**Ansible**

Ansible is open source, a configuration Management Tool and Deployment Tool, maintained by ***Redhat***

Configuration is some tasks, which we want to execute on the server, it can be

* Creating users/Groups
* Installing software or running software
* Creating / updating / copying files
* Start / stop services etc

It is not replacement tool, the main components of Ansible are playbooks, configuration management, deployment

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

Ansible was written in Python

Open-source configuration tools

* Ansible
* Chef
* Puppet
* Saltstack

**Ansible how different is from Terraform?**

**Terraform:** Infrastructure as a code

By using terraform we can create the server once we have a server, we can use Ansible to configure those servers based on our requirement

Ansible does not maintain state of server, by using Ansible also we can create the server but it is not better

Option.

**Ansible Features**

Ansible configure machines in an agent-less manner using SSH

Built it on top of Python and hence provide a lot of Python’s functionality

YAML – Based Playbooks

Uses SSH for secure connections

Follows Push based architecture for sending configurations

**Two types of configuration tools**

1.Push based

2.Pull based

**Push based**

Master Server (Installed Ansible) have some configuration scripts, then Main server only will connect to different host servers for copying files or install software’s etc

In other words, Center server pushes the configuration information on target servers

Ansible is push based

**Pull based**

Here, target server only pull to center server

Tools like Puppet and chef and pull based.

Agents on the server periodically checks for the configuration information from central server (Master)

**Note:**

**We cannot install ansible in window machine**

**We can configure windows machine using Ansible**

Ansible is not master slave architecture, host machine can be anything for example installed Jenkins, installed Docker, installed Kubernetes.

**Why Ansible**

* Predefined module/functionality
* No need to write everything from scratch
* Ansible scripts basically YAML based
* Not required to learn scripting language, because everything will be YAML
* It will save the time
* Ansible Modules are idempotent

Idempotent: If configuration already done for that server, it will not change anything

* Lot of predefined functionality available in Ansible
* Agent less and push based
* Manage the machine using SSH protocol

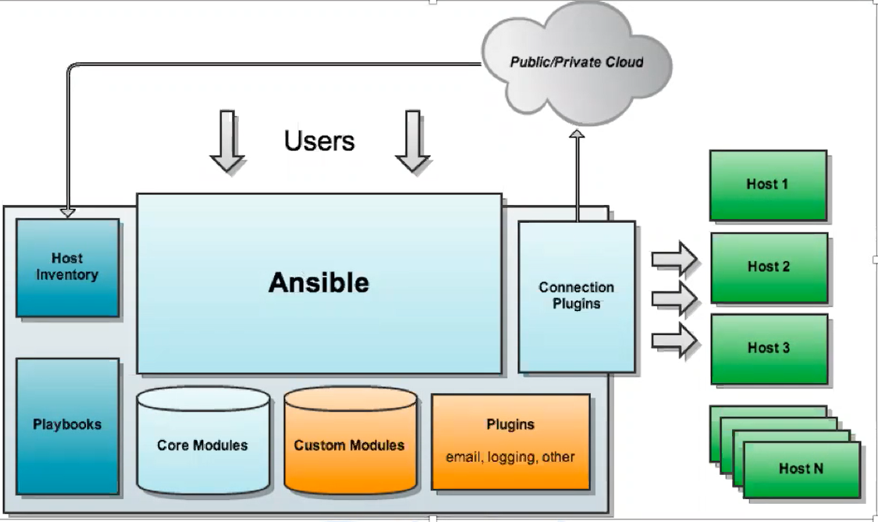
**What can Ansible do?**

Configuration management tool

Application Deployment

Continuous Delivery

**Ansible Architecture**



**Ansible Modules**

1.Host Inventory

2.Playbooks

3.Core Modules

4.Custom Modules

5.Plugins (email,logging,other)

6.Connection Plugins

**Inventory file**

Ansible’s inventory hosts file is used to list and group your servers. Its default location is

etc/ansible/hosts

See the content of the host file

#cat /etc/ansible/hosts (default inventory file path)

#192.168.122.1 This is one of the node IP

In Inventory file we can mentioned IP address or hostname also

We can create our own inventory for that we need to mentioned in below path

The ansible\* command will use a different host inventory file when they are used with the

--inventory PATHNAME option, -I PATHNAME for shot

Some important points in inventory file

* Comments begin with the ‘#’ character
* Blank lines are ignored
* Groups of hosts are delimited by [header] element
* You can enter hostnames or ip multiple groups
* A hostname/ip can be a member of multiple groups
* Ungrouped hosts are specifying before any group headers, like below

**Host Inventory**

It will maintain the details for HOST machine (server) to configure, it can be list and group of servers(Host)

1.Static Inventory

2.Dynamic Inventory

**Static Inventory**

It is file in which we can list host(server) details like (host name, password details etc and we can group the servers

Default location of host inventory

* /etc/ansible/hosts

**Dynamic Inventory**

If any changes in infrastructure like remove server add some servers or terminate server

We need to update the inventory file based on the changes

It is script (like python, shell scripts) which will fetch host details dynamically from external source like cloud Provider etc

No need to update the details manually

**When we can go for Static and Dynamic Inventory?**

If our requirement is not dynamic (like not creating server / deleting servers / terminating / modifying server very frequently then we can go for static inventory or else we can use dynamic inventory

Connection Plugin

To connect windows or Linux it will use some connection plugin

For Linux 🡪 SSH

For window 🡪 WINRM(windows remote machine)

**Playbooks**

Playbook is a script which we want to execute(configuration) YAML based scripts

It contains task which we want to perform on the servers

**Sample script**

host: all 🡪 all the machine going to perform below task

name: create a user

user:

name: rajesh

createHome: yes

password: abc123

**Module**

Modules are the programs that perform the actual work of the tasks of a play

**Core Module**

Ansible comes with lot of predefined modules we will use ansible module to write task

There are 400 core modules

**Tasks**

The goal of play is to map a group of hosts to some well-defined roles, represented by things ansible calls tasks. At a basic level, a task is nothing more than a call to an ansible module

**Custom Module**

If expected module which is not available in the core module, in this case we will create our own module and will publish same will called as Custom Module

**Playbook:**

Ansible playbooks are written using the YAML it’s Markup Language (YAML) language.

We can create a file with **.yml** or **yaml** extension

YAML files optionally begin with a three dash (---)

Next immediate line starts with single dash (-). Name is optional

hosts expect value like all or group

Do you want to become a root on target server uses become

What action do you want to perform? specify under tasks