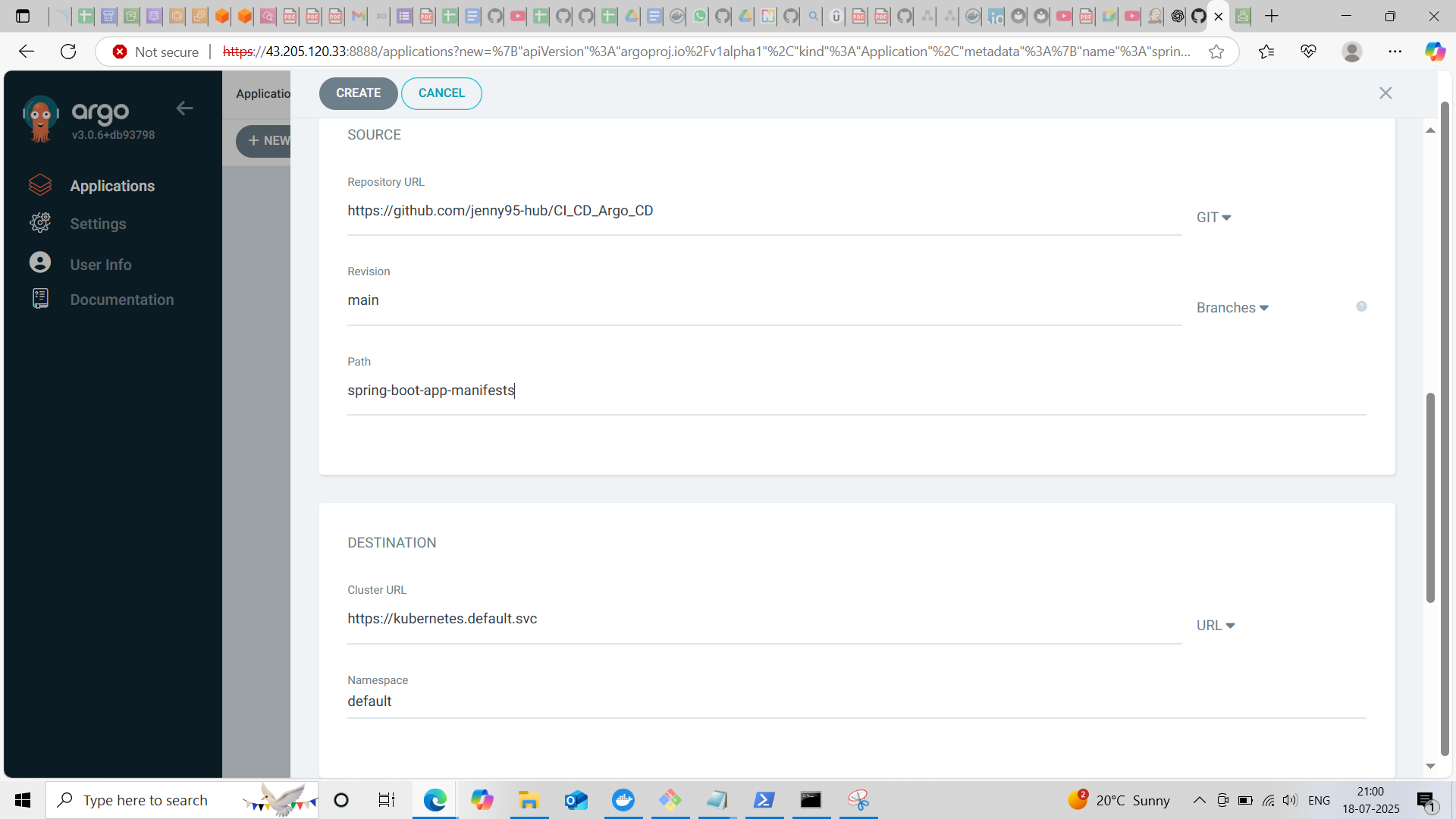
****

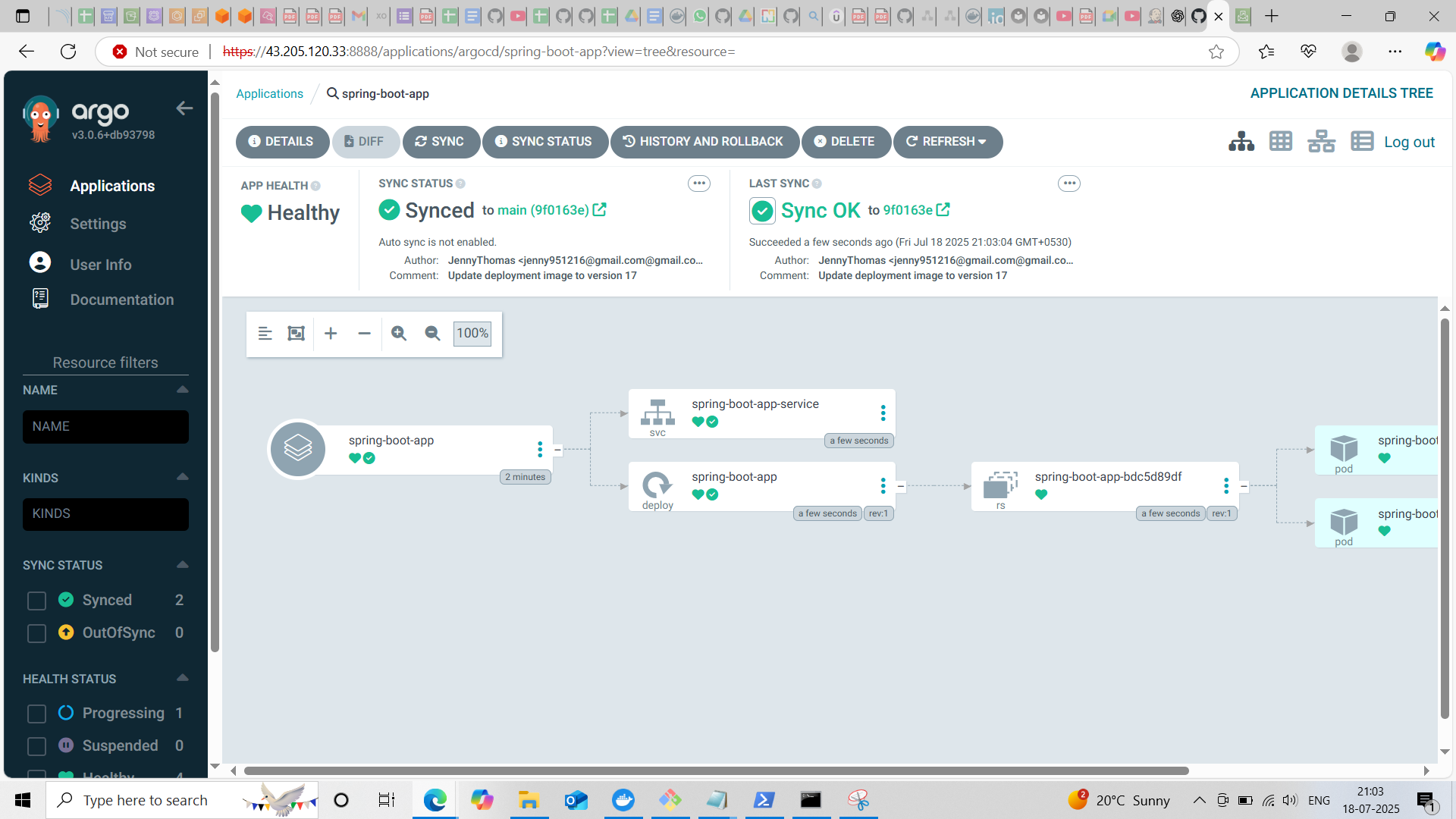
