**ANSIBLE PLAYBOOK**

Ansible is a widely used open-source automation tool in the DevOps ecosystem. It facilitates IT automation, configuration management, and application deployment, enabling DevOps teams to improve efficiency, reduce errors, and streamline processes.

Here’s how Ansible is commonly used in DevOps:

* Configuration Management
* Infrastructure as Code (IaC)
* Application Deployment
* Continuous Integration and Continuous Deployment (CI/CD)
* Security and Compliance
* Orchestration

**Why Ansible in DevOps?**

* Agentless: No need to install agents on managed nodes, reducing overhead.
* Ease of Use: Simple, human-readable YAML syntax.
* Idempotent: Ensures the same playbook can run multiple times without side effects.
* Extensibility: Large library of modules for various tasks and support for custom modules.
* Community: A strong community and ecosystem, offering a vast collection of reusable roles.

**Ansible Inventory File:**

An Ansible inventory file is a key component in Ansible that defines the systems (hosts) and groups of systems to manage. It tells Ansible where to find the machines it will manage and how to connect to them.

**Types of Inventory**

* Static Inventory: A fixed inventory file where hosts and groups are explicitly listed.
* Dynamic Inventory: A script or plugin dynamically generates the inventory based on external data sources (e.g., AWS, GCP, Azure, etc.).

**Passwordless Authentication:**

**Passwordless authentication** in Ansible involves configuring **SSH key-based authentication** between the Ansible control node and the managed hosts. This eliminates the need for manually entering passwords during playbook execution or ad-hoc commands.

**Key Components of Ansible Syntax:**

#### 1. **Playbooks**

A playbook is a YAML file that defines tasks to automate.

#### 2. **Inventory**

An inventory defines managed hosts and groups. It can be in **INI** or **YAML** format.

#### 3. **Modules**

Modules are reusable, standalone scripts that perform tasks.

#### 4. **Variables**

Variables store dynamic values.

#### 5. **Handlers**

Handlers are triggered by tasks and typically manage state changes like restarting services

#### 6. **Loops**

Loops allow you to repeat tasks.

#### 7. **Conditionals**

Conditionals allow tasks to run only when specific criteria are met.

#### 8. **Roles**

Roles are a way to organize playbooks into reusable components.

**Common Syntax Rules**

**YAML Basics:**

* Use spaces for indentation (no tabs).
* Indentation determines hierarchy.
* Key-value pairs use key: value.

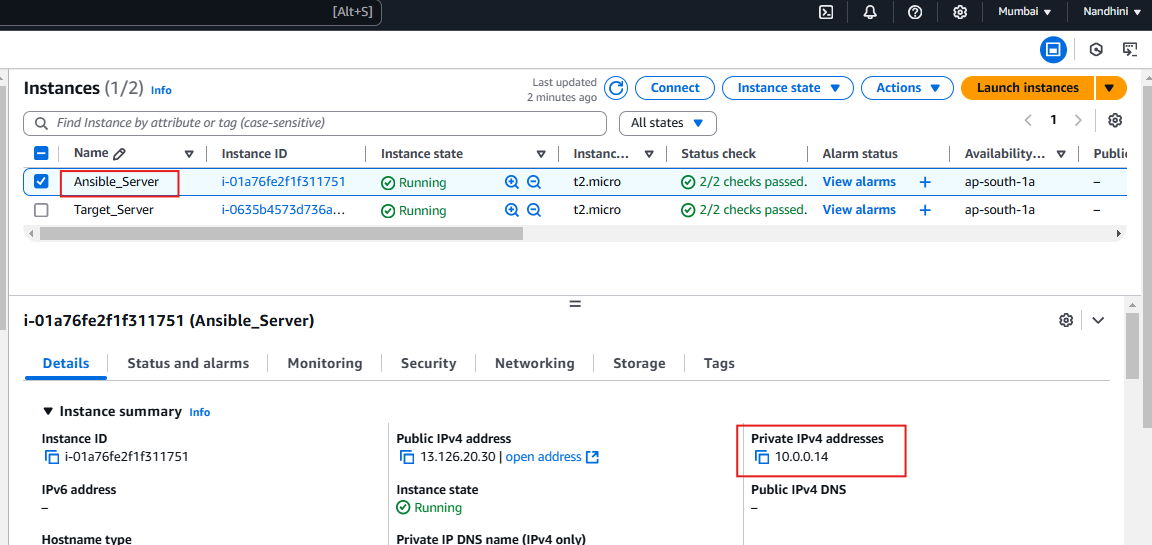
**Quoting Variables:**

* Use double curly braces ({{ variable }}) for variables.
* Wrap variables in quotes when used in strings.

