**Developing Workflow in SonarQube for Node.js Codebase's Gap Analysis – Assignment**

**Process:**

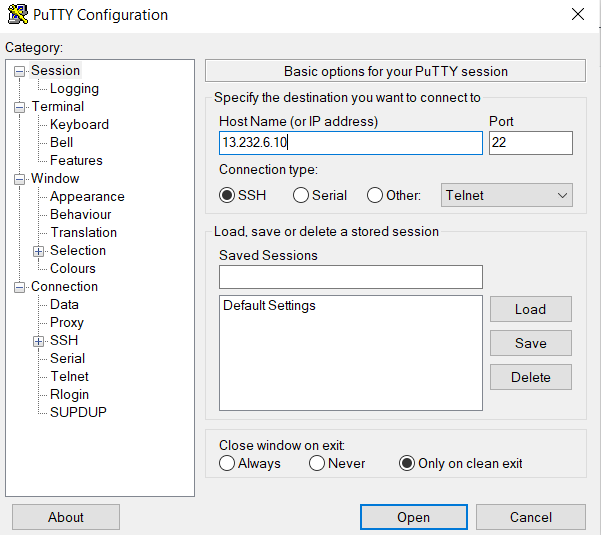
**Set Up AWS Resource**

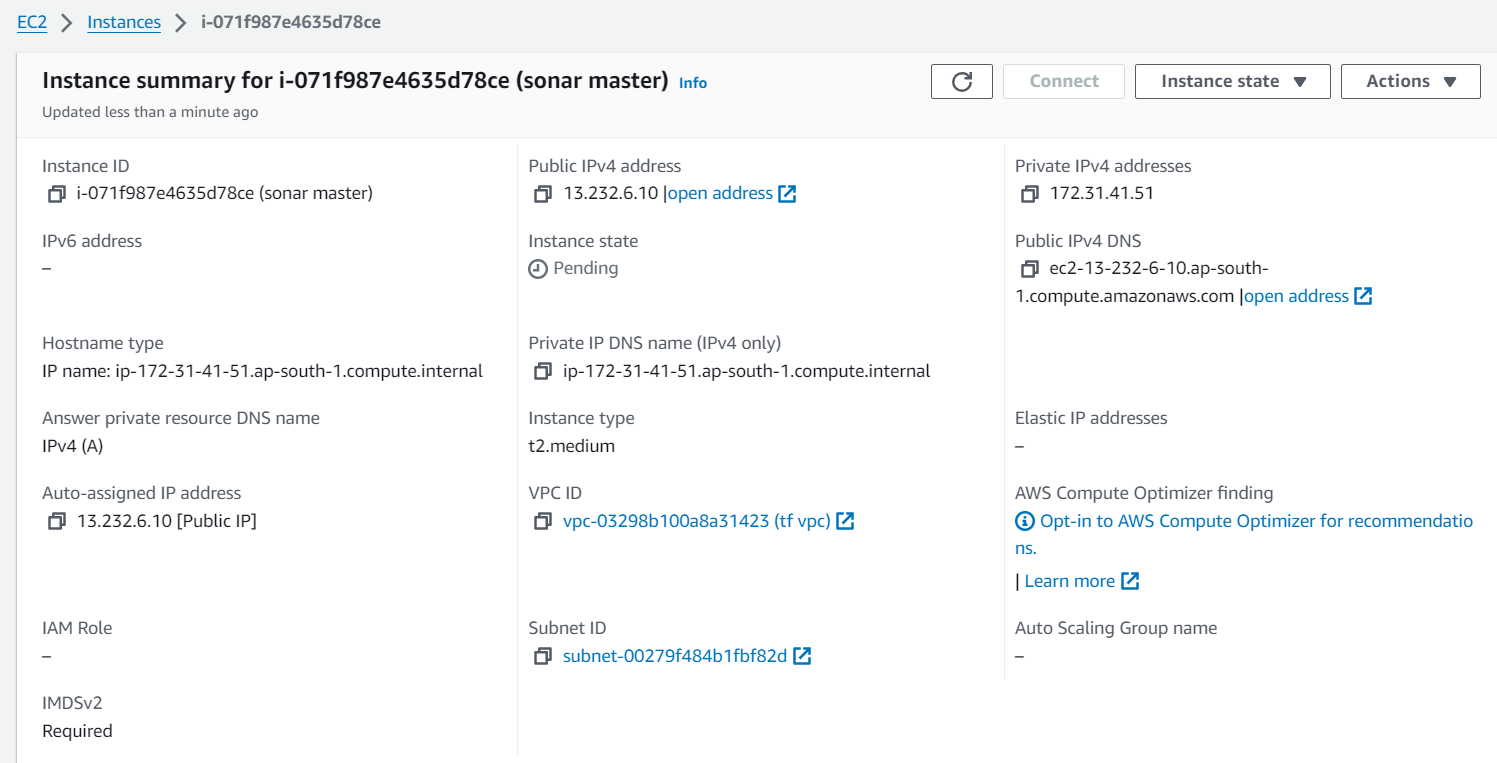
Logged in to AWS Management Console.  
• Launched EC2 Instance: (virtual machine - Ubuntu).