**Installation of Argo CD**

****

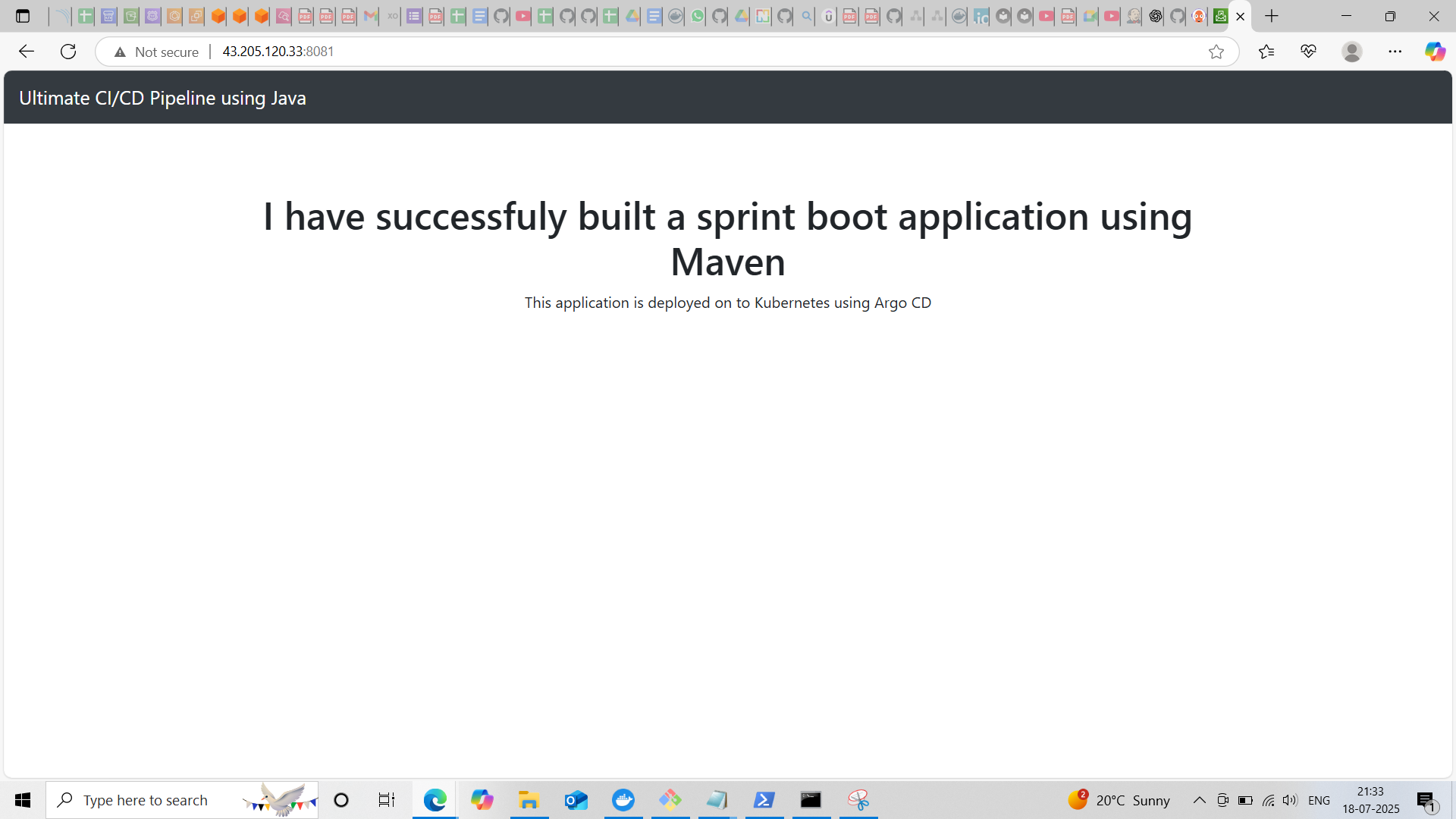
