Ansible

Agenda

What is Ansible?
What can Ansible do?
Ansible Installation
Configuration Management
Deployment

Introduction

Ansible is an open source, an IT automation tool. It can configure systems, deploy software, and orchestrate more advanced IT tasks such as continuous deployments or zero downtime rolling updates, like Chef, Puppet or Salt.

The main components of Ansible are playbooks, configuration management, deployment.

Ansible uses playbooks to deploy, manage, build, test and configure anything from full server environments to custom compiled source code for applications.

Ansible was written in Python.

Ansible Features

Ansible manages machines in an agent-less manner using SSH.
Built on top of Python and hence provides a lot of Python's functionality.
YAML-Based Playbooks
Uses SSH for secure connections.
Follows Push based architecture for sending configurations.

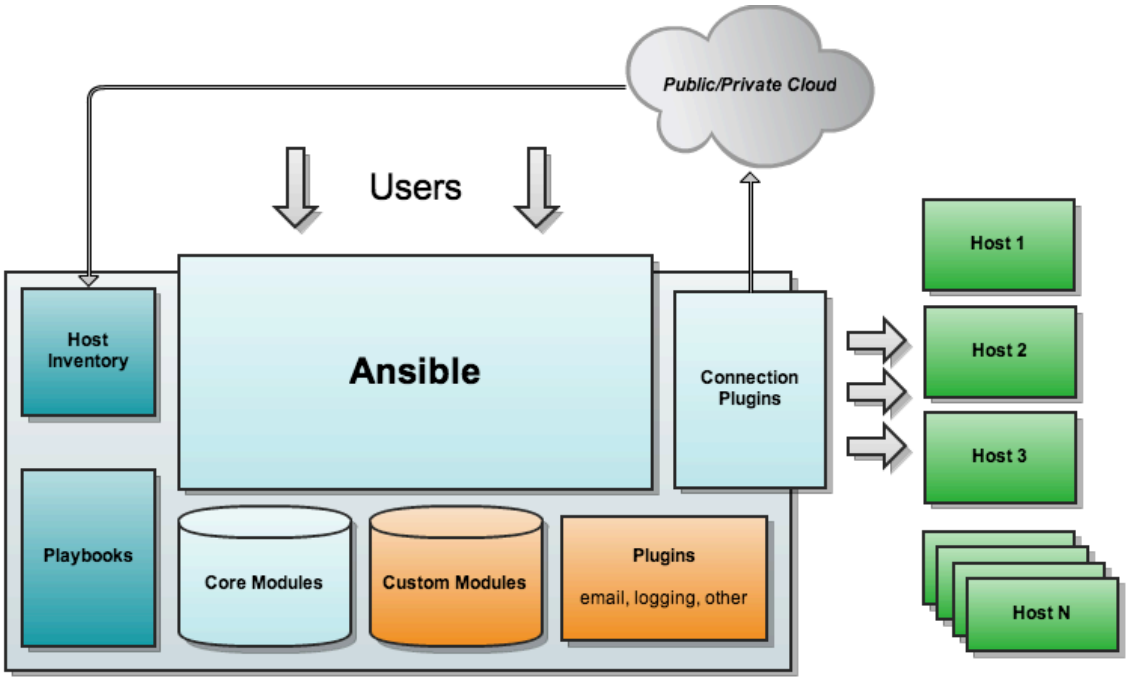
Push Based Vs Pull Based

Tools like Puppet and Chef are pull based.
Agents on the server periodically checks for the configuration information from central server (Master).

Ansible is push based.
Central server pushes the configuration information on target servers.
You control when the changes are made on the servers.

What can Ansible do?
Configuration Management
App Deployment
Continuous Delivery
Security & Compliance

Ansible Architecture



Inventory file

After you've installed Ansible, then you'll want Ansible to know which servers to connect to and manage.
Ansible's inventory hosts file is used to list and group your servers. Its default location is /etc/ansible/hosts.

