Task 11.3: Restarting HTTPD Service is not idempotence in nature and consume more resources suggest a way to rectify this challenge in Ansible playbook

We must make the playbook idempotence.

➔ What is idempotence?

- Idempotence is a term given to certain operations whereby:
- **an action which, when performed multiple times, has no further effect on its subject after the first time it is performed.**

- In general, idempotence is "the property of certain operations in mathematics and computer science that can be applied multiple times without changing the result beyond the initial application". For Ansible it means after 1 run of a playbook to set things to a desired state, further runs of the same playbook should result in 0 changes. In simplest terms, idempotency means you can be sure of a consistent state in your environment.

➔ **Problem Statement**

- **Restarting the httpd service in the Ansible playbook is not idempotence in nature and often consumes more resources to recommend a way to correct this problem. In detail, any time a change is made to its Configuration File, the HTTPD Webserver service has to be restarted. So given this fact, usually we will give the keyword restarted under state, the disadvantage of following this step is that any time we play the playbook, the service will restart even if no changes in the configuration file have been made. But we just want to restart the service when there are some configuration file changes...**
- **How to resolve this problem?**

Pre-requisite:

- **To overcome this problem, you should have the Ansible Control node configured. So, if you're not comfortable with the Ansible controller node setup, then you can visit my previous article.**
- **Link: https://www.linkedin.com/pulse/automate-docker-ansible-launching-webserver-anudeep-nalla/**

➔ **To Solve this issue, we have to use the concept of "notify & handlers" in the playbook.**

Handlers: running operations on change

➔ Sometimes you want a task to run only when a change is made on a machine. For example, you may want to restart a service if a task updates the configuration of that service, but not if the configuration is unchanged. Ansible uses handlers to address this use case. Handlers are tasks that only run when notified. Each handler should have a globally unique name.

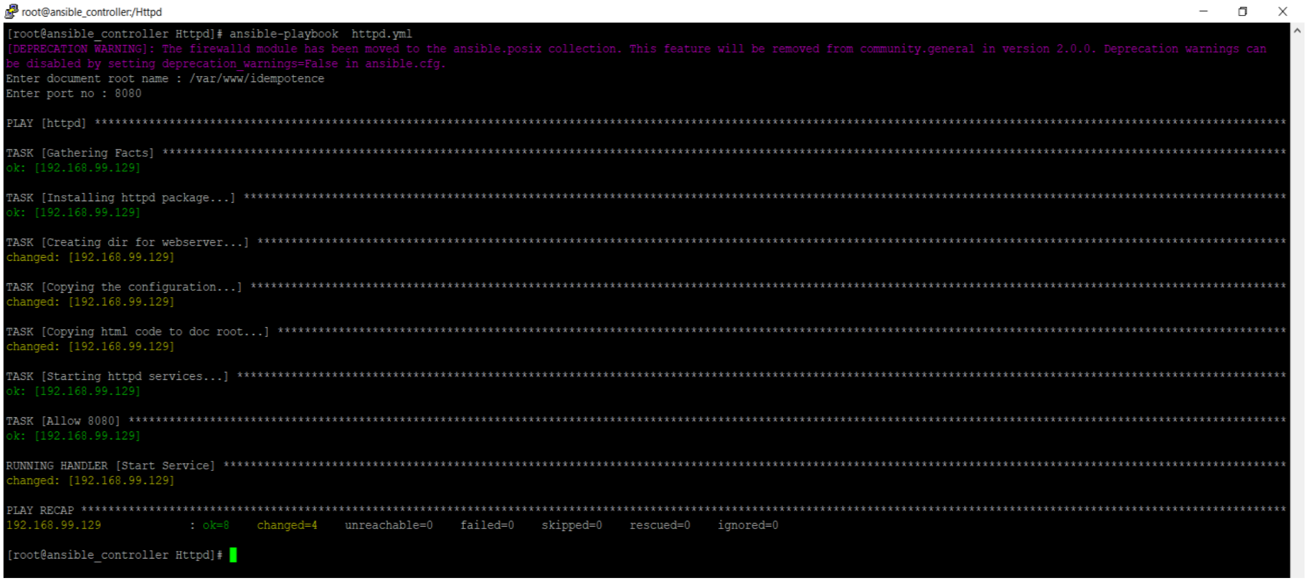
Step1: Create a configuration file for webserver with .conf extension.

```
Listen {{ http_port }}

<Virtualhost {{ ansible_facts[‘default_ipv4’][‘address’] }}:{{ http_port }}>

DocumentRoot {{ doc_root }}


</Virtualhost>
```

Step2: Create a playbook with .yml extension and start writing the below code inside it.

