Integration of docker and ansible of RH-294 -Linux Automation

Ansible(RH294) Task 1:

Write an Ansible playbook that does the following operations in the managed nodes:

- Configure Docker.
- Start and enable Docker services.
- Pull the httpd server image from the Docker Hub.
- Run the httpd container and expose it to the public.
- Copy the html code in /var/www/html directory and start the webserver.

About Ansible:

Ansible is an open-source software provisioning, configuration management, and application-deployment tool enabling infrastructure as code. It runs on many Unix-like systems, and can configure both Unix-like systems as well as Microsoft Windows. It includes its own declarative language to describe system configuration. Ansible was written by Michael DeHaan and acquired by Red Hat in 2015. Ansible is agentless, temporarily connecting remotely via SSH or Windows Remote Management (allowing remote PowerShell execution) to do its tasks.

About Docker:

Docker is a set of the platform as service products that use OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries, and configuration files; they can communicate with each other through well-defined channels.

What is httpd?

The Apache HTTP Server, colloquially called Apache, is a Web server application notable for playing a key role in the initial growth of the World Wide Web. Originally based on the NCSA HTTPd server, the development of Apache began in early 1995 after work on the NCSA code stalled. Apache quickly overtook NCSA HTTPd as the dominant HTTP server and has remained the most popular HTTP server in use since April 1996.

Ansible playbook:

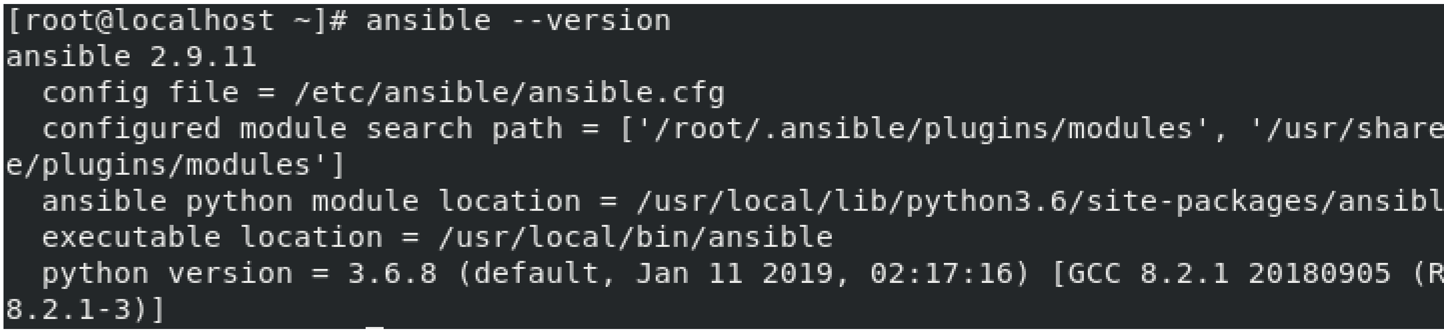
An Ansible playbook is an organized unit of scripts that defines work for a server configuration managed by the automation tool Ansible. Ansible is a configuration management tool that automates the configuration of multiple servers by the use of Ansible playbooks.

Pre-requisites:

~ Installation of Ansible:

As Ansible is made on top of python language ,so we have to install ansible using following command: # pip3 install ansible

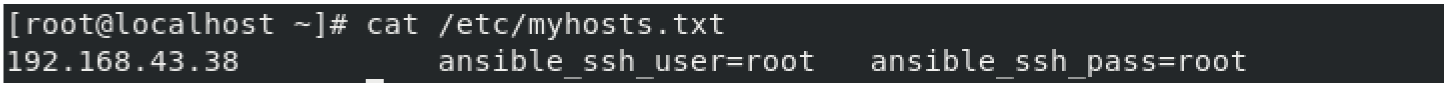
for checking version of installed ansible tool we can use following command.



```
[root@localhost ~]# ansible --version
ansible 2.9.11
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share
e/plugins/modules']
  ansible python module location = /usr/local/lib/python3.6/site-packages/ansibl
  executable location = /usr/local/bin/ansible
  python version = 3.6.8 (default, Jan 11 2019, 02:17:16) [GCC 8.2.1 20180905 (R
8.2.1-3)]
```

