I have finally managed to install Docker CE on a freshly CentOS 8.1 installed dedicated server. Now is the time to learn Docker module in Ansible.

Why Use a Docker Module in Ansible?

Instead of using Ansible’s shell or command functionality to form/start/stop Docker containers, it’s much better to use a Docker module.

How To Install the Docker Module in Ansible

For CentOS 8.x images, here’s the Ansible playbook fragment that gets the Docker module installed on a server:

```yaml
name: Install pip3
yum:
```
name: python3-pip
state: latest
tags: docker
name: Install Docker module for Python (used by Ansible)
pip:
  name:
    - docker

How To Use a Docker Module in Ansible

Here’s the example from Ansible documentation:

- name: Create a simple Docker container
docker_container:
  name: unixtutorial-server
  image: ubuntu
  volumes:
    - /storage/ubuntu-docker
tags:
  - docker
  - containers

In this example you can see how easy it is to specify a Docker container’s name, base image (busybox in this case) and even map volumes from physical host (that /storage thing).

WARNING: this is not a complete example yet. I think more parameters need to be specified before this becomes a usable container named unixtutorial-server.
Next Steps with Docker in Ansible

I have a number of Docker images that I plan on storing using self-hosted Docker registry. They’ll be pulled and formed into clusters using Ansible.

I’m not 100% sure, but it may well be that whole Dockerfiles (read: Docker stacks) can be created and orchestrated using Ansible — we’ll see.

See Also

- Ansible
- Docker for Mac
- Tagging Docker Images