Task – 27

ANSIBLE ROLES AND VAULTS

Ansible roles:

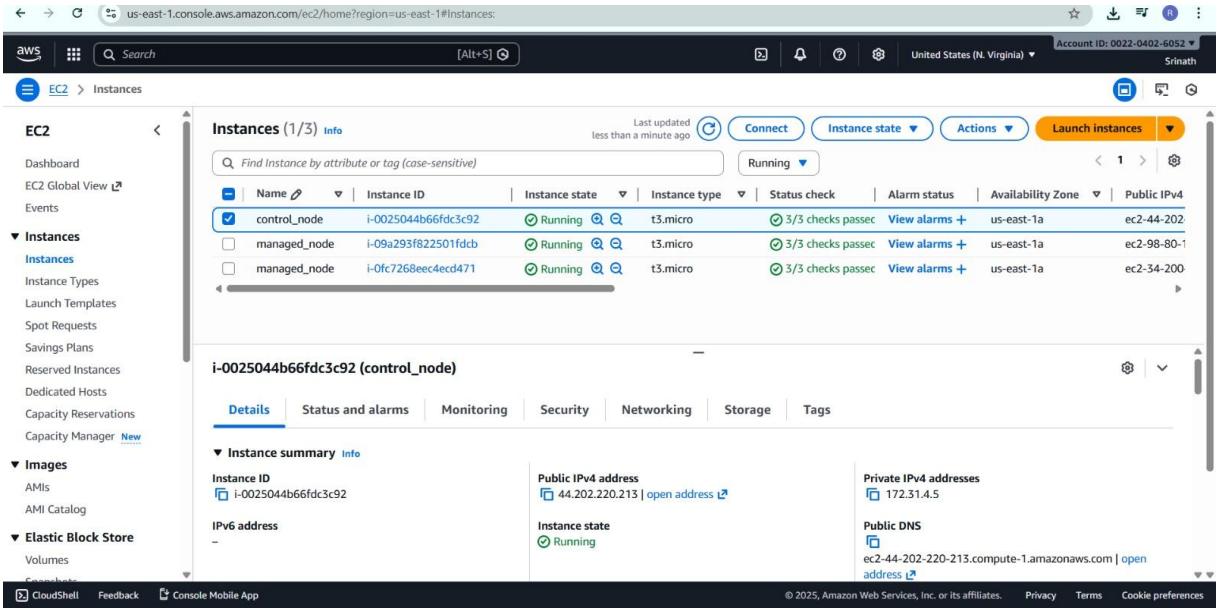
Ansible roles are the primary mechanism for organizing and packaging related automation content (tasks, variables, files, templates, and handlers) into a reusable and modular structure. They help in breaking down large, complex playbooks into smaller, manageable, and shareable components, following the principle of separation of concerns.