****

**Update the Argo CD with the git repo details**

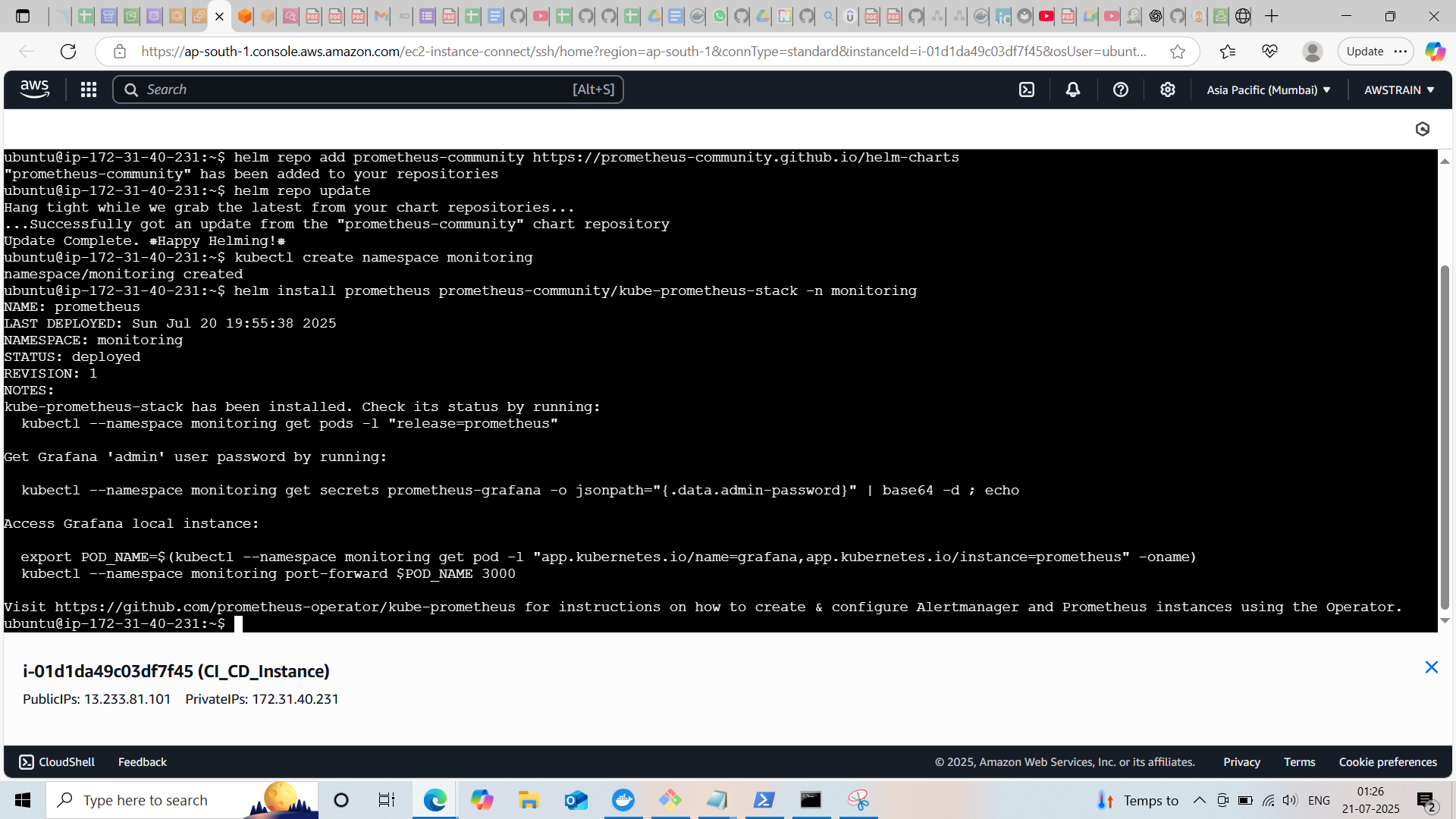
