There’s another small quirk to YAML. All YAML files (regardless of their association with Ansible or not) can optionally begin with --- and end with .... This is part of the YAML format and indicates the start and end of a document.

All members of a list are lines beginning at the same indentation level starting with a "- " (a dash and a space):

**---**  
*# A list of tasty fruits*  
fruits:  
 - Apple  
 - Orange  
 - Strawberry  
 - Mango  
**...**

A dictionary is represented in a simple key: value form (the colon must be followed by a space):

*# An employee record*  
martin:  
 name: Martin D'vloper  
 job: Developer  
 skill: Elite

More complicated data structures are possible, such as lists of dictionaries, dictionaries whose values are lists or a mix of both:

*# Employee records*  
- martin:  
 name: Martin D'vloper  
 job: Developer  
 skills:  
 - python  
 - perl  
 - pascal  
- tabitha:  
 name: Tabitha Bitumen  
 job: Developer  
 skills:  
 - lisp  
 - fortran  
 - erlang

Dictionaries and lists can also be represented in an abbreviated form if you really want to:

**---**  
martin: {name: Martin D'vloper, job: Developer, skill: Elite}  
fruits: ['Apple', 'Orange', 'Strawberry', 'Mango']

These are called “Flow collections”.

Ansible doesn’t really use these too much, but you can also specify a boolean value (true/false) in several forms:

create\_key: yes  
needs\_agent: no  
knows\_oop: True  
likes\_emacs: TRUE  
uses\_cvs: false

Values can span multiple lines using | or >. Spanning multiple lines using a “Literal Block Scalar” | will include the newlines and any trailing spaces. Using a “Folded Block Scalar” > will fold newlines to spaces; it’s used to make what would otherwise be a very long line easier to read and edit. In either case the indentation will be ignored. Examples are:

include\_newlines: |  
 exactly as you see  
 will appear these three  
 lines of poetry  
  
fold\_newlines: >  
 this is really a  
 single line of text  
 despite appearances

While in the above > example all newlines are folded into spaces, there are two ways to enforce a newline to be kept:

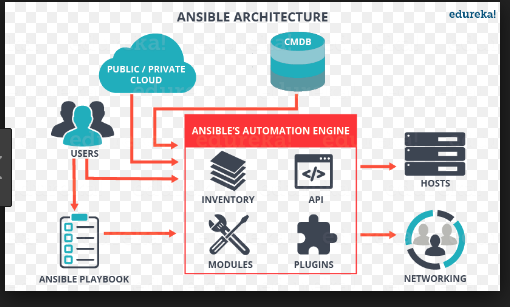
fold\_some\_newlines: >  
 a  
 b  
  
 c  
 d  
 e  
 f  
same\_as: "a b\nc d\n e\nf\n"

Let’s combine what we learned so far in an arbitrary YAML example. This really has nothing to do with Ansible, but will give you a feel for the format:

**---**  
*# An employee record*  
name: Martin D'vloper  
job: Developer  
skill: Elite  
employed: True  
foods:  
 - Apple  
 - Orange  
 - Strawberry  
 - Mango  
languages:  
 pe;rl: Elite  
 python: Elite  
 pascal: Lame  
education: |  
 4 GCSEs  
 3 A-Levels  
 BSc in the Internet of Things

That’s all you really need to know about YAML to start writing Ansible playbooks.

Ansible Architecture



**Instalaltion Ways** : 1) RPM 2) Yum 3 ) source code

**With EPEL:**

[root@ip-172-31-93-45 ~]# cat epel.sh

**#!/bin/bash**

**cd /tmp**

**mkdir ansiblepackage**

**cd ansiblepackage**

**echo "enabling the epel and downloading the rpm pack of ansible"**

**Yum install wget -y**

**wget http://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm**

**rpm -ivh epel-release-latest-7.noarch.rpm**

**cd /root**

**mkdir ansible**

**cd ansible**

**rm -r \*.rpm**

**ll**

**echo "installing the ansible"**

**yum install ansible -y**

**echo "installed ansible"**

**#ansible --version**

**[root@ip-172-31-93-45 ~]#**

[root@ip-172-31-93-45 ~]# **ansible --version**

ansible 2.5.3

config file = /etc/ansible/ansible.cfg

configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']

ansible python module location = /usr/lib/python2.7/site-packages/ansible

executable location = /bin/ansible

python version = 2.7.5 (default, Feb 20 2018, 09:19:12) [GCC 4.8.5 20150623 (Red Hat 4.8.5-28)]