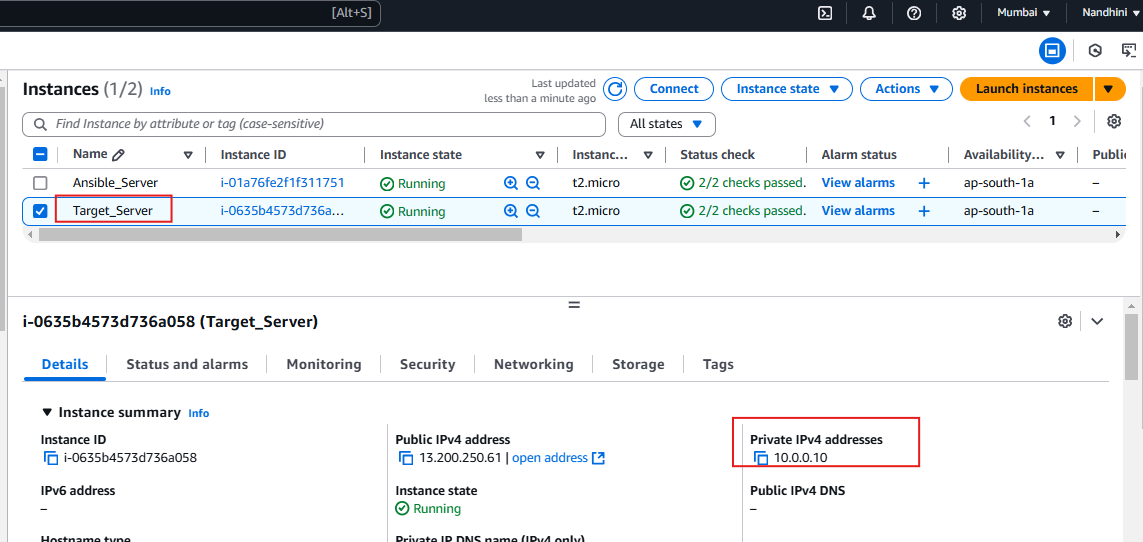
**Error Handling:**

* Syntax errors often occur due to incorrect indentation or missing colons.

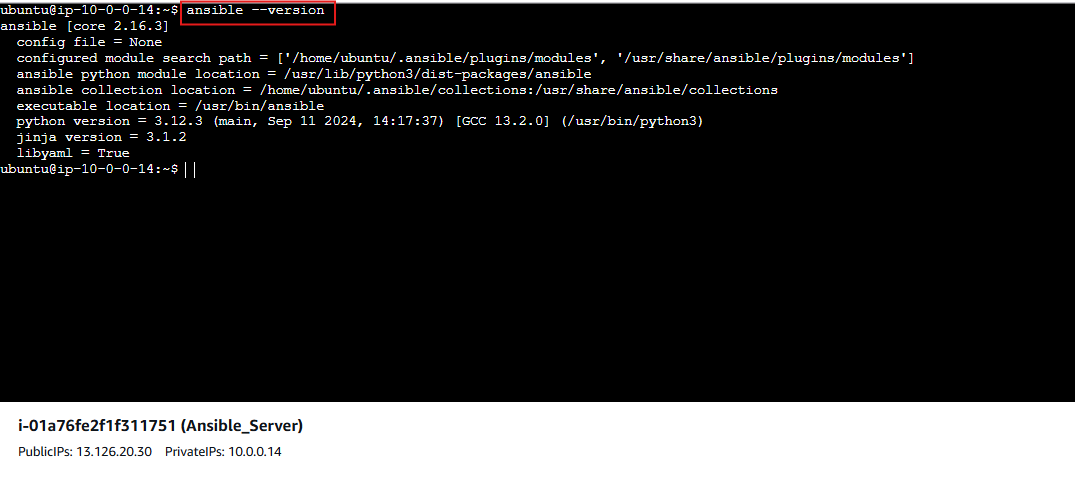
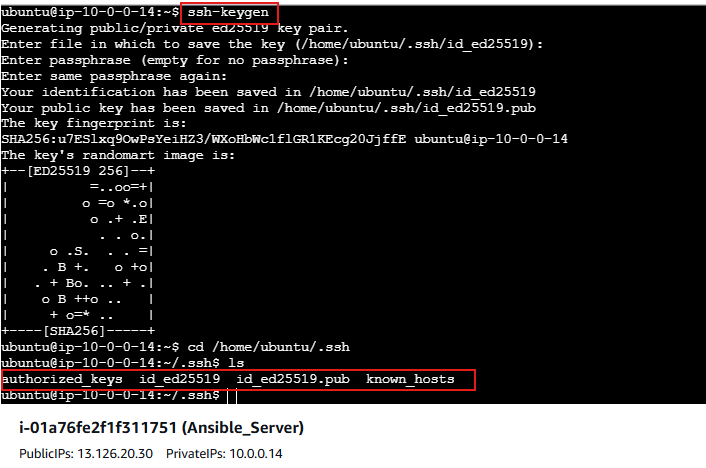
Created Ansible and Target server in EC2: Ansible Server:

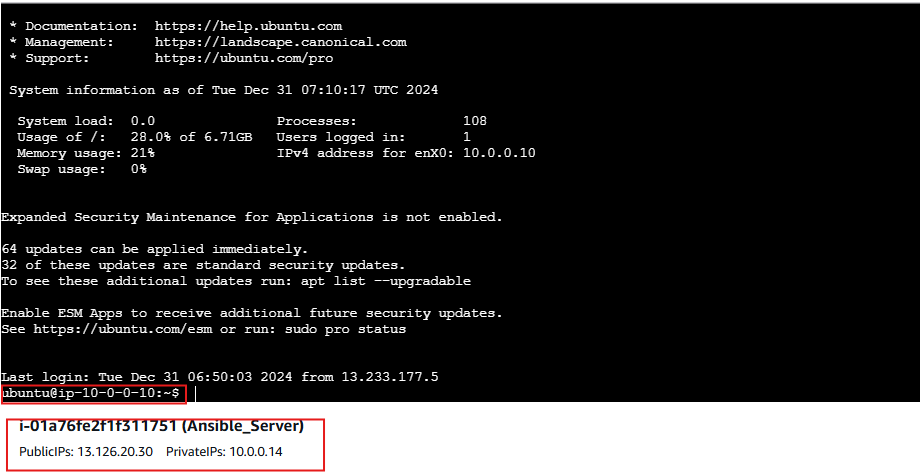
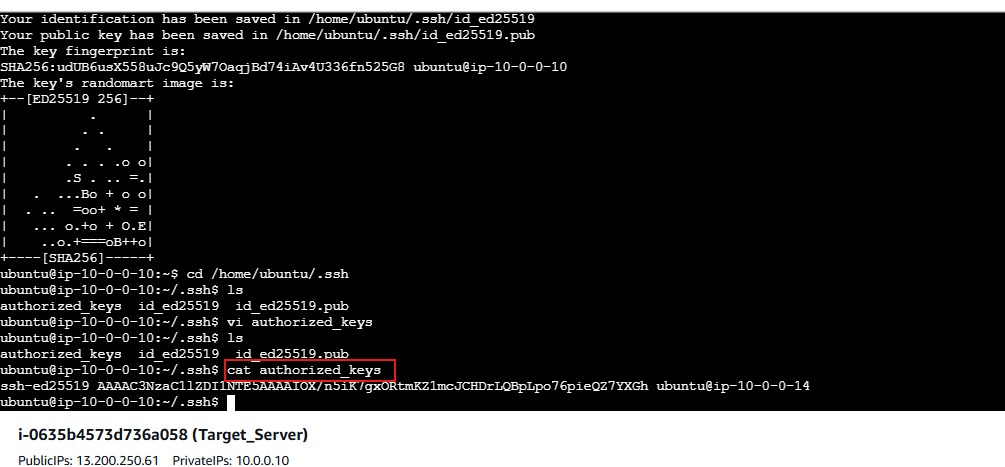
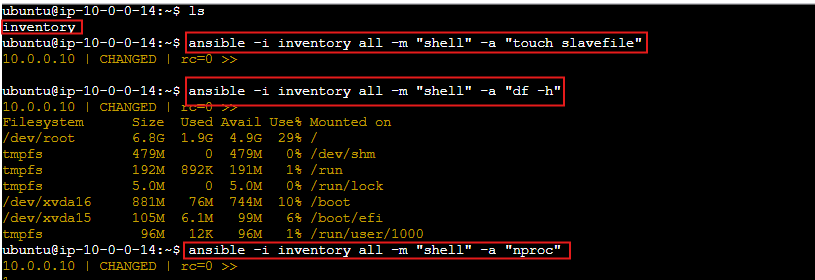
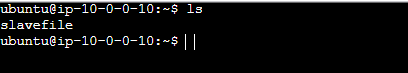