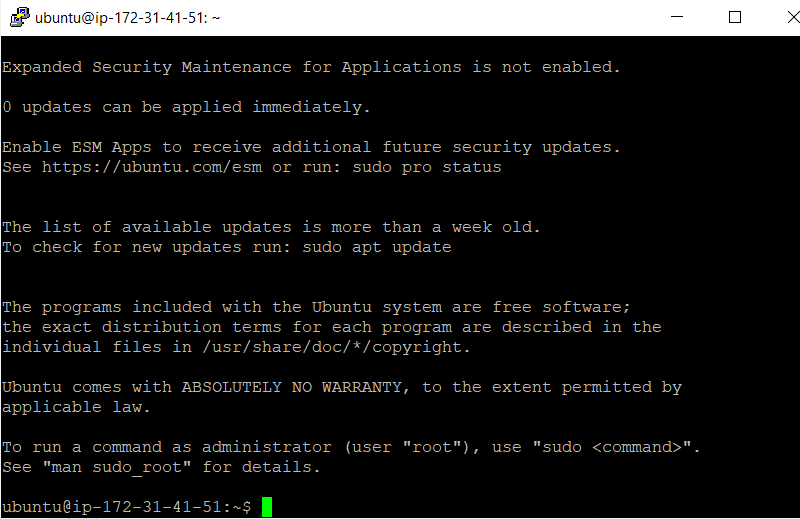
And security groups to allowed SSH (port 22) traffic with instance type - T2.medium and 15Gb disk size and created PPK file.

**Connect to Your EC2 Instance**

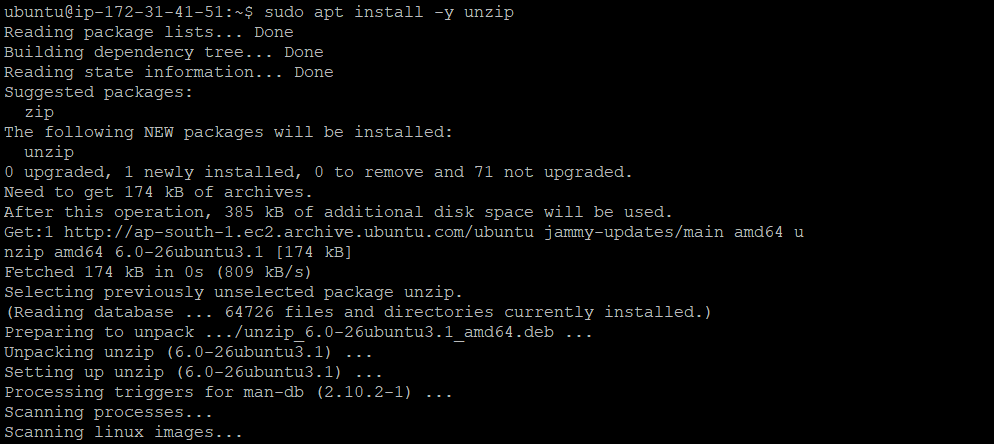
•    Used SSH to connect to EC2 instance using the PPK file via Putty as below.