[root@ip-172-31-93-45 ~]# **ansible -m ping loca**l

**[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'**

**[WARNING]: Could not match supplied host pattern, ignoring: local**

[root@ip-172-31-93-45 ansible]# ls -ltr

total 24

drwxr-xr-x. 2 root root 6 May 18 02:05 roles

-rw-r--r--. 1 root root 1016 May 18 02:05 hosts

-rw-r--r--. 1 root root 19315 May 18 02:05 ansible.cfg

[root@ip-172-31-93-45 ansible]# pwd

**/etc/ansible**

[root@ip-172-31-93-45 ansible]#

**Testing ansible in the local**

[root@ip-172-31-93-45 ansible]# **touch hosts**

[root@ip-172-31-93-45 ansible]# **vi hosts ( localhsot)**

[root@ip-172-31-93-45 ansible]# **ansible -m ping localhost**

**localhost | UNREACHABLE! => {**

**"changed": false,**

**"msg": "Failed to connect to the host via ssh: ssh: connect to host localhost port 22: Connection refused\r\n",**

**"unreachable": true**

**}**

[root@ip-172-31-93-45 ansible]# **ssh localhost**

ssh: connect to host localhost port 22: Connection refused

[root@ip-172-31-93-45 ansible]# ssh-keygen

Generating public/private rsa key pair.

Enter file in which to save the key (/root/.ssh/id\_rsa):

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /root/.ssh/id\_rsa.

Your public key has been saved in /root/.ssh/id\_rsa.pub.

The key fingerprint is:

SHA256:kqVJre0IBrHydZSAdjexpP1qoNHwTUVRyABMB76MxW0 root@ip-172-31-93-45.ec2.internal

The key's randomart image is:

+---[RSA 2048]----+

| .+=+\*\*++. |

| o=oB++o |

|..+.\*.E.o |

| o X B O |

| + O B S |

| + o = |

| . + . |

| . |

| |

+----[SHA256]-----+

[root@ip-172-31-93-45 ansible]#

Set the root password

[root@ip-172-31-93-45 .ssh]# **passwd**

Changing password for user root.

New password:

BAD PASSWORD: The password is shorter than 8 characters

Retype new password:

passwd: all authentication tokens updated successfully.

[root@ip-172-31-93-45 .ssh]#

[root@ip-172-31-93-45 .ssh]# **ssh-copy-id localhost**

/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id\_rsa.pub"

/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed

/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys

**root@localhost's password:**

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'localhost'"

and check to make sure that only the key(s) you wanted were added.

[root@ip-172-31-93-45 .ssh]# **ssh localhost**

Last login: Sun May 27 06:02:45 2018

[root@ip-172-31-93-45 ~]# exit

logout

Connection to localhost closed.

[root@ip-172-31-93-45 .ssh]#

[root@ip-172-31-93-45 .ssh]# **ansible -m ping localhost**

localhost | SUCCESS => {

"changed": false,

"ping": "pong"

}

[root@ip-172-31-93-45 .ssh]#

**Path** : /etc/ansbile

**Main Configuration file** : /etc/ansible/ansible.cfg

**Inventory file** : /etc/ansible/hosts ( two types of inventories static and dynamic)- servers list

**Sample Modules:**

Ping

Copy

Yum

File

Git

Service

Setup

shell

User

Creting a sample playbook

[root@controller ~]# pwd  
/root  
[root@controller ~]# mkdir playbooks  
[root@controller ~]# cd playbooks  
[root@controller playbooks]# pwd  
/root/playbooks

[root@controller playbooks]# cat HelloWorld.yml  
---  
- name: This is a hello-world example  
 **hosts:** webservers

**tasks:** - **name:** Create a file called '/tmp/testfile.txt' with the content 'hello world'.  
 **copy:** content="hello world\n" dest=/tmp/testfile.txt

**Executing the playbook**

[root@controller playbooks]# **ansible-playbook HelloWorld.yml**  
PLAY [**This is a hello-world example**] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
**TASK [setup]** \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
ok: [**webservers**]  
  
**TASK** [Create a file called '/tmp/testfile.txt' with the content 'hello world'.]  
changed: [**webservers**]  
  
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
webservers: ok=2 **changed=1** unreachable=0 failed=0