~ Configuration of Ansible:

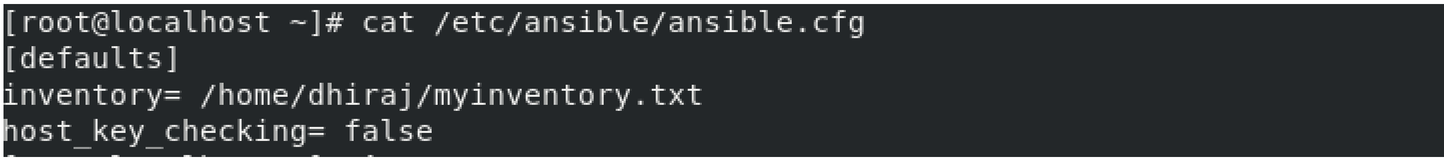
Next, we have to create one text file at /etc/myhost.txt which store IP of managed node along with username and password.



```
[root@localhost ~]# cat /etc/myhosts.txt
192.168.43.38           ansible_ssh_user=root   ansible_ssh_pass=root
```

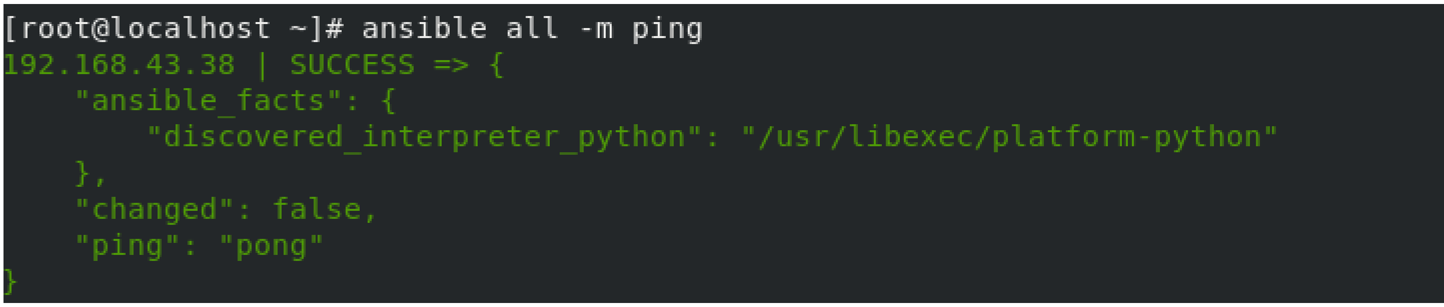
Ansible works against multiple systems in your infrastructure at the same time. It does this by selecting portions of the systems listed in Ansible's inventory file. You can specify a different inventory file using the -i <path> option on the command line. Not only is this inventory configurable, but you can also use multiple inventory files at the same time.

To check list of all host use following command:



```
[root@localhost ~]# cat /etc/ansible/ansible.cfg
[defaults]
inventory= /home/dhiraj/myinventory.txt
host_key_checking= false
```

We need to check, that all managed node is properly connected to the controller node or not. For that use the following command.