Standard Directory Structure

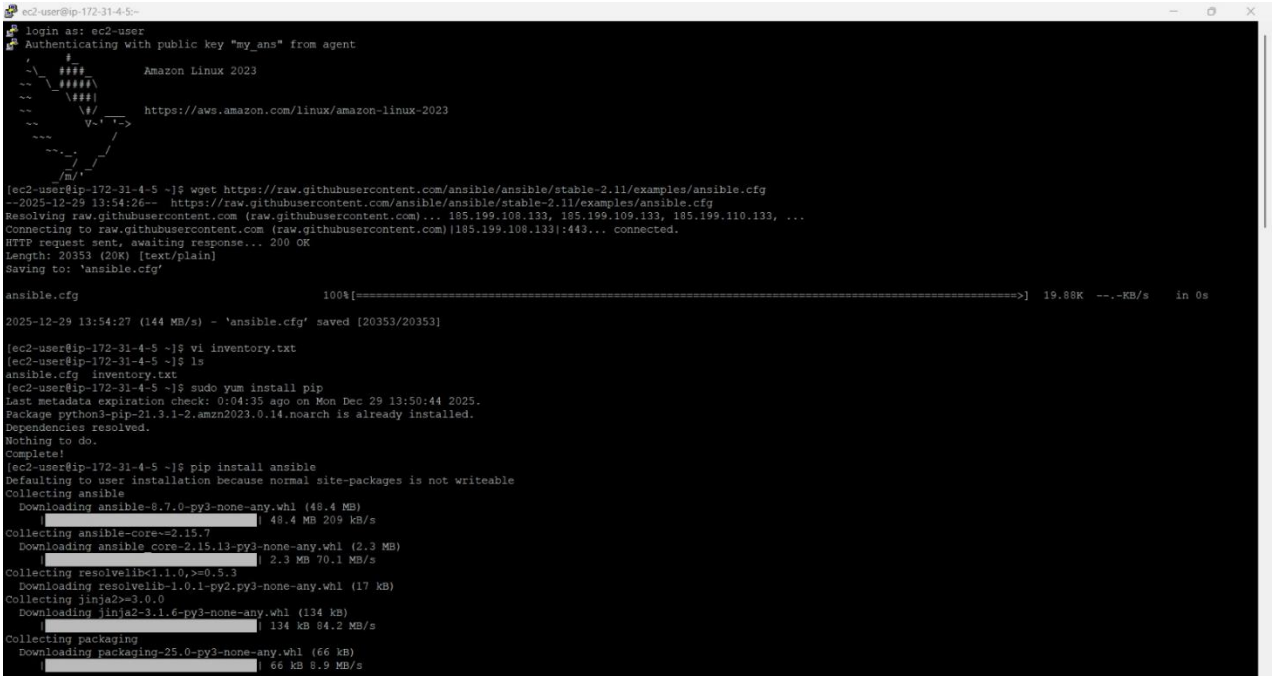
When you create a new role (often using the ansible-galaxy role init [role_name] command), Ansible sets up a specific directory structure. Ansible automatically looks for a main.yml file within each of these standard directories.

- **tasks/ -** Contains the main list of tasks executed by the role (main.yml).
- **handlers/** - Stores handlers that run only when notified (for example, restarting a service after configuration changes).
- **defaults/** - Holds low-priority default variables that can be easily overridden.
- **vars/** - Contains high-priority variables, mainly for internal role usage and harder to override.
- **files/** - Stores static files that are copied directly to target hosts without modification.
- **templates/** - Contains Jinja2 template files used to generate dynamic configuration files.
- **meta/** - Defines role metadata such as author, license, supported platforms, and role dependencies.
- **library/** - Includes custom Ansible modules specific to the role (optional and advanced).

Creating 3 instance both control and managed nodes



Installing pip and ansible in control node



Creating a role using ansible galaxy command:

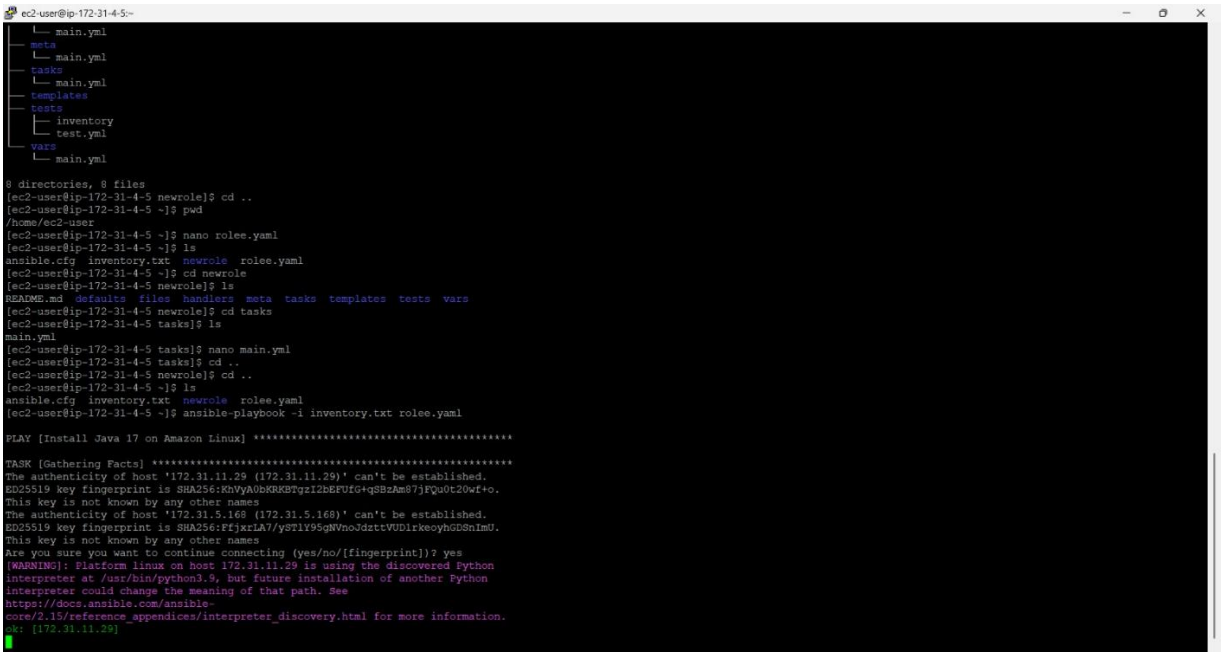
Code should store into rolee.yaml:

name: Install Java 17 on Amazon
Linux hosts: all
become: yes
roles:
- newrole

Code should store into main.yaml file and download java:

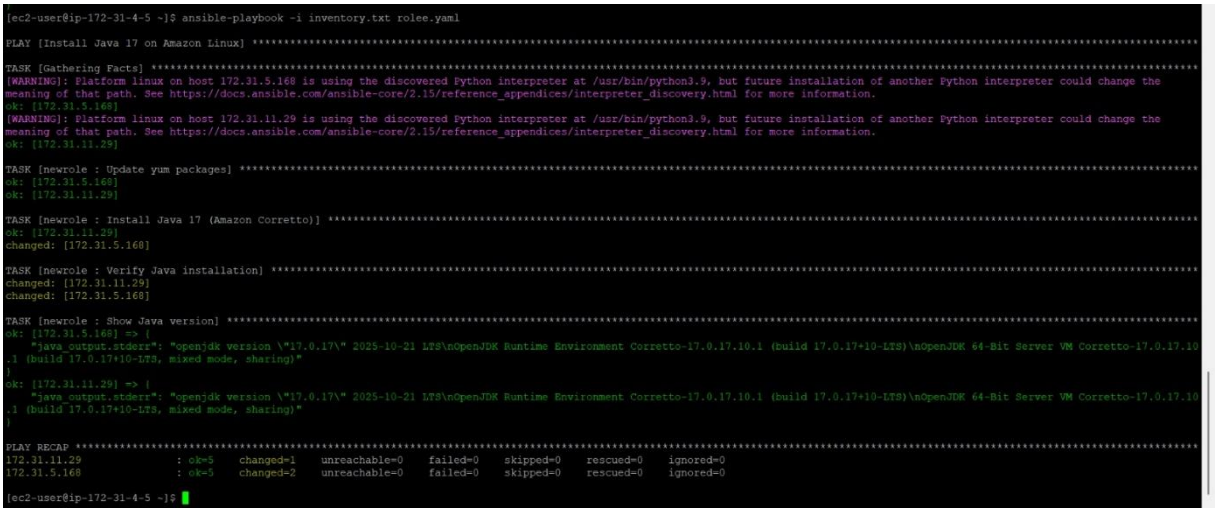
tasks file for newrole

```
- name: Update yum
  packages yum:
    name: "*"
    state:
    latest
- name: Install Java 17 (Amazon
  Corretto) yum:
    name: java-17-amazon-corretto
    state: present
- name: Verify Java
  installation command: java
  -version register:
  java_output ignore_errors:
  yes
- name: Show Java
  version debug:
    var: java_output.stderr
```