*Apache Playbook*

**/opt/playbooks/apache.yml**

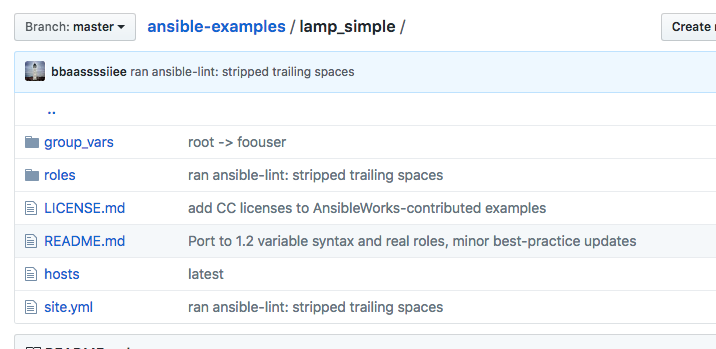
---  
- hosts: **apache** sudo: yes  
  **tasks:** - name: **install apache2**  
  **yum**: name=httpd update\_cache=yes state=latest

**Note**: state values are present(install),absent(remove/uninstall),latest(latest version)

Execute playbook:

**ansible-playbook apache.yml**

**LAMP playbook**



Directory Structure

roles-->common,db,web

common---->

→handlelrs(actions), called by notifyers

→tasks(main.yml)

→templates(jinza2 templates)

web--->

-->handlers

-->tasks

-->templates

-->files

db→

-->handllers

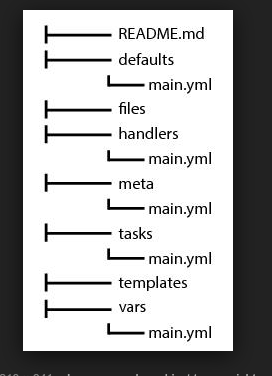
-->tasks

-->file

Executing above lamp playbook

**ansible-playbook -i hosts site.yml**

**Basic playbook structure**

****

**Basic Adhoc commands**

ansible -m shell -a 'echo $HOME'

ansible -m shell -a 'echo $HOME' localhost

ansible -m shell -a 'echo $USER' localhost

ansible -m shell -a 'echo $PWD' localhost

ansible -m copy -a "src=/etc/hosts dest=/tmp/hosts" localhost

ls -ltr /tmp

Check hosts file exist or not.

**Installaing apache**

ansible -m yum -a "name=httpd state=present" localhost

Start Apache

ansible localhost -m service -a "name=httpd state=started"

Stop Apache

ansible localhost -m service -a "name=httpd state=stopped"

Enable Apache

ansible -m service -a "name=httpd state=running enabled=yes" localhost

Uninstall apache

ansible -m yum -a "name=httpd state=absent" localhost

**Nginx**

Install

ansible -m yum -a "name=nginx state=present" localhost

Start Nginx

ansible localhost -m service -a "name=nginx state=started"

Stop Nginx

ansible localhost -m service -a "name=nginx state=stopped"

Enable Nginx

ansible -m service -a "name=nginx state=running enabled=yes" localhost

Uninstall Nginx

ansible -m yum -a "name=nginx state=absent" localhost

Ansible Playbooks

https://github.com/ansible/ansible-examples.git

git clone https://github.com/ansible/ansible-examples.git

[root@ip-172-31-93-45 ansible-examples]# pwd

/root/ansible-examples

[root@ip-172-31-93-45 ansible-examples]# ls -ltr

total 8

-rw-r--r--. 1 root root 122 May 28 02:52 README.md

drwxr-xr-x. 4 root root 158 May 28 02:52 jboss-standalone

drwxr-xr-x. 5 root root 159 May 28 02:52 lamp\_haproxy

drwxr-xr-x. 4 root root 101 May 28 02:52 lamp\_simple

drwxr-xr-x. 4 root root 101 May 28 02:52 **lamp\_simple\_rhel7**

drwxr-xr-x. 9 root root 4096 May 28 02:52 language\_features

drwxr-xr-x. 6 root root 132 May 28 02:52 mongodb

drwxr-xr-x. 4 root root 101 May 28 02:52 tomcat-memcached-failover

drwxr-xr-x. 4 root root 101 May 28 02:52 tomcat-standalone

drwxr-xr-x. 4 root root 188 May 28 02:52 windows

drwxr-xr-x. 4 root root 109 May 28 02:52 wordpress-nginx

drwxr-xr-x. 4 root root 109 May 28 02:52 wordpress-nginx\_rhel7

[root@ip-172-31-93-45 ansible-examples]#

To work above playbook follow below modifications.

1)update site.yml

2)update hosts

3)roles/web/templates/index.php.j2:

Update dbservers with qal

4)Rename under group\_vars/dbservers to group\_vars/qal

ansible-playbook -i hosts site.yml

Take the qal ip and test the url

(all sg with 80)

You will hav the page with o/p

Hello World! My App deployed via Ansible V6.