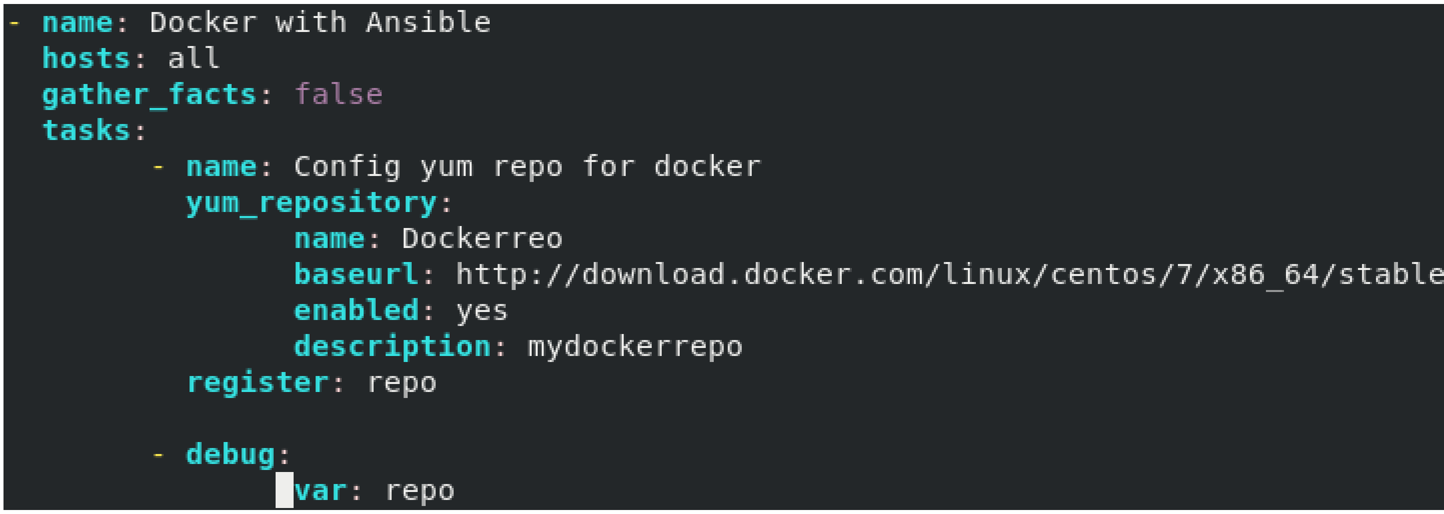


```
[root@localhost ~]# ansible all -m ping
192.168.43.38 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/libexec/platform-python"
    },
    "changed": false,
    "ping": "pong"
}
```

As our setup of ansible is completed. Next,we have to create ansible playbook in YAML language. Here, I have created docker.yml playbook file.

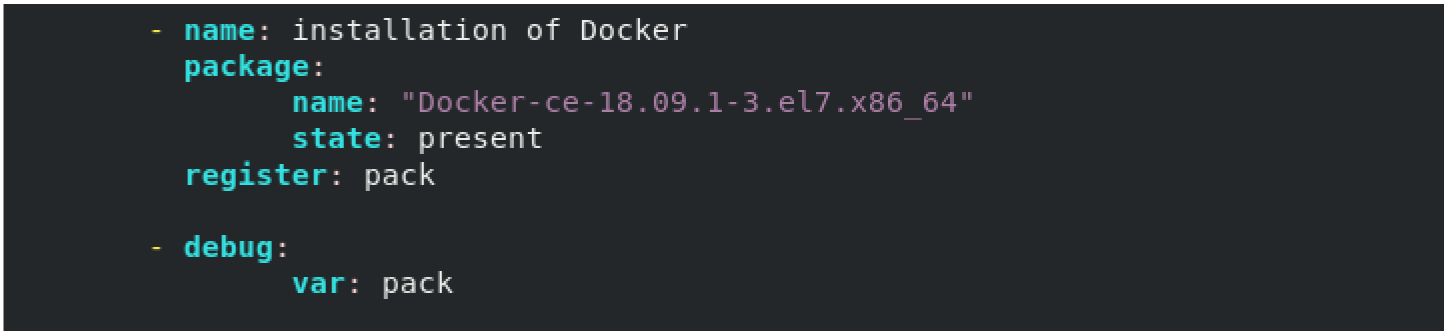
STEPS TO BE FOLLOW:

STEP 1 : <u>Configuration of yum repository</u>



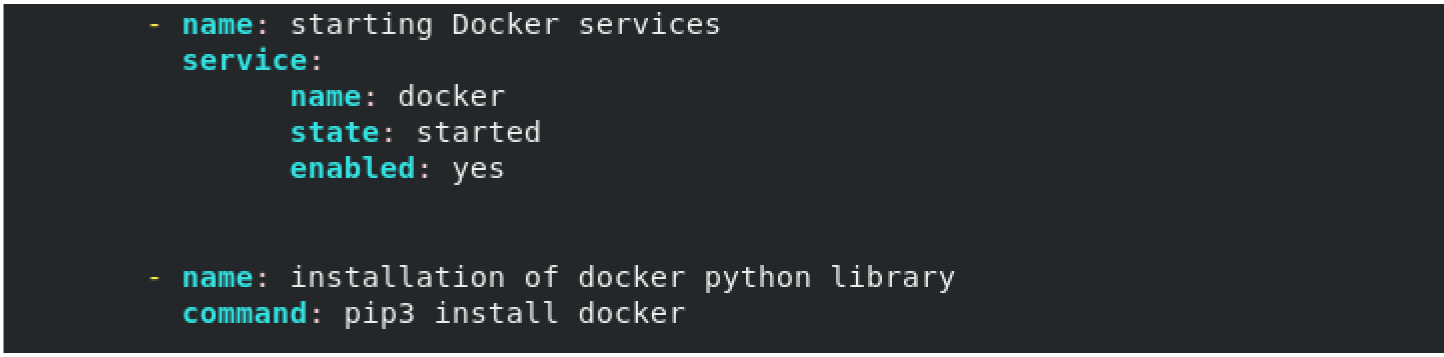
```
- name: Docker with Ansible
  hosts: all
  gather_facts: false
  tasks:
        - name: Config yum repo for docker
          yum_repository:
                name: Dockerreo
                baseurl: http://download.docker.com/linux/centos/7/x86_64/stable
                enabled: yes
                description: mydockerrepo
          register: repo

        - debug:
                var: repo
```

STEP 2 : Installing docker using package module:



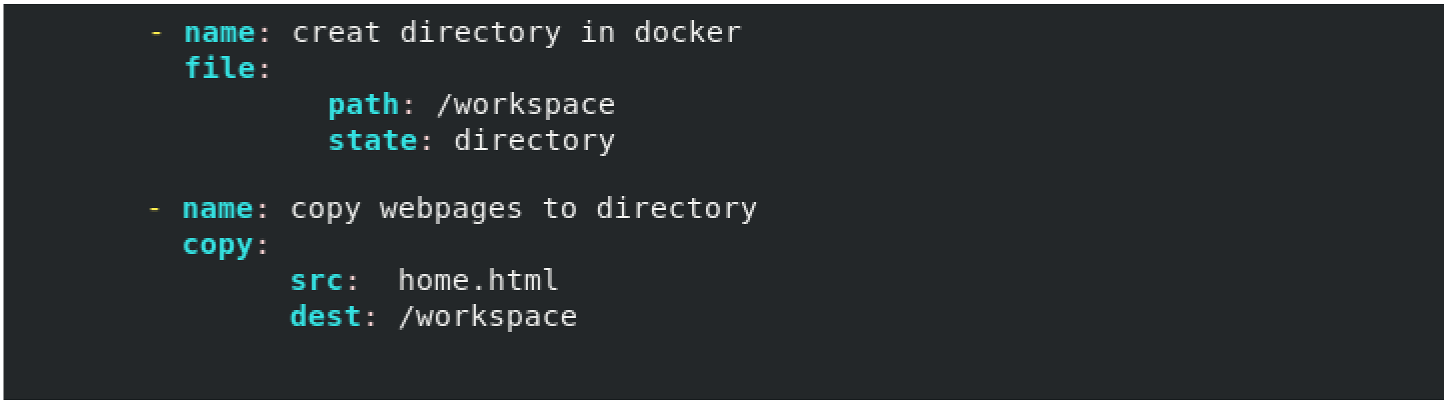
```
        - name: installation of Docker
          package:
                name: "Docker-ce-18.09.1-3.el7.x86_64"
                state: present
          register: pack

        - debug:
                var: pack
```

STEP 3:Then, start and enable the docker service and install the python for the docker:



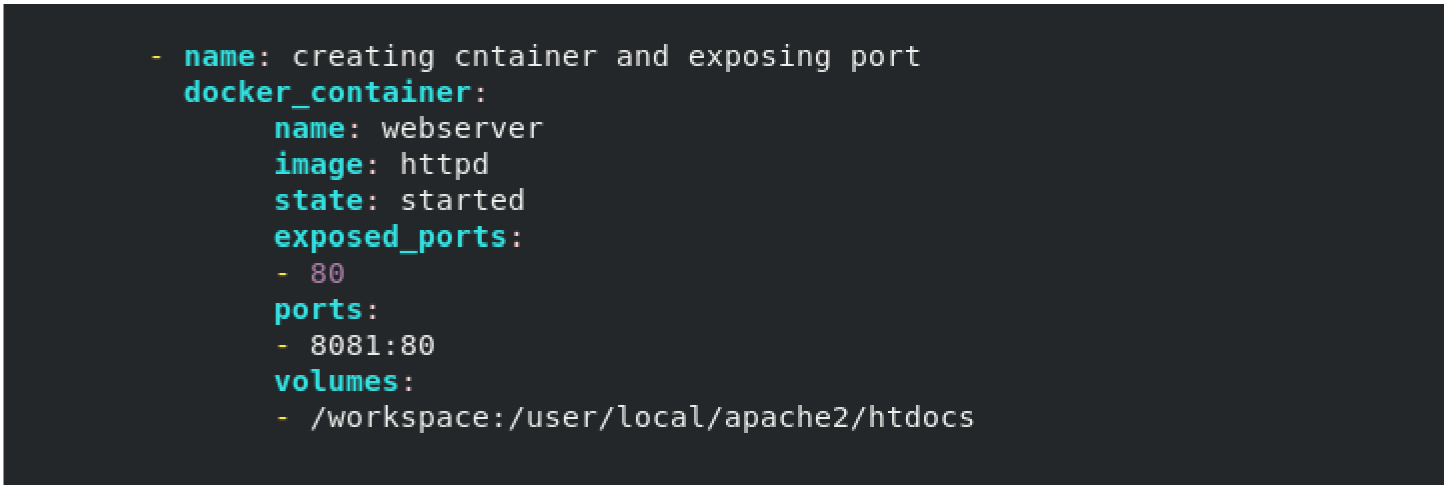
```
        - name: starting Docker services
          service:
                name: docker
                state: started
                enabled: yes


        - name: installation of docker python library
          command: pip3 install docker
```

STEP 4 : Creating the Directory and Copying the html file from the controller node to the managed node.



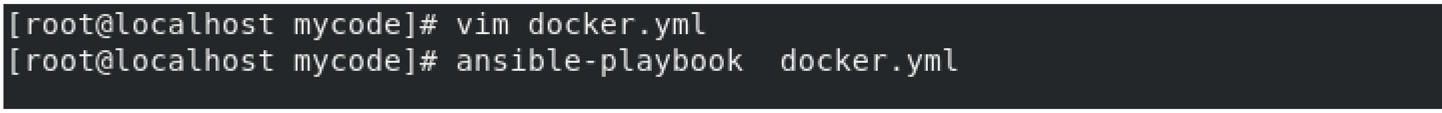
```
        - name: creat directory in docker
          file:
                  path: /workspace
                  state: directory

        - name: copy webpages to directory
          copy:
                src:  home.html
                dest: /workspace
```

STEP 5 : Creating a container using httpd image & expose the port.