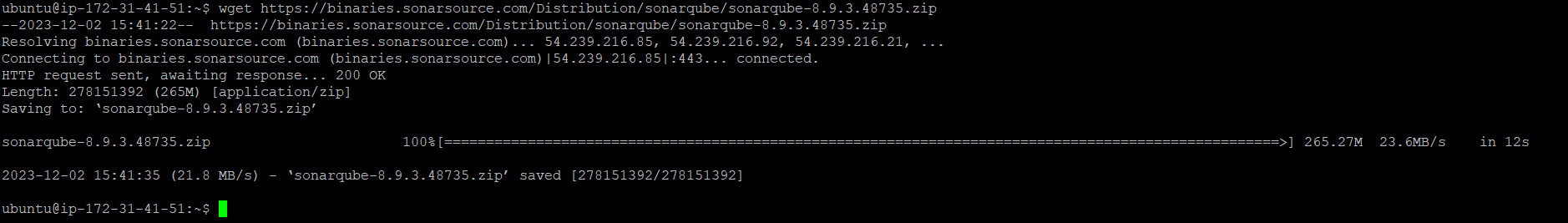


### Installed Java and unzip on to the machine. (Install Prerequisites)

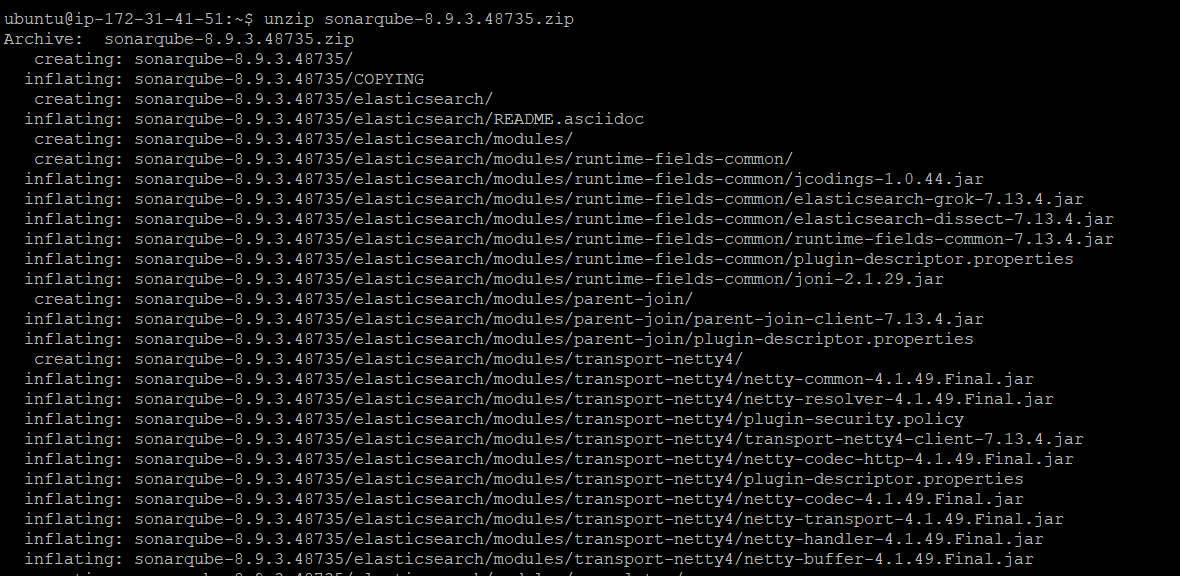




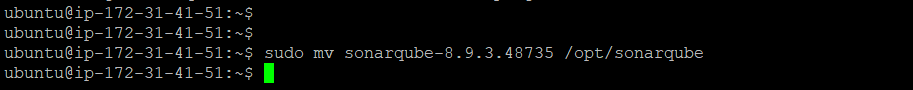
