**Connect to Ansible playbook to docker**

**First step is to setup Docker and pre-requisite**

**1.**[**https://github.com/Sumanth17-git/APMTrianing**](https://github.com/Sumanth17-git/APMTrianing)**/**[**setup\_debian\_server.sh**](https://github.com/Sumanth17-git/APMTrianing/blob/main/setup_debian_server.sh)

**2.**

--Allow ansible user to access Docker --

sudo usermod -aG docker $USER

id $USER

sudo systemctl restart docker

sudo cat /etc/docker/daemon.json

onfigure Docker Socket Permissions Permanently

Create a daemon.json Configuration Add or edit the Docker configuration file at /etc/docker/daemon.json

sudo nano /etc/docker/daemon.json

{

"group": "docker"

}

sudo systemctl restart docker

**3.**

**Ansible -Target ( Ubuntu Machine)**

--Create ansible user (you can also use root user , best practice is to use separate user , i.e ansible user)

**# Create Ansible user and set up sudo permissions**

**sudo adduser --disabled-password --gecos "" ansible**

**sudo usermod -aG sudo ansible**

**vi /etc/sudoers**

**ansible ALL=(ALL) NOPASSWD: ALL**

**sudo usermod -aG sudo ansible**

**su - ansible**

**mkdir -p ~/.ssh**

**chmod 700 ~/.ssh**

**touch ~/.ssh/authorized\_keys**

**chmod 600 ~/.ssh/authorized\_keys**

**sudo chmod 700 /home/ansible/.ssh**

**sudo chmod 600 /home/ansible/.ssh/authorized\_keys**

**sudo chown -R ansible:ansible /home/ansible/.ssh**

**sudo vi /etc/ssh/sshd\_config**

**Make sure the following lines are set (uncomment them if necessary):**

**PubkeyAuthentication yes**

**AuthorizedKeysFile .ssh/authorized\_keys**

**PubkeyAuthentication yes**

**sudo systemctl restart ssh**

**vi ~/.ssh/authorized\_keys**

**--Paste the public key from control vm into the file and save it.--**

**4.**

**sudo systemctl restart ssh**

**ls -ld ~/.ssh**

**ls -l ~/.ssh/authorized\_keys**

**cat ~/.ssh/authorized\_keys**

**------Validation from Control------**

**ssh ansible@10.128.0.20**

**Type 'Yes'**

**ssh-copy-id** [**ansible@10.150.0.22**](mailto:ansible@10.150.0.22)

**5.**

**Create inventory.yml or using existing host**

all:

hosts:

vm2:

ansible\_host: 10.128.0.56

ansible\_user: ansible # Replace 'user' with the SSH username for vm2

**ansible -i inventory.yml all -m ping**

**sudo vi /etc/ansible/hosts**

[mytargets]

vm2 ansible\_host=10.128.0.52 ansible\_user=user

**ansible -i /etc/ansible/hosts mytargets -m ping**

**This confirms everything is in places to start the docker container :**

6.

docker container rm fastapi-observability

docker container rm springboot

docker container rm nodemetrics

docker run -d -p 8001:8000 --name fastapi-observability ghcr.io/blueswen/fastapi-observability/app:latest

docker run -d -p 8881:8881 --name springboot sumanth17121988/springbootmetric:1

docker run -d -p 9991:9991 --name nodemetrics sumanth17121988/node-metrics-app:1

--

docker container restart fastapi-observability

docker container restart springboot

docker container restart nodemetrics

7.

To use Docker modules in Ansible, you need to install the community.docker collection from Ansible Galaxy. This collection provides modules like docker\_container, docker\_image, and more, which are essential for managing Docker in your playbooks.

**ansible-galaxy collection install community.docker**

**ansible-galaxy collection list**

check docker is mentioned there

8.

vi restart\_docker.yml

---

- name: Restart Docker container on vm2

hosts: vm2

become: true # Escalate privileges to root if necessary

tasks:

- name: Restart the Docker container using its name

community.docker.docker\_container:

name: nodemetrics # Container name

state: started

restart: yes # Ensures the container is restarted if already running

To pass the container name dynamically, you can use Ansible variables. Here's how to update your playbook and pass the container name as an extra variable when running the playbook.

vi restart\_docker\_dynamic.yml

---

- name: Restart Docker container dynamically

hosts: vm2

become: true # Escalate privileges to root if necessary

vars:

container\_name: "{{ container\_name }}" # Dynamically passed variable

tasks:

- name: Restart the Docker container

community.docker.docker\_container:

name: "{{ container\_name }}"

state: started

restart: yes

**ansible-playbook -i inventory.yml restart\_docker\_dynamic.yml --extra-vars "container\_name=fastapi-observability"**

**To Run it from Jenkins**

**Create a Sudoers File in /etc/sudoers.d/**

**sudo vi /etc/sudoers.d/jenkins**

jenkins ALL=(ansible) NOPASSWD: ALL

**Test Your Configuration**

sudo su – jenkins

sudo -u ansible whoami

**The output should be:**

ansible