ANSIBLE

1. What is Ansible?

Ansible is one of the configuration Management Tools. It is a method through we automate system admin tasks. Configuration refers to each and every minute details of a system. If we do any changes in system means we are changing the configuration of a machine. That means we are changing the configuration of the machine. All windows/Linux system administrators manage the configuration of a machine manually. All DevOps engineers are managing this configuration automatic way by using some tools which are available in the market. One such tool is Ansible. That's why we call Ansible as configuration management tool.

2. Working process of Ansible?

Here we crate file called playbook and inside playbook we write script in YAML format to create infrastructure. Once we execute this playbook, automatically code will be converted into Infrastructure. We call this process as IAC (Infrastructure as Code). We have open source and enterprise editions of Ansible. Enterprise edition we call Ansible Tower.

3. Architecture of Ansible?

We create Ansible server by installing Ansible package in it. Python is pre-requisite to install ansible. We need not to install ansible package in nodes. Because, communication establishes from server to node through “ssh” client. By default all Linux machine will have “ssh” client. Server is going to push the code to nodes that we write in playbooks. So Ansible follows pushing mechanism.

4. Ansible components?

Server: - It is the place where we create playbooks and write code in YML format
Node: - It is the place where we apply code to create infrastructure. Server pushes code to nodes.
Ssh: - It is an agent through ansible server pushes code to nodes.
Setup: - It is a module in ansible which gathers nodes information.
Inventory file:- In this file we keep IP/DNS of nodes.

5. Disadvantages in other SCM (Source Code Management) tools?

Huge overhead of Infrastructure setup
Complicated setup
Pull mechanism
Lot of learning required

6. Advantages of Ansible over other SCM (Source Code Management) tools?

Agentless
Relies on “ssh”
Uses python
Push mechanism

7. How Ansible works?

We give nodes IP addresses in hosts file by creating any group in ansible server why because, ansible doesn't recognize individual IP addresses of nodes. We create playbook and write code in YAML script. The group name we have to mention in playbook and then we execute the playbook. By default, playbook will be executed in all those nodes which are under this group. This is how ansible converts code into infrastructure.

8. What do you mean by Ad-Hoc commands in Ansible?

These are simple one liner Linux commands we use to meet temporary requirements without actually saving for later. Here we don't use ansible modules. So there, Idempotency will not work with Ad-Hoc commands. If at all we don't get required YAML module to write to create infrastructure, then we go for it.

Without using playbooks we can use these Ad-Hoc commands for temporary purpose.

9. Differences between Chef and Ansible?

Ansible	chef
Playbook	Recipe
Module	Resource
Host	Node
Setup	Ohai
Ssh	Knife
Push	Pull

10. What is Playbook in Ansible?

Playbook is a file where we write YAML script to create infrastructure in nodes. Here, we use modules to create infrastructure. We create so many sections in playbook. We mention all modules in task section. You can create any no of playbooks. There is no limit. Each playbook defines one scenario. All sections begin with "-" & its attributes & parameters beneath it.

11. Mention some list of sections that we mention in Playbook?

1. Target section

2. Task section
3. Variable section
4. Handler section

12. What is Target section in Ansible playbook?

This is one of the important sections in Playbook. In this section, we mention the group name which contains either IP addresses or Hostnames of nodes. When we execute playbook, then code will be pushed too all nodes which are there in the group that we mention in Target section. We use "all" key word to refer all groups.
-

13. What is Task section in Ansible playbook?

This is second most important section in playbook after target section. In this section, we are going to mention list of all modules. All tasks we mention in this task section. We can mention any no of modules in one playbook. There is no limit. If there is only one task, then instead of going with big playbook, simply we can go with arbitrary command where we can use one module at a time. If more than one module, then there is no option except going with big playbook.

14. What is Variable section?

In this section we are going to mention variables. Instead of hard coding, we can mention as variables so that during runtime it pulls the actual value in place of key. We have this concept in each and every programming language and scripting language. We use "vars" key word to use variables.

15. What is Handler section?

All tasks we mention in tasks section. But some tasks where dependency is there, we should not mention in tasks section. That is not good practice. For example, installing package is one task and starting service is one more task. But there is dependency between them. I.e. after installing package only, we have to start service. Otherwise it throws error. These kind of tasks, we mention in handler section. In above example, package task we mention in task section and service task we mention in handler section so that after installing task only service will be started.

16. What is Dry run in playbook?

Dry run is to test playbook. Before executing playbook in nodes, we can test whether the code in playbook is written properly or not. Dry run won't actually executes playbook, but it shows output as if it executed playbook. Then by seeing the output, we can come to know whether the playbook is written properly or not. It checks whether the playbook is formatted correctly or

not. It tests how the playbook is going to behave without running the tasks.

17. Why are we using loops concept in Ansible?

Sometimes we might need to deal with multiple tasks. For instance, Installing multiple packages, Creating many users, creation many groups..etc. In this case, mentioning module for every task is complex process. So, to address this issue, we have a concept of loops. We have to use variables in combination with loops.

18. Where do we use conditionals in Playbooks?

Sometimes, your nodes could be mixture of different flavors of Linux OS. Linux commands vary in different Linux operating systems. In this case, we can't execute common set of commands in all machines, at the same time, we can’t execute different commands in each node separately. To address this issue, we have conditionals concept where commands will be executed based up on certain condition that we give.

19. What is Ansible vault?

Sometimes, we use sensitive information in playbooks like passwords, keys ...etc. So any one can open these playbooks and get to know about this sensitive information. So we have to protect our playbooks from being read by others. So by using Ansible vault, we encrypt playbooks so that, those who ever is having password, only those can read this information. It is the way of protecting playbooks by encrypting them.

20. What do you mean by Roles in Ansible?

Adding more & more functionality to the playbooks will make it difficult to maintain in a single file. To address this issue, we organize playbooks into a directory structure called “roles”. We create separate file to each section and we just mention the names of those sections in playbook instead of mentioning all modules in main playbook. When you call main playbook, main playbook will call all sections files respectively in the order whatever order you mention in playbook. So, by using this Roles, we can maintain small playbook without any complexity.

21. Write a sample playbook to install any package?

```
--- # My First YAML playbook
- hosts: demo
  user: ansible
  become: yes
  connection: ssh
```

```
  tasks:
    - name: Install HTTPD on centos 7
      action: yum name=httpd state=installed
```

22. Write a sample playbook by mentioning variables instead of hard coding?

```
--- # My First YAML playbook
- hosts: demo
  user: ansible
  become: yes
  connection: ssh
  vars:
    pkgname: httpd
  tasks:
    - name: Install HTTPD server on centos 7
      action: yum name='{{pkgname}}' state=installed
```