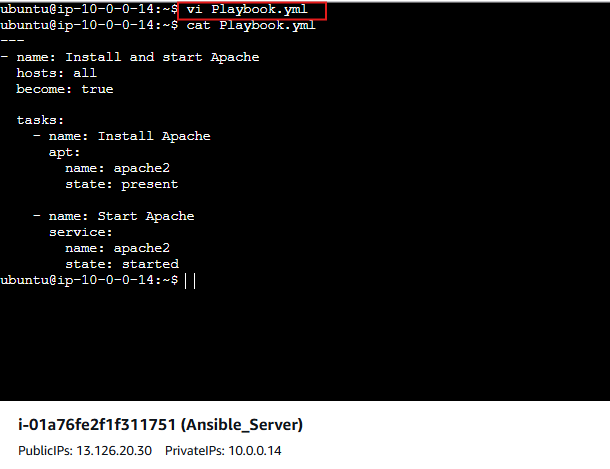
TARGET SERVER:

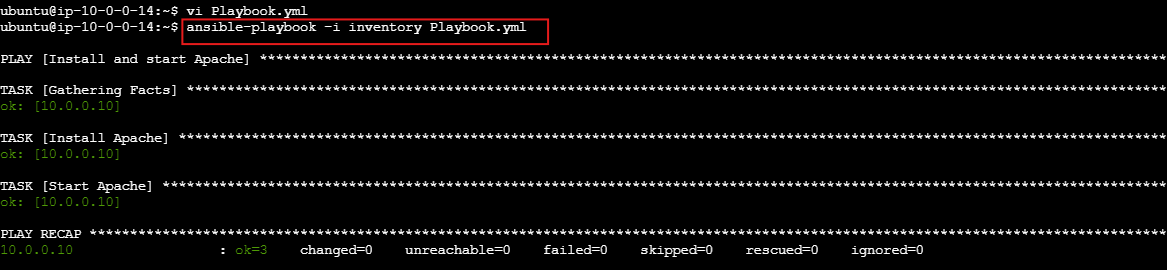


Installed Ansible in Master server:

Using Passwordless Authentication, to access Target server in Master server.

Created Inventory file in Ansible server.****File created in Ansible server using inventory command, reflected in Target server.****

****

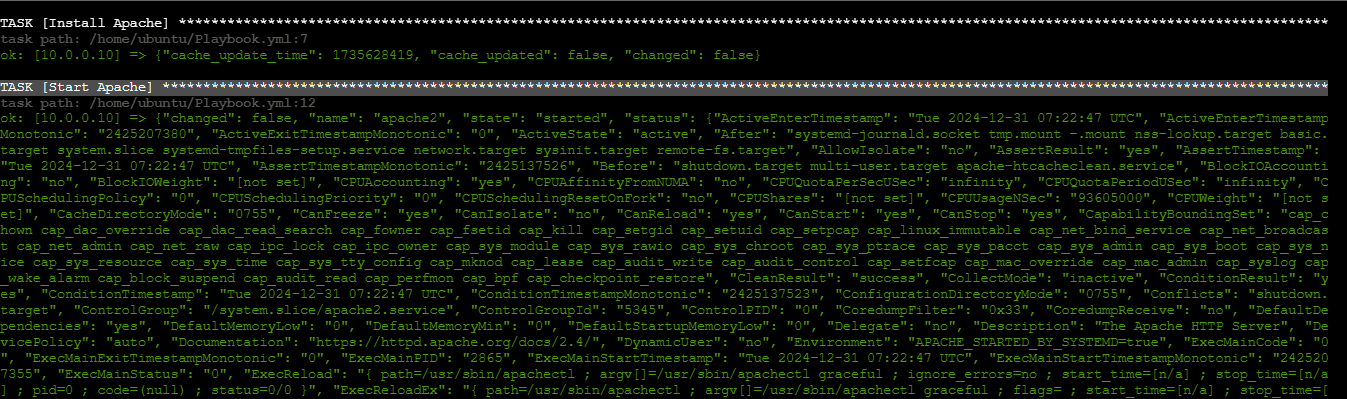
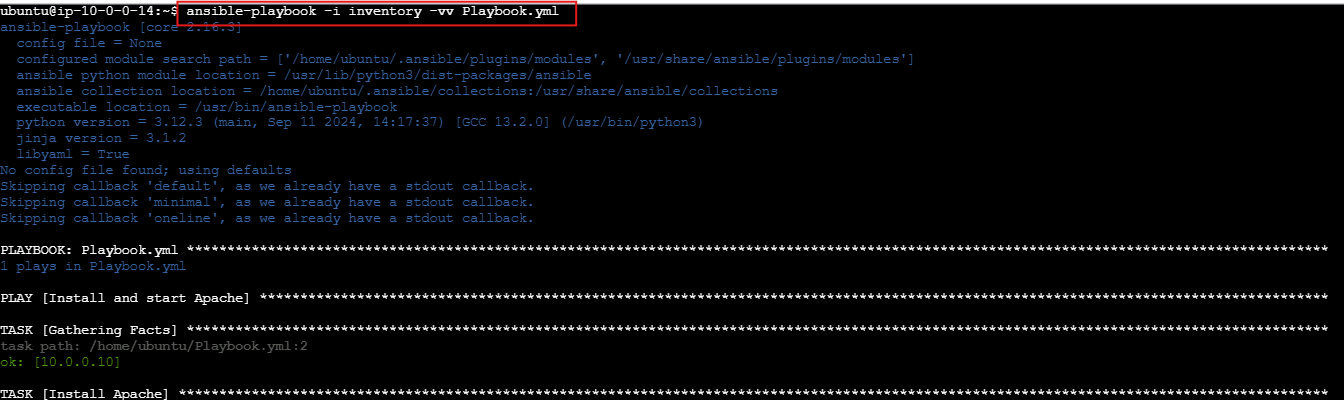
**Executed inventory file using playbook.**