See the contents in hosts file as follows.

cat /etc/Ansible/hosts (default inventory file path)
192.168.122.1 ---> This is one of the nodes IP
192.168.122.2
mithuntechnolopgies.dev.com

In Inventory file you can mention IP address or Hostnames also.

Some important points in Inventory file.

- Comments begin with the '#' character
- Blank lines are ignored
- Groups of hosts are delimited by [header] elements
- You can enter hostnames or ip addresses
- A hostname/ip can be a member of multiple groups

Sample Inventory file1

We can use '#' for comments in inventory file.

#Blank line are ignored.

#Ungrouped hosts are specify before any group headers, like below
192.168.122.1
192.168.122.2
mithun-technolopgies.dev.com

[webservers]
192.168.122.1
192.168.122.2
192.168.122.3

[dbservers]
#mithun-techno.db1.com
#mithun-techno.db2.com
#mithun-techno.db3.com
mithun-techno.db[1:3].com
192.168.122.4
192.168.122.5
192.168.122.6

```
appservers        ansible_host=mithun-techno.appserver1.com   ansible_connection=ssh
mailservers       ansible_host=mithun-techno.mailserver.com   ansible_connection=winrm
databseservers    ansible_host=mithun-techno.db.com           ansible_connection=ssh
```

Inventory Parameters

ansible_connection=ssh/winrm/localhost
ansible_port=22/5986
ansible_user=root/administrator
ansible_ssh_lass=<<Password for node>>

for localhost

```
localhost ansible_connection=localhost
```

If you want to have your Ansible hosts file in another location, then you can set this environment variable:

export ANSIBLE_HOSTS=/root/ansible_hosts

Or you can specify the Ansible hosts location when running commands with the --inventory-file= (or -i) flag:

ansible all --inventory-file=/root/ansible_hosts -m ping

Reference URL: http://docs.ansible.com/ansible/latest/intro_inventory.html

Setup Connectivity between Ansible server and Nodes

Step 1) Generate the SSH Key using **'ssh-keygen'** command.

#ssh-keygen -t rsa -C "mithun@mithuntechnologies.com"

Step 2) copy your public key to the nodes using **'ssh-copy-id'** command.

ssh-copy-id mithun@mithuntechnologies.dev.com

Ansible AD-HOC Commands

ansible-doc -l: It will display the all the modules available in Ansible.

ansible all -m ping: It will ping all the servers which you have mentioned in inventory file (/etc/Ansible/hosts).
ansible all -m ping -o: It will display the output in single line.

```
[root@localhost ~]# ansible all -m ping
192.168.122.1 | SUCCESS => {
    "changed": false,
    "failed": false,
    "ping": "pong"
}
[root@localhost ~]# ansible all -m ping -o
192.168.122.1 | SUCCESS => {"changed": false, "failed": false, "ping": "pong"}
[root@localhost ~]#
```

ansible all -m shell -a 'uptime' : Uptime of all the machines
(OR)
ansible all -a 'uptime'

```
[root@localhost ~]# ansible all -m shell -a 'uptime'
192.168.122.1 | SUCCESS | rc=0 >>
 17:15:46 up 22:23,  6 users,  load average: 0.35, 0.22, 0.15

[root@localhost ~]#
```

```
[root@localhost ~]# ansible all -a 'uptime'
192.168.122.1 | SUCCESS | rc=0 >>
 17:17:56 up 22:25,  6 users,  load average: 0.36, 0.21, 0.15

[root@localhost ~]#
```

ansible all -m shell -a 'date' : Date of all machines

```
[root@localhost ~]# ansible all -m shell -a 'date'
192.168.122.1 | SUCCESS | rc=0 >>
Sat Nov 11 17:13:51 IST 2017

[root@localhost ~]#
```

ansible all -m shell -a 'cat /etc/redhat-release' : Redhat release of all the machines.
ansible all -m shell -a 'mount' : Kind of mount on all the machines
ansible all -m shell -a 'service sshd status' : Check the service status on all the machines.