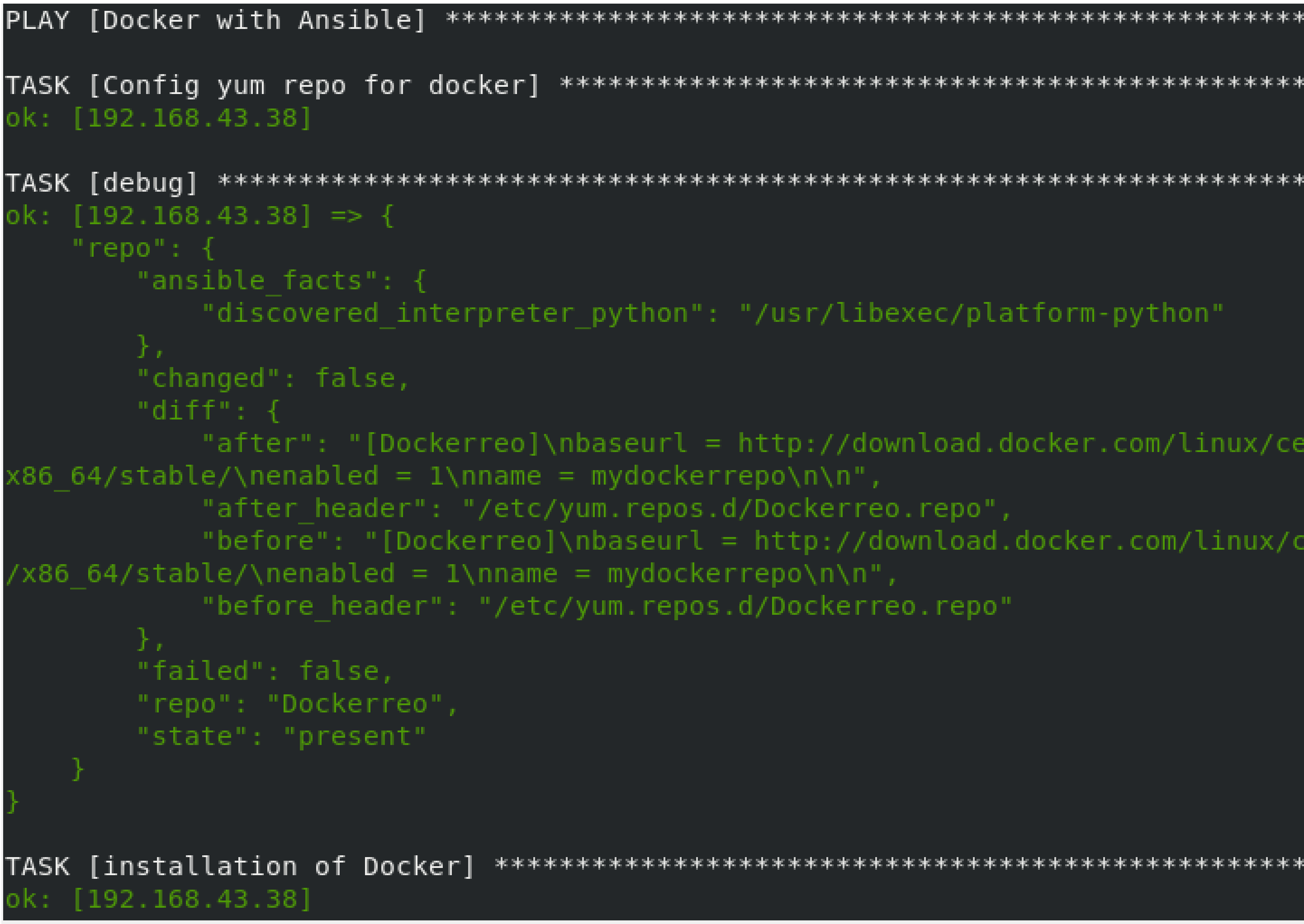
```
        - name: creating cntainer and exposing port
          docker_container:
               name: webserver
               image: httpd
               state: started
               exposed_ports:
               - 80
               ports:
               - 8081:80
               volumes:
               - /workspace:/user/local/apache2/htdocs
```

Now run the following command for the running the ansible-playbook:

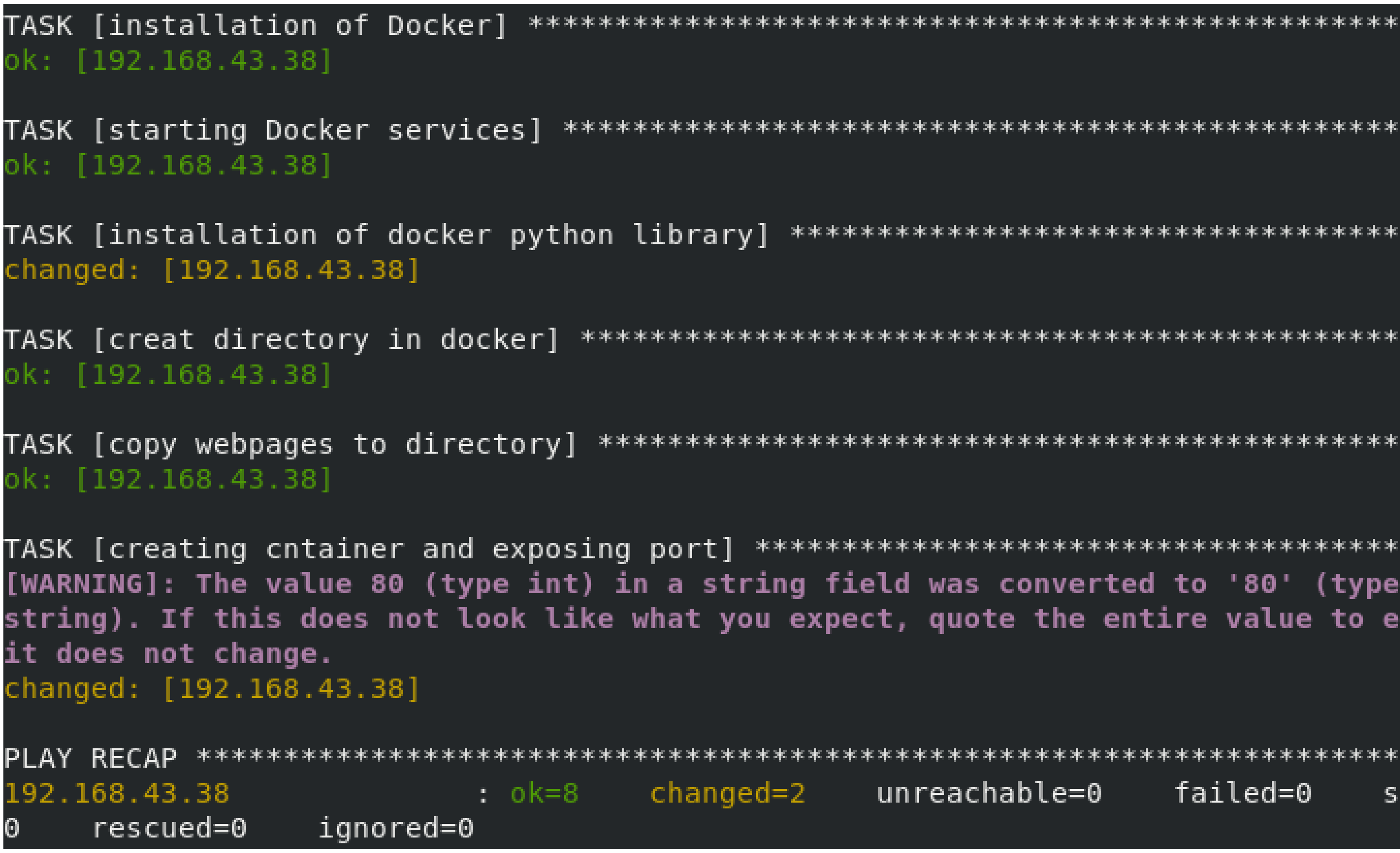


```
[root@localhost mycode]# vim docker.yml
[root@localhost mycode]# ansible-playbook  docker.yml
```

Output:



```
PLAY [Docker with Ansible] *****************************************************

TASK [Config yum repo for docker] *********************************************
ok: [192.168.43.38]

TASK [debug] *******************************************************************
ok: [192.168.43.38] => {
    "repo": {
        "ansible_facts": {
            "discovered_interpreter_python": "/usr/libexec/platform-python"
        },
        "changed": false,
        "diff": {
            "after": "[Dockerreo]\nbaseurl = http://download.docker.com/linux/ce
x86_64/stable/\nenabled = 1\nname = mydockerrepo\n\n",
            "after_header": "/etc/yum.repos.d/Dockerreo.repo",
            "before": "[Dockerreo]\nbaseurl = http://download.docker.com/linux/c
/x86_64/stable/\nenabled = 1\nname = mydockerrepo\n\n",
            "before_header": "/etc/yum.repos.d/Dockerreo.repo"
        },
        "failed": false,
        "repo": "Dockerreo",
        "state": "present"
    }
}

TASK [installation of Docker] **************************************************
ok: [192.168.43.38]
```



```
TASK [installation of Docker] **************************************************
ok: [192.168.43.38]

TASK [starting Docker services] ************************************************
ok: [192.168.43.38]

TASK [installation of docker python library] ***********************************
changed: [192.168.43.38]

TASK [creat directory in docker] ***********************************************
ok: [192.168.43.38]

TASK [copy webpages to directory] **********************************************
ok: [192.168.43.38]

TASK [creating cntainer and exposing port] *************************************
[WARNING]: The value 80 (type int) in a string field was converted to '80' (type
string). If this does not look like what you expect, quote the entire value to e
it does not change.
changed: [192.168.43.38]

PLAY RECAP *********************************************************************
192.168.43.38              : ok=8    changed=2    unreachable=0    failed=0    s
0    rescued=0    ignored=0
```