Ansible – What is playbook? Playbooks are useful to perform multiple tasks and eliminates the limitation of Ad-hoc mode. Playbooks are the configuration, deployment and orchestration language of Ansible and it’s expressed in YAML format. If Ansible modules are the tools in your workshop, playbooks are your instruction manuals. YAML (Yet Another Markup Language) is human-readable data serialization language. This article will demonstrate that how to create a simple playbook and execute it.

**Writing the First Playbook:**

1. Login to Ansible engine server.

2. Create a new playbook to install “httpd” on Linux servers.

---

- hosts: all

become: yes

tasks:

- name: Install the latest version of Apache

yum: name=httpd state=latest update\_cache=yes

* **Playbook** should always start with three “**–**” and follow with mandatory hosts field which starts with “**–**“.
* For package management, we need to gain the root privileges. “**become**: yes”
* **tasks**: List of tasks that you would like to perform on the client nodes. You must provide the unique name for each task. In the Second line, we are using “yum” module to install the “httpd” version should be latest and requested to update the cache.

3. Save the file with an extension of “yaml”.

$ ls -lrt install\_httpd.yaml

-rw-rw-r-- 1 linadm linadm 294 Jul 25 21:47 install\_httpd.yaml

$

4. Create the list of servers where you would like to install the latest version of Apache.

$ ls -lrt lin-servers

-rw-r--r-- 1 linadm linadm 1370 Jul 12 02:26 lin-servers

$

5. Execute the playbook to deploy Apache on all the list of servers.

$ ansible-playbook -i lin-servers install\_httpd.yaml

PLAY [gpfslinapp1] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TASK [Gathering Facts] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [gpfslinapp1]

TASK [Install the latest version of Apache] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [gpfslinapp1]

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

gpfslinapp1 : ok=2 changed=1 unreachable=0 failed=0

6. Let’s verify our work.

$ ansible -a "rpm -qa httpd" -i lin-servers all

gpfslinapp1 | SUCCESS | rc=0 >>

httpd-2.4.6-80.el7.centos.1.x86\_64

7. Remove the “**httpd**” package before proceeding with next exercise.

We could be done the same thing in the ad-hoc mode as well.

**What’s the advantages of using playbook? Let’s explore.**

We should be able to use multiple tasks, handlers, loops and conditional execution using the playbook.

**How to use handlers in Ansible Playbook:?**

1. Update the playbook like below.

---

- hosts: all

become: yes

tasks:

- name: Install the latest version of Apache

yum: name=httpd state=latest update\_cache=yes

ignore\_errors: yes

notify: start Apache

handlers:

- name: start Apache

service: name=httpd enabled=yes state=started

* Included handlers in the playbook.
* Once the installation is completed, service must be started.
* Service must be enabled across the system reboot.

2. Execute the playbook.

$ ansible-playbook install\_httpd.yaml -i lin-servers.1

PLAY [gpfslinapp1] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TASK [Gathering Facts] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [gpfslinapp1]

TASK [Install the latest version of Apache] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [gpfslinapp1]

RUNNING HANDLER [start Apache] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [gpfslinapp1]

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

gpfslinapp1 : ok=3 changed=2 unreachable=0 failed=0

Here you could see that package “**httpd**” is installed and service is started automatically. It also enabled across the server reboot.

3. Verify our work by checking the apache’s service status on the client node.

$ ansible -a "systemctl status httpd" -i lin-servers.1 all

gpfslinapp1 | SUCCESS | rc=0 >>

● httpd.service - The Apache HTTP Server

Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)

Active: active (running) since Thu 2018-07-26 01:21:40 UTC; 4min 39s ago

Docs: man:httpd(8)

man:apachectl(8)

Main PID: 81943 (httpd)

Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"

CGroup: /system.slice/httpd.service

├─81943 /usr/sbin/httpd -DFOREGROUND

We have successfully created the first Ansible playbook. Using the playbook , we have installed apache on the servers and started the service automatically. In the upcoming sections, we will be writing many playbooks using different modules and functionality.  Stay tuned.