ansible dbservers -a "df -h" : Here it will check the disk space use for all the nodes which are from dbservers group

ansible multi -a "free -m":

ansible multi -a "date":

ansible all -m shell -a 'uname -a' -v

```
[root@localhost ~]# ansible all -m shell -a 'uname -a' -v
Using /etc/ansible/ansible.cfg as config file
192.168.122.1 | SUCCESS | rc=0 >>
Linux localhost.localdomain 3.10.0-693.el7.x86_64 #1 SMP Tue Aug 22 21:09:27 UTC
 2017 x86_64 x86_64 x86_64 GNU/Linux

[root@localhost ~]#
```

ansible all -m yum -a 'name=httpd state=present' : To install httpd package in node machines.
ansible all -m service -a 'name=httpd state=started' -s :

rpm -qa | grep httpd : It will check weather httpd package is installed or not.

If ant access issue need to give the sudo access to ansadm in all client machines(nodes) as follows.

vim /etc/sudoers
ANSIBLE ADMIN user
ansadm ALL=NOPASSWD: ALL

Commands: https://www.youtube.com/watch?v=lRwGkO3PtB8

Playbooks

XML	JSON	YAML

<Servers>
<Server>
<name>Server1</name>
<owner>Mithun</owner>
<created>08112012</created>
<status>active</status>
</Server>
</Server>
	{
	Servers: [
	{
	name: Server1,
	owner: Mithun,
	created: 08112012,
	status: active,
	}
	]
	}
		Servers:
		- name: Server1
		owner: Mithun
		created: 08112012
		status: active

Key Value Pair

Fruit: Apple
Vegetable: Carrot
Liquid: Water
Meet: Chicken
Note: Need to give the space between ':' and value.

