Ansible Project-1

This Ansible playbook is designed to automate the installation and setup of Java, Docker, SonarQube, and Trivy on target hosts. The playbook uses a series of tasks to install the required software, configure Docker permissions, and deploy a SonarQube container. It also installs Trivy, a vulnerability scanner, and configures it through repository setup.

Here's a breakdown of what each task does in detail:

1. Overview of the Project

- **Goal:** Automate the installation of the following tools on the target hosts:
    - **Java (OpenJDK 17):** Required for running Java-based applications like SonarQube.
    - **Docker:** A containerization platform that enables running applications in containers.
    - **SonarQube (Community Edition):** A popular tool for continuous inspection of code quality.
    - **Trivy:** An open-source security scanner for Docker images, filesystems, and Git repositories.
- **Hosts:** The playbook is designed to be run on **all** hosts defined in the Ansible inventory file.
- **Privilege Escalation:** The become: yes directive ensures that all tasks requiring elevated privileges are executed using sudo on the target machines.

```
---
- name: Install Java, Docker, SonarQube, and Trivy
  hosts: all
  become: yes
  tasks:
    - name: update repo
      command: sudo apt update

    - name: Install OpenJDK 17
      apt:
        name: openjdk-17-jre-headless
        state: present

    - name: Install Docker
      apt:
        name: docker.io
        state: present

    - name: Set permissions for Docker socket
      command: chmod 666 /var/run/docker.sock
      become: true

    - name: Run SonarQube container
      command: docker run -d -p 9000:9000 sonarqube:lts-community
```

```
    become: true

  - name: Install dependencies for Trivy
    apt:
      name: "{{ item }}"
      state: present
    loop:
      - wget
      - apt-transport-https
      - gnupg
      - lsb-release

  - name: Add GPG key for Trivy
    shell: wget -qO - https://aquasecurity.github.io/trivy-repo/deb/public.key | gpg --dearmor | sudo tee /usr/share/keyrings/trivy.gpg > /dev/null
    become: true

  - name: Add Trivy repository
    shell: echo "deb [signed-by=/usr/share/keyrings/trivy.gpg] https://aquasecurity.github.io/trivy-repo/deb $(lsb_release -sc) main" | sudo tee -a /etc/apt/sources.list.d/trivy.list
    become: true

  - name: Update apt cache
    apt:
      update_cache: yes

  - name: Install Trivy
    apt:
      name: trivy
      state: present
```

Playbook Breakdown

Playbook Header

```
- name: Install Java, Docker, SonarQube, and Trivy
  hosts: all
  become: yes
```

This defines the Ansible playbook's name and the target hosts for execution.

- **Name**: The name of this playbook is "Install Java, Docker, SonarQube, and Trivy."
- **Hosts**: The playbook runs on all target hosts (hosts: all).
- **Become**: The become: yes directive ensures that tasks are executed with elevated privileges (sudo), which is necessary for installing system packages and running commands that require root access.

Tasks Section

1. Update Repository

```
- name: update repo
  command: sudo apt update
```

- **Description**: Updates the package repository cache using the apt update command.
- **Purpose**: Ensures the latest package information is available for installations.

---

2. Install OpenJDK 17

```
- name: Install OpenJDK 17
  apt:
    name: openjdk-17-jre-headless
    state: present
```

- **Description**: Installs the OpenJDK 17 runtime environment using the apt module.
- **Package**: openjdk-17-jre-headless
- **Purpose**: Java is required for running applications such as SonarQube

---

3. Install Docker

```
- name: Install Docker
  apt:
    name: docker.io
    state: present
```

- **Description**: Installs Docker using the apt module.
- **Package**: docker.io
- **Purpose**: Docker is necessary to run containerized applications such as SonarQube.

---

4. Set Permissions for Docker Socket

```
- name: Set permissions for Docker socket
  command: chmod 666 /var/run/docker.sock
  become: true
```

- **Description**: Sets read and write permissions (chmod 666) on the Docker socket to allow non-root users to run Docker commands.
- **Command**: chmod 666 /var/run/docker.sock
- **Purpose**: Enables Docker to run without needing root privileges.

---

5. Run SonarQube Container

```
- name: Run SonarQube container
  command: docker run -d -p 9000:9000 sonarqube:lts-community
  become: true
```

- **Description**: Starts a SonarQube container in detached mode.
- **Command**: docker run -d -p 9000:9000 sonarqube:lts-community
- **Purpose**: Runs SonarQube, which is a tool for static code analysis and code quality management.
- **Port Mapping**: Maps port 9000 on the host to 9000 on the container.

---

6. Install Dependencies for Trivy

```
- name: Install dependencies for Trivy
  apt:
    name: "{{ item }}"
    state: present
```

```
  loop:
    - wget
    - apt-transport-https
    - gnupg
    - lsb-release
```

- **Description**: Installs the required dependencies for Trivy using the apt module.
- **Packages**: Loops through the list of dependencies (wget, apt-transport-https, gnupg, lsb-release) to install each one.
- **Purpose**: These dependencies are necessary for setting up the Trivy repository and running Trivy.

---

7. Add GPG Key for Trivy

```
- name: Add GPG key for Trivy
  shell: wget -qO - https://aquasecurity.github.io/trivy-repo/deb/public.key | gpg --dearmor | sudo tee /usr/share/keyrings/trivy.gpg > /dev/null
  become: true
```

- **Description**: Downloads and adds the GPG key for the Trivy repository.
- **Command**: The wget command fetches the GPG key, and the gpg command adds it to the keyring /usr/share/keyrings/trivy.gpg.
- **Purpose**: Ensures secure installation of Trivy by verifying the packages using the GPG key.

8. Add Trivy Repository

```
- name: Add Trivy repository
  shell: echo "deb [signed-by=/usr/share/keyrings/trivy.gpg] https://aquasecurity.github.io/trivy-repo/deb $(lsb_release -sc) main" | sudo tee -a /etc/apt/sources.list.d/trivy.list
  become: true
```

- **Description**: Adds the Trivy package repository to the system's sources list.
- **Command**: Appends the repository configuration to /etc/apt/sources.list.d/trivy.list.
- **Purpose**: Adds the Trivy repository to the system, making it possible to install Trivy from this source.

9. Update Apt Cache

```
- name: Update apt cache
  apt:
    update_cache: yes
```

- **Description**: Updates the local package cache.
- **Purpose**: Ensures that the system is aware of the newly added Trivy repository.

10. Install Trivy

```
- name: Install Trivy
  apt:
    name: trivy
    state: present
```

- **Description**: Installs Trivy, a vulnerability scanner for containers, using the apt module.
- **Package**: trivy
- **Purpose**: Trivy is used for scanning container images for vulnerabilities.