➔ **Playing the playbook for the first time**

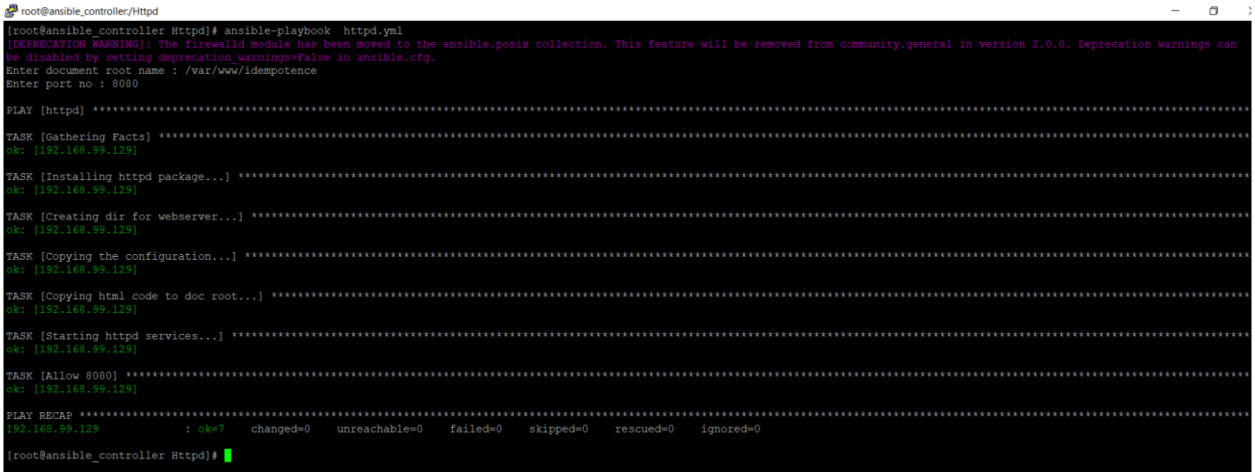
- ansible-playbook playbook_name.yml



```
[root@ansible_controller Httpd]# ansible-playbook  httpd.yml
[DEPRECATION WARNING]: The firewalld module has been moved to the ansible.posix collection. This feature will be removed from community.general in version 2.0.0. Deprecation warnings can
be disabled by setting deprecation_warnings=False in ansible.cfg.
Enter document root name : /var/www/idempotence
Enter port no : 8080

PLAY [httpd] ****************************************************************************************************************************************************

TASK [Gathering Facts] ******************************************************************************************************************************************
ok: [192.168.99.129]

TASK [Installing httpd package...] ******************************************************************************************************************************
ok: [192.168.99.129]

TASK [Creating dir for webserver...] ****************************************************************************************************************************
changed: [192.168.99.129]

TASK [Copying the configuration...] *****************************************************************************************************************************
changed: [192.168.99.129]

TASK [Copying html code to doc root...] *************************************************************************************************************************
changed: [192.168.99.129]

TASK [Starting httpd services...] *******************************************************************************************************************************
ok: [192.168.99.129]

TASK [Allow 8080] ***********************************************************************************************************************************************
ok: [192.168.99.129]

RUNNING HANDLER [Start Service] *********************************************************************************************************************************
changed: [192.168.99.129]

PLAY RECAP ******************************************************************************************************************************************************
192.168.99.129             : ok=8    changed=4    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible_controller Httpd]#
```

When you will run your playbook for the first time then you will find your all tasks are run and being shown the orange color, That means the task is run for the first time or Any modification is done in the task.