**Download and Install SonarQube**



Unziped the downloaded file

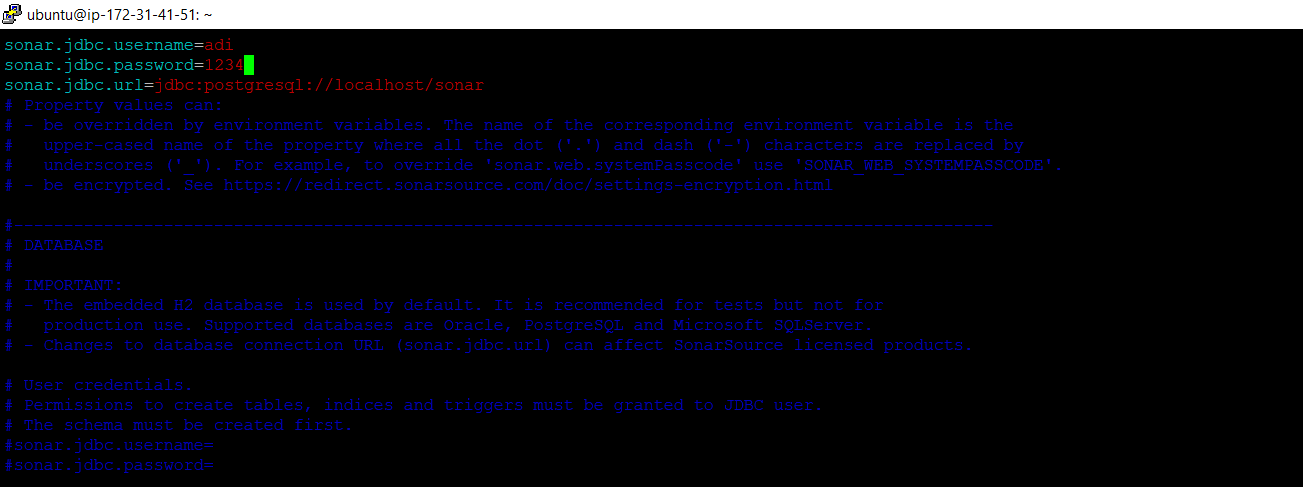


Move the extracted directory to /opt



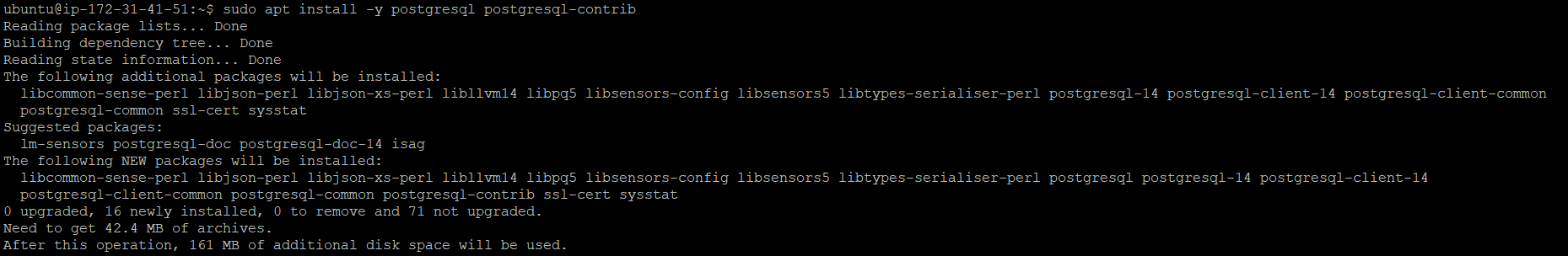
### Configure SonarQube:

Edited the SonarQube configuration file:

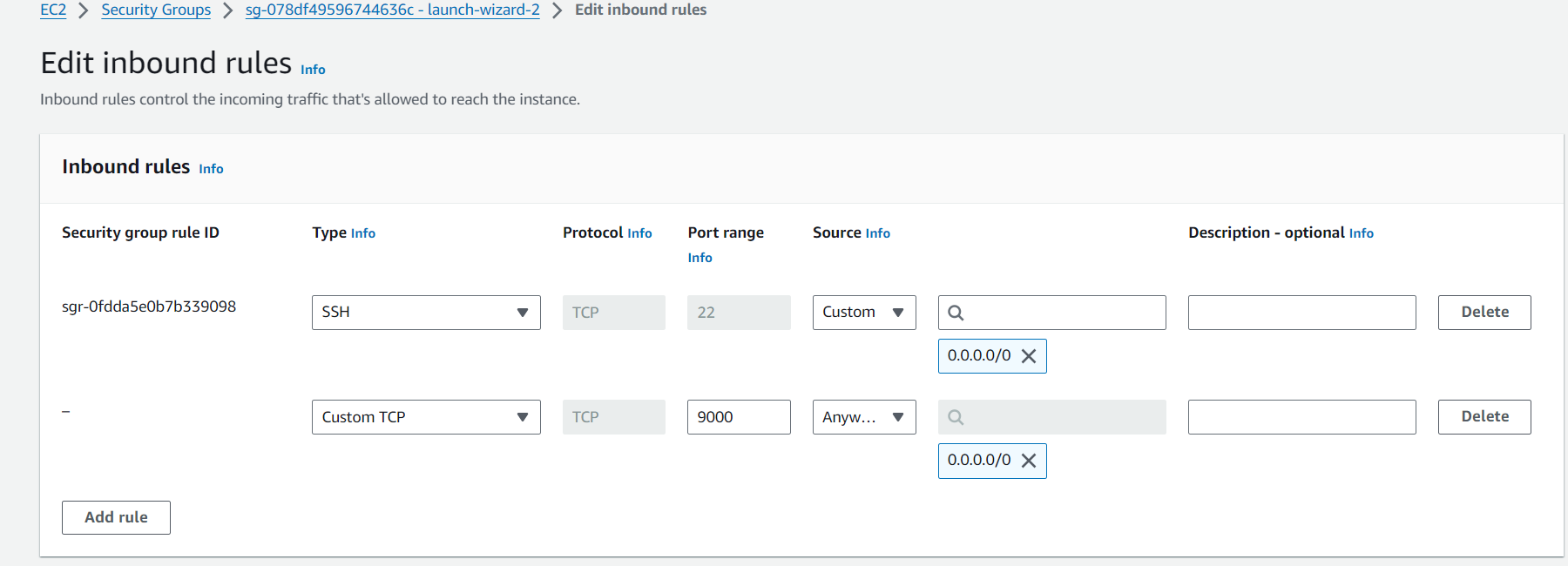


Set Up PostgreSQL (Assuming you are using PostgreSQL):

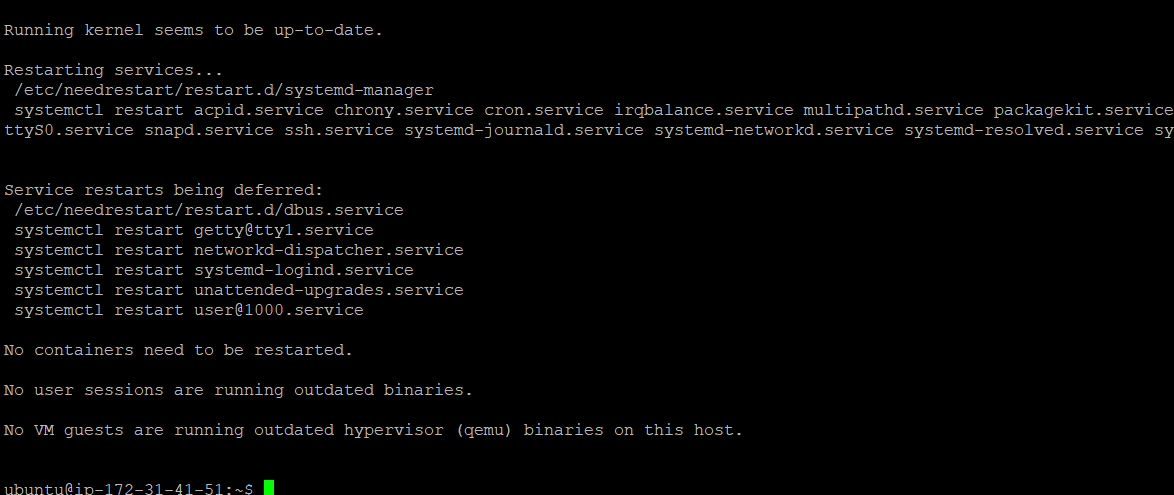




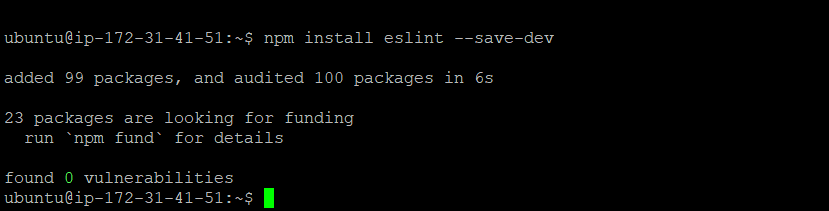
Added 9000 port entry to the machine as sonarqube running on the same port.

