Ansible is a Configuration management tool used for deployment, automation, orchestration and it’s a push based tool.

Features of Ansible

* simple
* agentless
* powerful
* efficient

Ansible tower, which is a GUI for managing the organization, like monitoring, user permissions, assigning the jobs based on user level, we can manage the inventory as well. it pretty much similar to a dashboard.

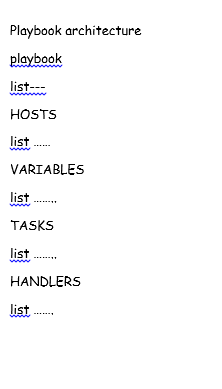
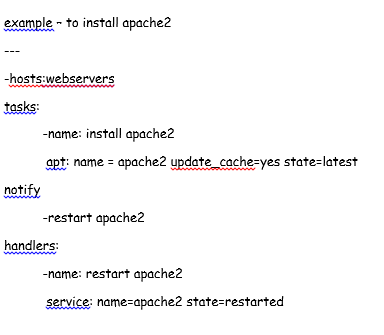
**Playbooks** contains plays

**Plays** contain Tasks

**Tasks** call modules

**tasks** run **sequentially**

**Handlers** are triggered by **tasks**, and are run once, at the end of **plays**.

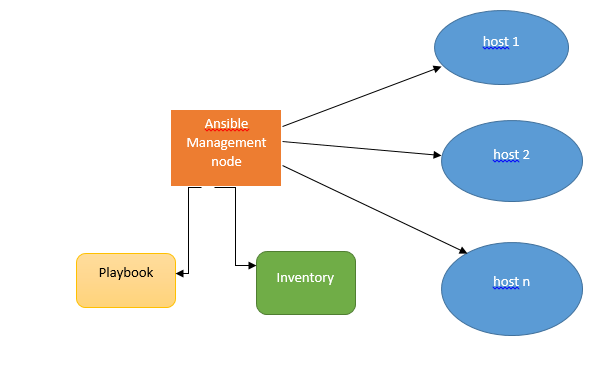
 

Ansible – Terminology

* Playbook – describes automation jobs (i.e. Ansible commands)

YAML – Playbooks are written using YAML

* Ansible is meant for multi-tier deployment and this is done connecting the nodes through ssh.
* Once this connecting of nodes is done, ansible executes the ansible modules. These modules consists small programs and once the execution completes on the nodes then it removes these.
* Management node – acts as a controlling node for the execution of the playbook
* Inventory file – will give the list of host on which the execution/installation need to be done



* when we discuss about Installation of Ansible, we need to keep in mind that there will be 2 machines on which we will be working,
  + - * + control machine
        + remote machine
  + Ansible code is written using Python, Ansible can’t be installed on windows and its always needs to be executed using ssh on the control machine.

it can be installed using pip, apt-get, yum, dpkg, etc.

* + - * + Follow the below steps to install the ansible on Ubuntu

sudo apt-get update

sudo apt-get install software-properties-common

sudo apt-add-repository ppa:ansible/ansible $ sudo apt-get update

sudo apt-get install ansible

sudo ansible – version

Now with the execution of the above commands you are ready to manage the remote machines using the control machine.

* YAML – sample code,

---

-devopsforum:

user:admin

posts:10

div: A

sex: male

Expert:

- AWS

- DevOps

result:

AWS: 87

DevOps: 45

Ansible: 56

passed: TRUE

messageIncludeNewLines: |

Congratulation!!

You are getting trained

messageExcludeNewLines: >

Congratulation!!

You are getting trained

Ansible has got Ad Hoc commands, which are meant to be one time use.

Ansible playbook is used for management configuration and deployment.

Sample AdHoc commands,

* To transfer the files we need to frame the command as shown below,

$ Ansible abc -m copy -a "src = /etc/yum.conf dest = /tmp/yum.conf"

The above command says to transfer file to many servers/machines.

* Creating a directory,

$ Ansible abc -m file -a "dest = /path/user1/new mode = 777 owner = user1 group = user1 state = directory"

* Deleting the whole directory

$ Ansible abc -m file -a "dest = /path/user1/new state = absent"

$ Ansible all -m setup

Playbooks, these are the files where Ansible code is written In the YAML format. Playbook contains the steps which the user wants to execute on the machine.

The tags in playbook,

**name**

This tag specifies the name of the Ansible playbook. As in what this playbook will be doing. Any logical name can be given to the playbook.

**hosts**

This tag specifies the lists of hosts or host group against which we want to run the task. The hosts field/tag is mandatory. It tells Ansible on which hosts to run the listed tasks. The tasks can be run on the same machine or on a remote machine. One can run the tasks on multiple machines and hence hosts tag can have a group of hosts’ entry as well.

**vars**

Vars tag lets you define the variables which you can use in your playbook. Usage is similar to variables in any programming language.

**tasks**

All playbooks should contain tasks or a list of tasks to be executed. Tasks are a list of actions one needs to perform. A tasks field contains the name of the task. This works as the help text for the user. It is not mandatory but proves useful in debugging the playbook. Each task internally links to a piece of code called a module. A module that should be executed, and arguments that are required for the module you want to execute.

git credentialsId: '2b1dd599-bd98-4dcd-9b33-fdf9c23a4357', url: 'https://github.com/YvsVenkat/HelloWorld.git'