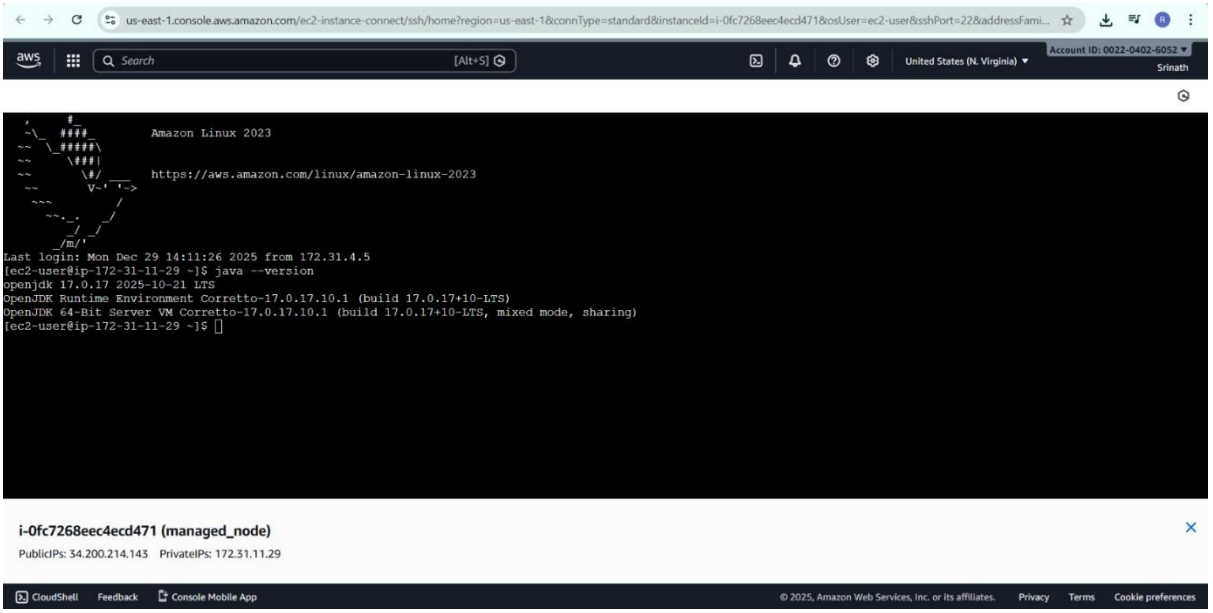


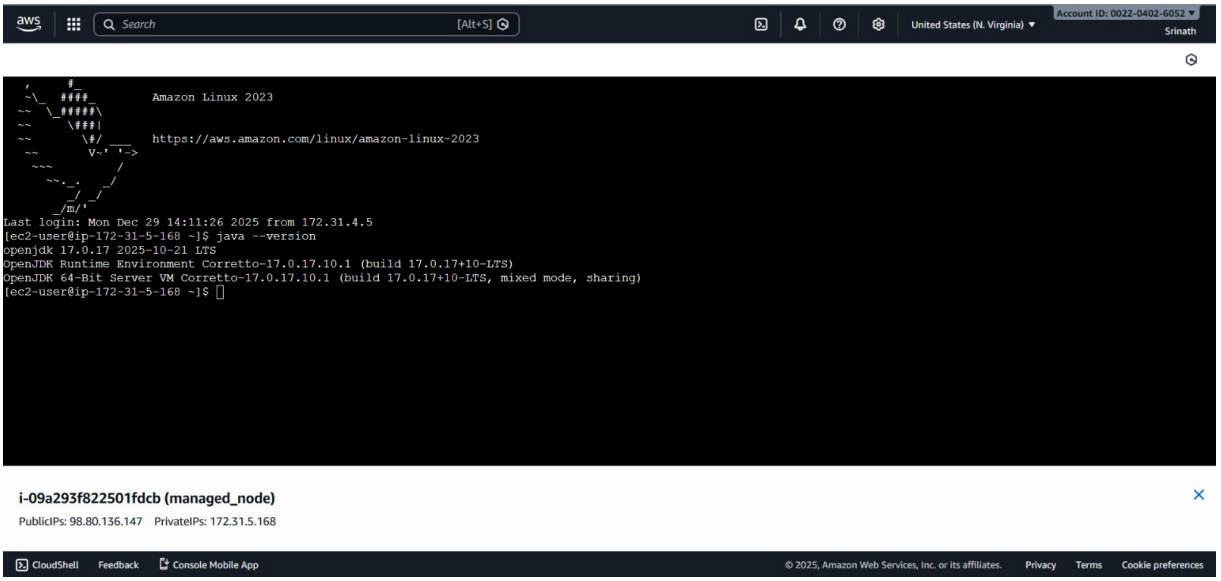
Then run the yaml file(playbook):

Command: ansible-playbook -I inventory.txt rolee.yaml



Checking whether java is installed in managed nodes…





Ansible vaults:

Ansible Vault is a built-in Ansible feature that encrypts sensitive data, like passwords, API keys, and tokens, within Ansible files (playbooks, variables) to prevent them from being stored in plain text, allowing secure management of secrets alongside automation code, with operations like encryption, decryption, viewing, and editing handled by a password. It uses AES256 encryption and ensures that even if someone gets the file, the secrets remain unreadable without the correct password, making it safe for version control systems like Git.