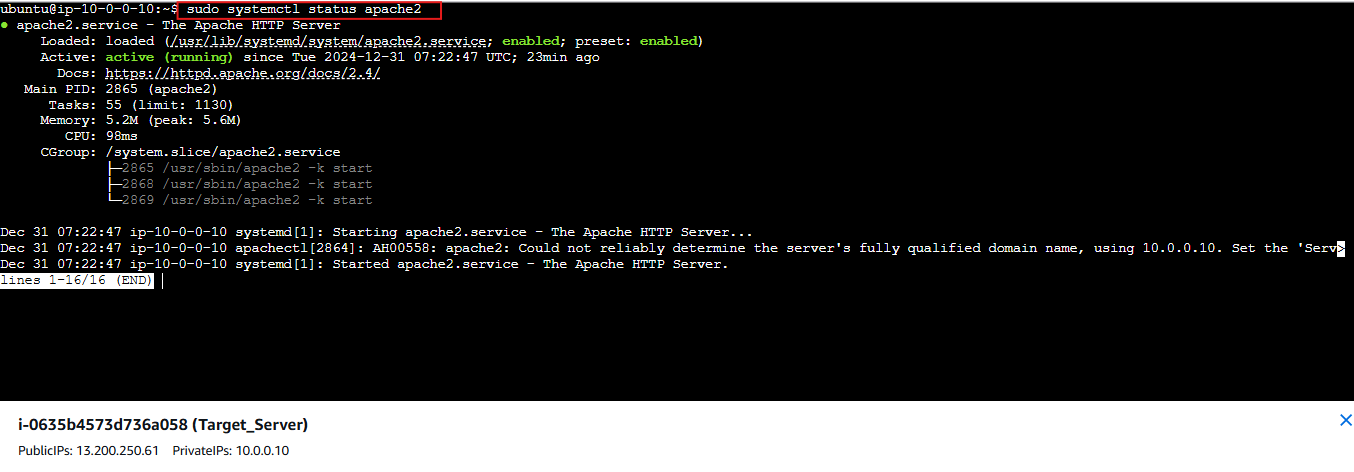
**Generated Verbosity report:**

In Ansible, verbosity refers to the level of detail included in the output during the execution of Ansible commands. Verbosity is useful for debugging and understanding what happens during a playbook execution or any Ansible command. You can control the verbosity level by adding the -v options to your command.

**Verbosity Levels in Ansible:**

1. **-v (Verbose):** Provides basic information about the execution. Useful for seeing more than the default output but not overwhelming.
2. **-vv (More Verbose):** Gives additional details, including tasks being executed and intermediate steps.
3. **-vvv (Debug Level):** Outputs detailed information, including variable values and other internal details.
4. **-vvvv (Connection Debugging):** Extremely verbose, includes all connection details and raw SSH output, useful for debugging connection issues.

****

**Apache server is running in target server.**