****

****

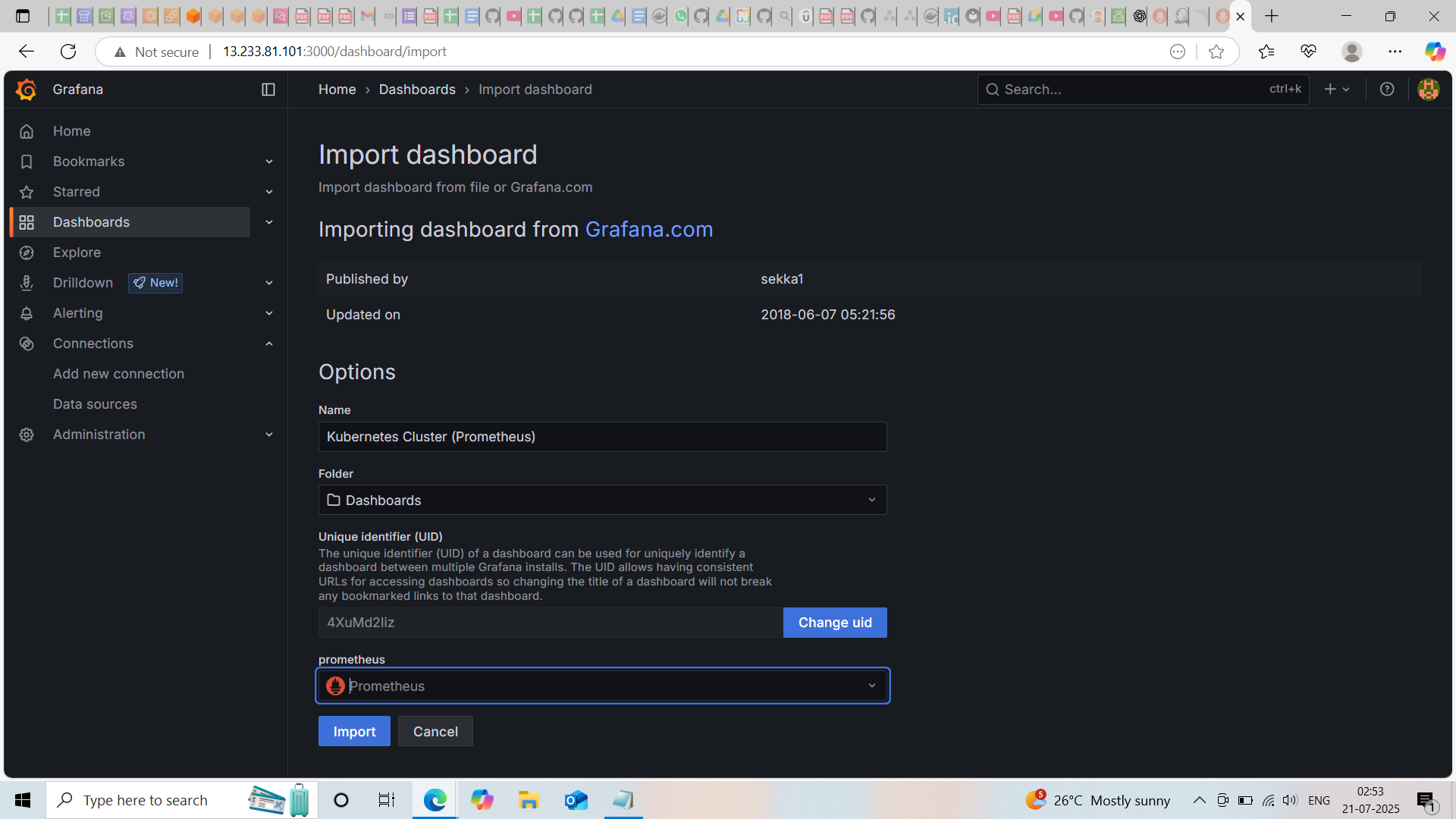
**Spring boot app in the browser**