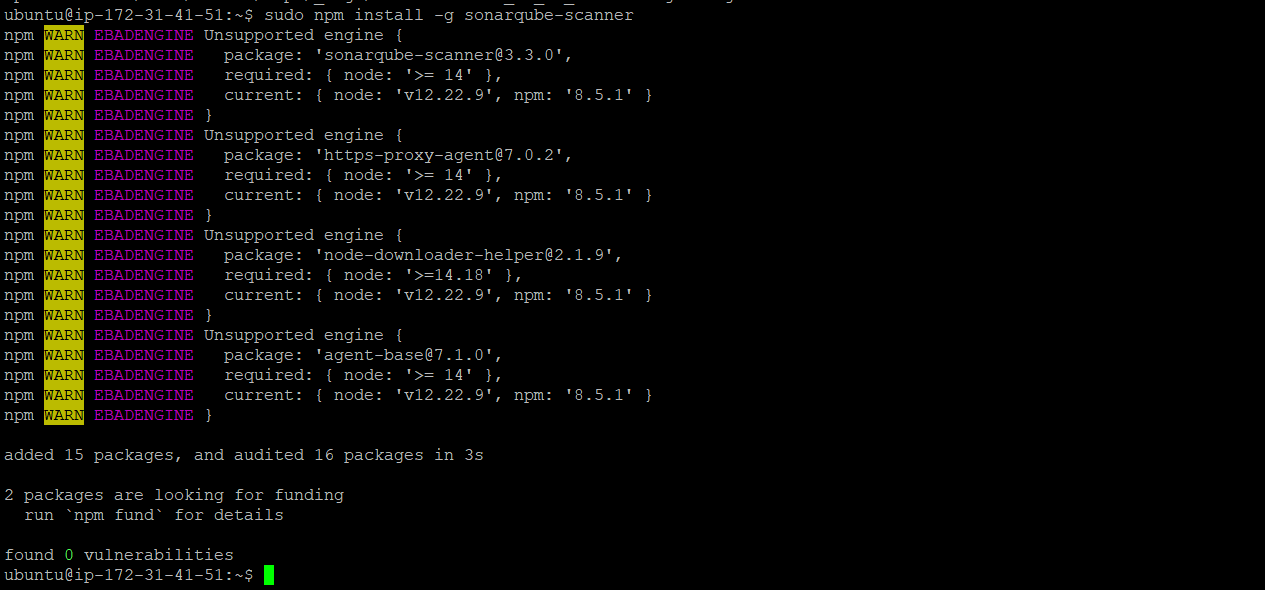


Installed npm

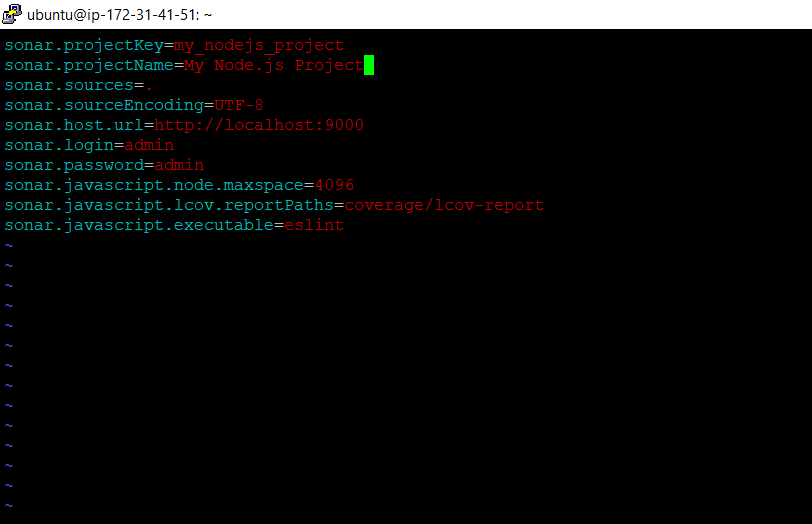


**Configure Node.js Scanner: Node.js project for which we want to perform gap analysis.**

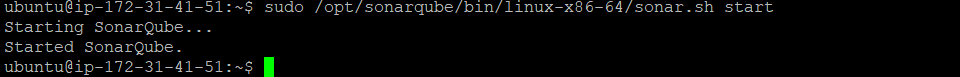




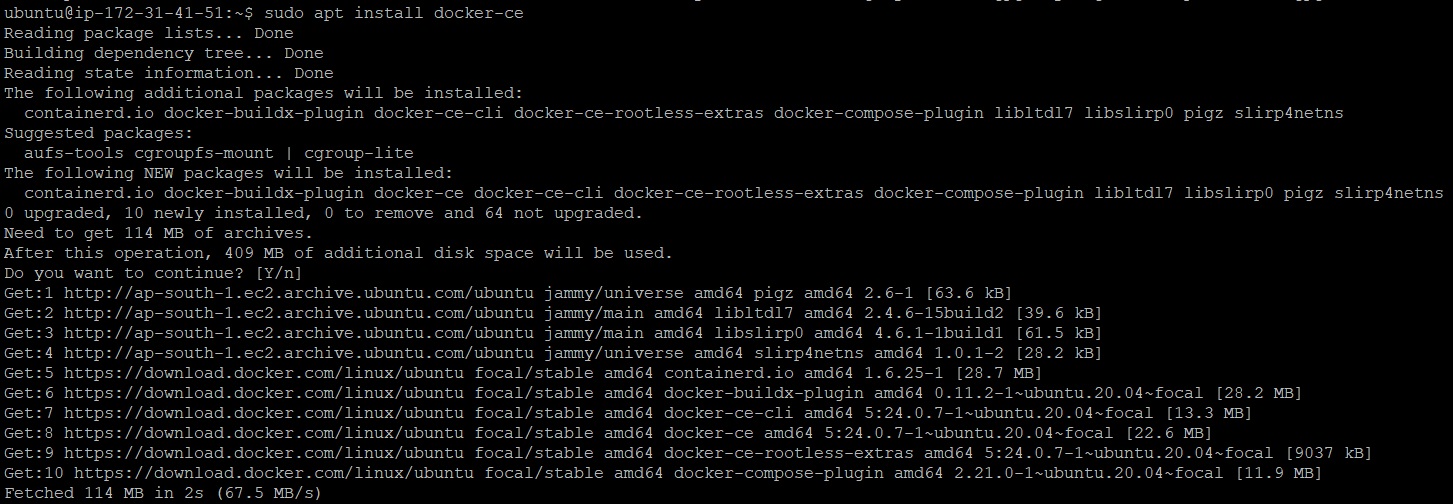
**Created SonarQube Properties File:**



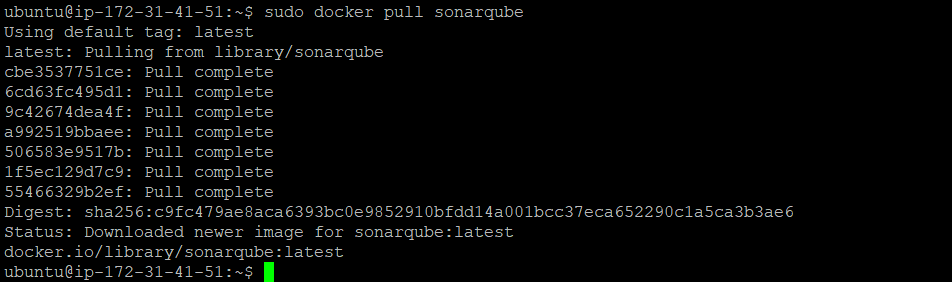
Starting and running the **SonarQube :**



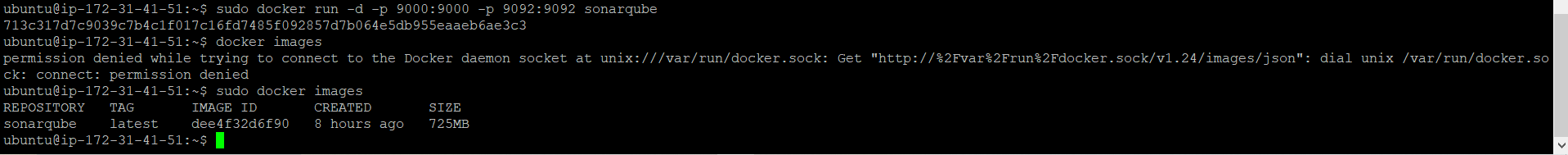
Installed Docker on machine as below:



Pull the SonarQube Docker image: **docker pull sonarqube**



Now Run SonarQube in a Docker container:



Accessed the SonarQube web interface at **http://** 13.232.6.10**:9000**

AS we can see we are able to login and access SonarQube web interface