Array/List

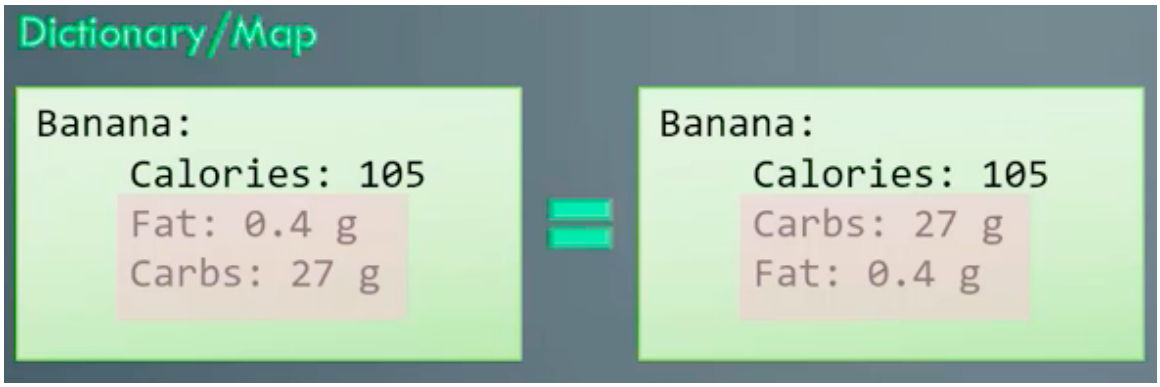
Fruits:
- Orange
- Apple
- Banana

Vegetables:
- Carrot
- Cauliflower
- Tomoto

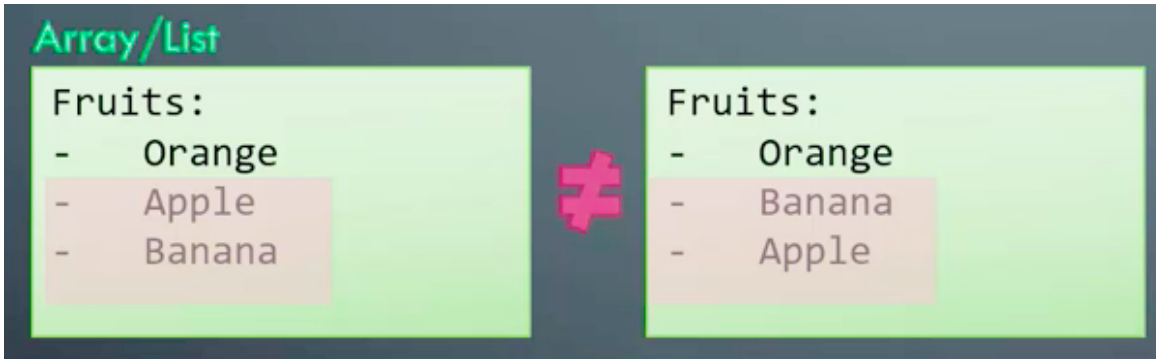
Dictornary/Map

Banana:
Calories: 105
Fat: 0.4 g
Carbs: 27 g

Grapes:
Calories: 105
Fat: 0.4 g
Carbs: 27 g



Dictonary is Unordered.



List is Ordered.
List of Fruits (Hash # Is used for comments)
Fruits:
- Orange
- Apple
- Banana

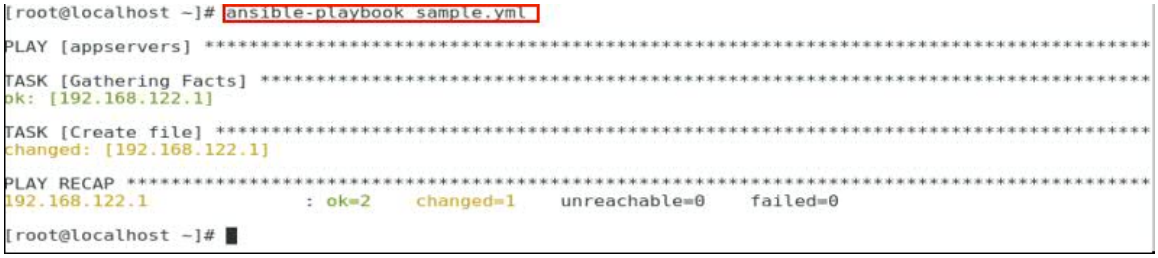
Playbook is a single YAML file, composed of one or more ‘plays’ in a list.

Play defines a set of activities (tasks) to be run on hosts.
Task is an action to be perform on the host.
Examples are a) Execute a command
b) Run a shell script
c) Install a package
d) Shutdown/Restart the hosts.

Playbooks start with the YAML three dashes (---) followed by:

FileName: sample.yml

```
- hosts: appservers
  tasks:
     - name: Create file
       file:
          path: /tmp/mithun
          state: touch
```



Items that begin with a - are considered list items.

FileName: playbook1.yml

```
---
-
  name: Playbook1
  hosts: appservers
  tasks :
    - name: Execute the command the 'date'
      command: date

    - name: Execute script on server
      script: sample_script.sh

    - name: Install httpd service
      yum:
          name: httpd
          state: present

    - name: Start web server
      service:
          name: httpd
          state: started

-
  name: Playbook2
  hosts: dbservers
  tasks :
    - name: Execute the command the 'date'
      command: date
```

Run the playbook as follows.
ansible-playbook <<Playbbok file name>>
ansible-playbook playbook1.yml

ansible-playbook --help: It will provides help on ansible_playbook command.

Modules
System
Commands
Files
Database
Cloud
Windows

Video 6

Variables

Conditionals

Loops

Roles

Resources
https://valdhaus.co/writings/ansible-mac-osx/
http://binarynature.blogspot.in/2016/01/install-ansible-on-os-x-el-capitan_30.html