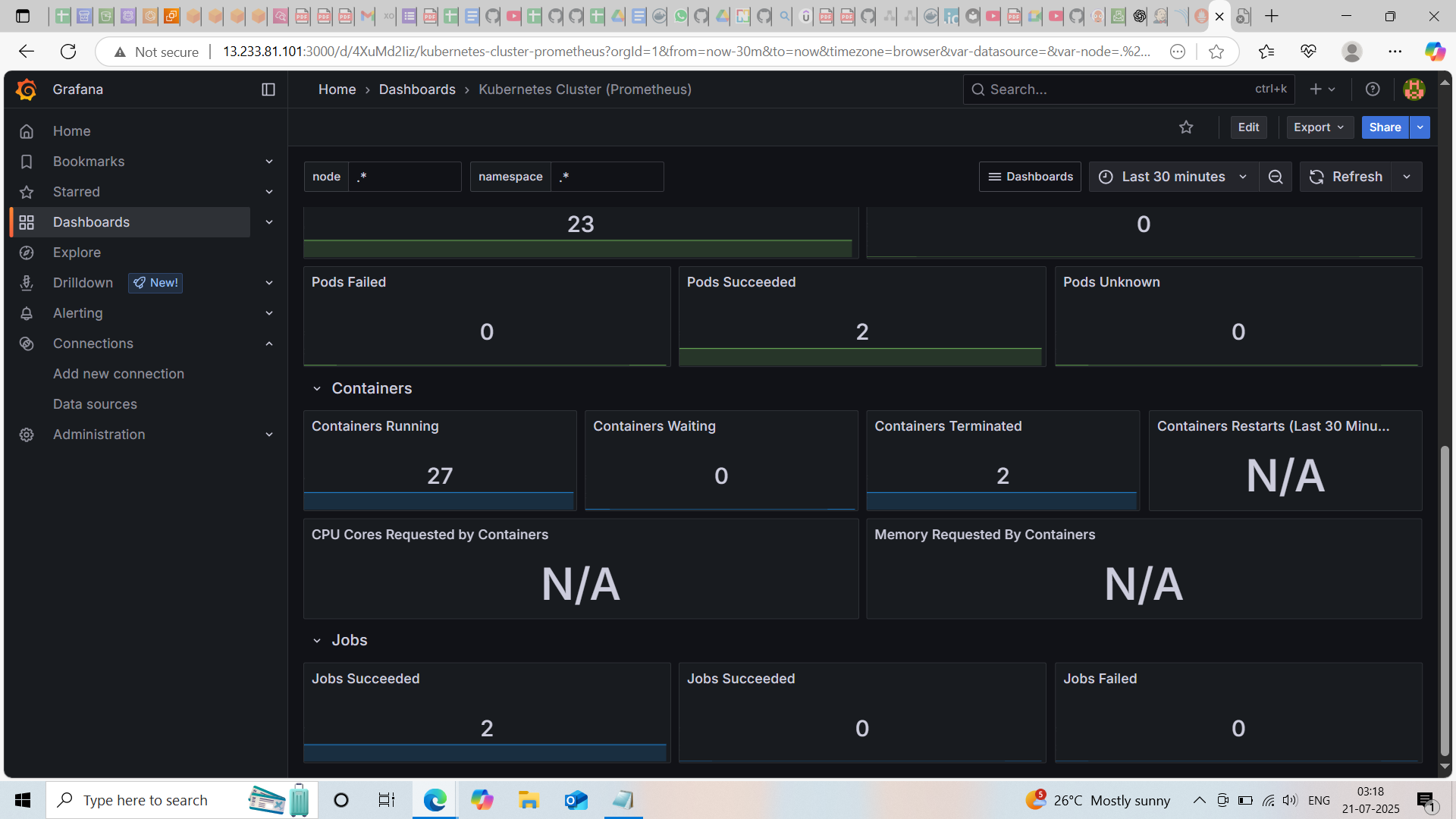
****

**Install kube-prometheus-stack via Helm**

****

**Import the dashboard in grafana and can monitor the matrix**

****

****