**6. Ansible docker swarm deployment**

--- **cat docker-swarm-inventroy**

[docker\_servers]

Docker-swarm-master-0        ansible\_host=34.224.97.34   ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

Docker-swarm-master-1        ansible\_host=3.88.166.93    ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

Docker-swarm-master-2        ansible\_host=52.87.195.66   ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

Docker-swarm-workder-nodes-1 ansible\_host=54.242.94.226  ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

Docker-swarm-workder-nodes-2 ansible\_host=54.166.160.207 ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

Docker-swarm-workder-nodes-3 ansible\_host=3.91.3.243     ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

[docker\_master]

Docker-swarm-master-0        ansible\_host=34.224.97.34   ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

[docker\_managers]

Docker-swarm-master-1        ansible\_host=3.88.166.93    ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

Docker-swarm-master-2        ansible\_host=52.87.195.66   ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

[docker\_workers]

Docker-swarm-workder-nodes-1 ansible\_host=54.242.94.226  ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

Docker-swarm-workder-nodes-2 ansible\_host=54.166.160.207 ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

Docker-swarm-workder-nodes-3 ansible\_host=3.91.3.243     ansible\_connection=ssh     ansible\_user=ansadmin   ansible\_port=22

--- cat docker-swarm-deploy.yml

---

  - name: Install Docker and Configure Docker Swarm

    hosts: docker\_servers

    become: yes

    become\_user: root

    tasks:

      - name: Install Docker on all docker\_servers

        shell: curl https://get.docker.com | bash

      - name: Check Docker Version

        shell: docker version | grep -w Version | head -1

        register: version

      - debug:

          var: version

    tags:

     - install

  - name: Enable Docker Swarm

    hosts: docker\_master

    become: yes

    become\_user: root

    tasks:

      - name: Enable Docker Swarm on Master docker\_servers

        shell: docker swarm init

        ignore\_errors: yes

      - name: Get Docker Worker Token

        shell: docker swarm join-token -q worker

        register: token

      - set\_fact:

           swarm\_token: "{{ token.stdout }}"

      - debug:

          var: token.stdout

      - name: Get Docker Manager Token

        shell: docker swarm join-token -q manager

        register: managertoken

      - set\_fact:

           swarmmanager\_token: "{{ managertoken.stdout }}"

      - debug:

          var: swarmmanager\_token.stdout

      - name: Get Docker Master Private IP

        shell: curl http://169.254.169.254/latest/meta-data/local-ipv4/

        register: private\_ip

      - set\_fact:

           swarm\_ip: "{{ private\_ip.stdout }}"

      - debug:

          var: private\_ip.stdout

      - name: add variables to dummy host 1

        add\_host:

         name: "docker\_master\_node\_token"

         shared\_variable:  "{{ swarm\_token }}"

      - name: add variables to dummy host 3

        add\_host:

         name: "docker\_master\_node\_ip"

         shared\_variable:  "{{ swarm\_ip }}"

      - name: add variables to dummy host 4

        add\_host:

         name: "docker\_master\_managernode\_token"

         shared\_variable:  "{{ swarmmanager\_token }}"

    tags:

     - swarm

  - name: Add Workers to Swarm

    hosts: docker\_workers

    become: yes

    become\_user: root

    vars:

      private\_ip: "{{ hostvars['docker\_master\_node\_ip']['shared\_variable'] }}"

      token: "{{ hostvars['docker\_master\_node\_token']['shared\_variable'] }}"

    tasks:

      - debug:

          var: token

      - debug:

          var: private\_ip

      - name: if it is already part of a swarm then leave.

        shell: docker swarm leave

        ignore\_errors: yes

      - name: Add Workers to Swarm

        shell: docker swarm join --token "{{ token }}" "{{ private\_ip }}":2377

        ignore\_errors: yes

    tags:

     - workers

  - name: Add Managers to Swarm

    hosts: docker\_managers

    become: yes

    become\_user: root

    vars:

      private\_ip: "{{ hostvars['docker\_master\_node\_ip']['shared\_variable'] }}"

      token: "{{ hostvars['docker\_master\_managernode\_token']['shared\_variable'] }}"

    tasks:

      - debug:

          var: token

      - debug:

          var: private\_ip

      - name: if it is already part of a swarm then leave.

        shell: docker swarm leave

        ignore\_errors: yes

      - name: Add Managers to Swarm

        shell: docker swarm join --token "{{ token }}" "{{ private\_ip }}":2377

        ignore\_errors: yes

    tags:

     - managers

  - name: Deploy Test Application

    hosts: docker\_master

    become: yes

    become\_user: root

    vars:

      private\_ip: "{{ hostvars['docker\_master\_node\_ip']['shared\_variable'] }}"

      token: "{{ hostvars['docker\_master\_managernode\_token']['shared\_variable'] }}"

    tasks:

      - debug:

          var: token

      - debug:

          var: private\_ip

      - name: Deploy Sample Application

        shell: docker service rm nginx002 && docker service create --name nginx002 -p 8000:80 --replicas 6 sreeharshav/rollingupdate:v5

        ignore\_errors: yes

      - name: Validate Deployment

        shell: sleep 10 && curl http://"{{ private\_ip.stdout }}":8000

        register: html

        ignore\_errors: yes

      - debug:

         var: html.stdout

    tags:

     - managers

**# Execute the playbook**

--- ansible-playbook -i docker-swarm-inventory docker-swarm-deploy.yml --extra-vars "ansible\_sudo\_pass=ansadmin"

**Ansible docker swarm module**

--- **note** – currently we are using docker shell for doing deployment. We have other option to do deployment that is docker swarm module.

--- Reference - <https://docs.ansible.com/ansible/2.9/modules/docker_swarm_module.html>

**Requirements on ansible host**

**# Docker API >= 1.25**

--- curl <https://get.docker.com> | bash

# Docker SDK for Python: Please note that the docker-py Python module has been superseded by docker (see here for details). For Python 2.6, docker-py must be used. Otherwise, it is recommended to install the docker Python module. Note that both modules should not be installed at the same time. Also note that when both modules are installed and one of them is uninstalled, the other might no longer function and a reinstall of it is required.

--- **Reference** - <https://docker-py.readthedocs.io/en/stable/>