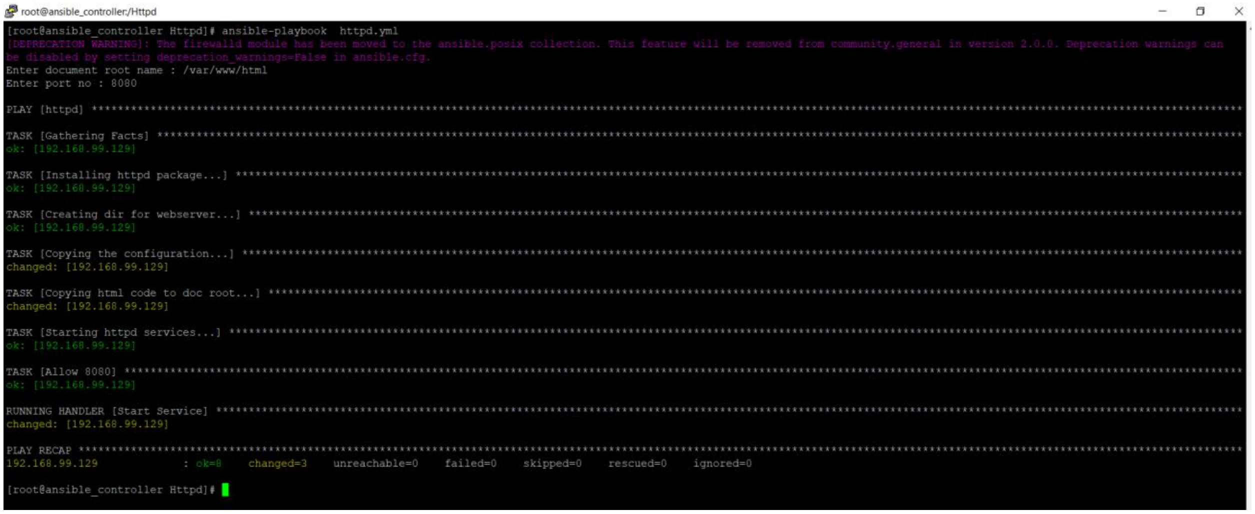
Playing the Playbook for the Second time without any change



```
[root@ansible_controller Httpd]# ansible-playbook  httpd.yml
[DEPRECATION WARNING]: The firewalld module has been moved to the ansible.posix collection. This feature will be removed from community.general in version 2.0.0. Deprecation warnings can
be disabled by setting deprecation_warnings=False in ansible.cfg.
Enter document root name : /var/www/idempotence
Enter port no : 8080

PLAY [httpd] ****************************************************************************************************************************************************

TASK [Gathering Facts] ******************************************************************************************************************************************
ok: [192.168.99.129]

TASK [Installing httpd package...] ******************************************************************************************************************************
ok: [192.168.99.129]

TASK [Creating dir for webserver...] ****************************************************************************************************************************
ok: [192.168.99.129]

TASK [Copying the configuration...] *****************************************************************************************************************************
ok: [192.168.99.129]

TASK [Copying html code to doc root...] *************************************************************************************************************************
ok: [192.168.99.129]

TASK [Starting httpd services...] *******************************************************************************************************************************
ok: [192.168.99.129]

TASK [Allow 8080] ***********************************************************************************************************************************************
ok: [192.168.99.129]

PLAY RECAP ******************************************************************************************************************************************************
192.168.99.129             : ok=7    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible_controller Httpd]#
```

Since there are no changes in the configuration file the handlers are not notified, and the Service is not restarted and also green color presents there is no change has happened.

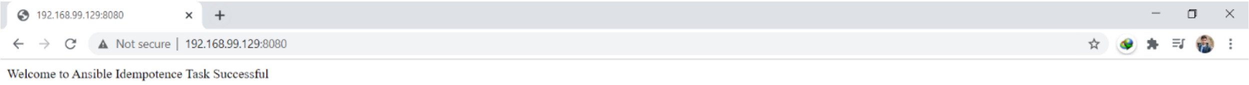
Playing the Playbook for the third time with a different Document root



```
[root@ansible_controller Httpd]# ansible-playbook  httpd.yml
[DEPRECATION WARNING]: The firewalld module has been moved to the ansible.posix collection. This feature will be removed from community.general in version 2.0.0. Deprecation warnings can
be disabled by setting deprecation_warnings=False in ansible.cfg.
Enter document root name : /var/www/html
Enter port no : 8080

PLAY [httpd] ************************************************************************************************************************

TASK [Gathering Facts] **************************************************************************************************************
ok: [192.168.99.129]

TASK [Installing httpd package...] **************************************************************************************************
ok: [192.168.99.129]

TASK [Creating dir for webserver...] ************************************************************************************************
ok: [192.168.99.129]

TASK [Copying the configuration...] *************************************************************************************************
changed: [192.168.99.129]

TASK [Copying html code to doc root...] *********************************************************************************************
changed: [192.168.99.129]

TASK [Starting httpd services...] ***************************************************************************************************
ok: [192.168.99.129]

TASK [Allow 8080] *******************************************************************************************************************
ok: [192.168.99.129]

RUNNING HANDLER [Start Service] *****************************************************************************************************
changed: [192.168.99.129]

PLAY RECAP **************************************************************************************************************************
192.168.99.129             : ok=8    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible_controller Httpd]# 
```

Since the playbook has been rerun with a change of Document root, there are some changes in the configuration file and the services has been restarted. (represented in orange color).

So let's see the webpage is working or not.



Welcome to Ansible Idempotence Task Successful

Conclusion

Here you have seen the concept of **idempotency** and the usages of the **handler.**

GitHub Link: https://github.com/Anuddeeph/Ansible_idempotence.git