Ansible Assignment 4

1. Describe the Ansible register.

Ansible register is a way to capture the output from task execution and store it in a variable. This is an important feature, as this output is different for each remote host, and the basis on that we can use conditions loops to do some other tasks. Also, each register value is valid throughout the playbook execution.

2. In Ansible, how can we delegate tasks?

If we want to perform a task on one host with reference to other hosts, use the **delegate_to keyword** on a task. This is ideal for managing nodes in a load balanced pool or for controlling outage windows. We can use delegation with the serial keyword to control the number of hosts executing at one time.

3. What is the best way to install Ansible on a CentOS system?

Step 1: Set up EPEL Repository

EPEL (Extra Packages for Enterprise Linux) is an open source and free community-based repository project from Fedora team which provides high-quality add-on software packages for Linux distribution including RHEL (Red Hat Enterprise Linux), CentOS, and Scientific Linux.

The Ansible package is not available in the default yum repositories, so we will enable EPEL repository by using the below command:

```
sudo rpm -ivh http://dl.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm
```

This will download all the necessary packages which will be required to install Ansible.

Step 2: Install Ansible

Now that your EPEL repository has been added, all you have to do now is install Ansible using the command below:

```
yum install ansible -y
```

4. What is Ansible and how does it differ from other Configuration Management software?

Ansible is an open source IT Configuration Management, Deployment & Orchestration tool. It aims to provide large productivity gains to a wide variety of automation challenges.

Here’s a list of features that makes Ansible such an effective Configuration Management and Automation tool:

- Simple: Uses a simple syntax written in YAML called playbooks.
- Agentless: No agents/software or additional firewall ports that you need to install on the client systems or hosts which you want to automate.
- Powerful and Flexible: Ansible’s capabilities allow you to orchestrate the entire application environment regardless of where it is deployed.
- Efficient: Ansible introduces modules as basic building blocks for your software. So, you can even customize it as per your needs.

5. What are the various parts of ansible? Describe the architecture of Ansible.

The main component of Ansible is the Ansible automation engine. This engine directly interacts with various cloud services, Configuration Management Database (CMBD) and different users who write various playbooks to execute the Ansible Automation engine.

The Ansible Automation engine consists of the following components:

- Inventories: These are a list of nodes containing their respective IP addresses, servers, databases, etc. which needs to be managed.

- APIs: Just like any other API, the Ansible APIs are used for commuting various Cloud services, public or private services.

- Modules: The modules are used to manage system resources, packages, libraries, files, etc. Ansible modules can be used to automate a wide range of tasks. Ansible provides around 450 modules that automate nearly every part of your environment.

- Plugins: If you want to execute Ansible tasks as a job, Ansible Plugins can be used. They simplify the execution of a task by building a job like an environment that basically contains pieces of code corresponding to some specific functionality. There are 100s of Plugins provided by Ansible. An example is the Action plugin, which acts as front ends to modules and can execute tasks on the controller before calling the modules themselves.

- Networking: Ansible can also be used to automate different networks and services. It can do this by creating a playbook or an Ansible role that easily spans different network hardware.

- Hosts: The Ansible Hosts/ Node systems are machines (Linux, Windows, etc) that are getting automated.

- Playbooks: Playbooks are simple code files which describe the tasks that need to be executed. The Playbooks are written in YAML format. They can be used to automate tasks, declare configurations, etc.

- CMDB: It is a database that acts as a storehouse for various IT installations. It holds data about various IT assets (also known as configuration items (CI)) and describes the relationships between such assets.

- Cloud: It is a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server.

6. What are the requirements for Ansible Server?

We need to have a virtual machine in which Linux should be installed. It requires Python 2.6 version or higher.

7. Describe a handful of Ansible's basic terms and concepts.

- Controller Machine: The Controller machine is responsible for provisioning the servers that are being managed. It is the machine where Ansible is installed.

- Inventory: An inventory is an initialization file that has details about the different servers you are managing.

- Playbook: It is a code file written in the YAML format. A playbook basically contains the tasks that need to be executed or automated.

- Task: Each task represents a single procedure that needs to be executed, e.g. Install a library.

- Module: A module is a set of tasks that can be executed. Ansible has 100s of built-in modules, but you can also create custom ones.

- Role: An Ansible role is a pre-defined way for organizing playbooks and other files in order to facilitate sharing and reusing portions of provisioning.

- Play: A task executed from start to finish or the execution of a playbook is called a play.

- Facts: Facts are global variables that store details about the system, like network interfaces or operating system.

- Handlers: Are used to trigger the status of a service, such as restarting or stopping a service.

8. Describe the Infrastructure as Code idea (IaC).

Infrastructure as Code (IaC) is a process for managing and operating data servers, storage systems, system configurations, and network infrastructure.

In traditional configuration management practices, each minute configuration change required manual action by system administrators and the IT support team. But with IaC, all the configuration details are managed and stored in a standardized file system, wherein the system automatically manages infrastructure changes and deals with system configurations.

Therefore, we do not require most of the manual effort since everything is managed and automated by following the IaC approach. Tools such as Ansible can be used to implement IaC approach

9. What do you mean by ad-hoc commands? Give a specific example.

Ad-hoc commands are simple one-line commands used to perform a certain task. We can think of Adhoc commands as an alternative to writing playbooks. An example of an Adhoc command is as follows:

```
ansible host -m netscaler -a "nsc_host=nsc.example.com user=apiuser password=apipass"
```

The aove command is to disable server.

10. In Ansible, what are the variables?

Variables in Ansible are very similar to variables in any programming language. Just like any other variable, an Ansible variable is assigned a value which is used in computing playbooks. You can also use conditions around the variables.

11. What is the difference between a variable name and a variable that is part of the environment?

Variable name:

- You need to add strings to create variable names
- You can easily create multiple variable names by adding strings
- We use the ipv4 address for variable names

Environment variable:

- You need existing variables to access environment variables.
- To create environment variables we must refer advanced Ansible playbook
- We use {{ ansible_env.SOME_VARIABLE }} for remote environment variables.

12.What are the Ansible Modules, and what do they do? Describe its various types.

Ansible modules are a small set of programs that perform a specific task. Modules can be used to automate a wide range of tasks. Modules in Ansible are considered to be idempotent or in other words, making multiple identical requests has the same effect as making a single request.

There are 2 types of modules in Ansible:

Core modules
Extras modules

Core Modules

These are modules that the core Ansible team maintains and will always ship with Ansible itself. They will also receive a slightly higher priority for all requests than those in the “extras” repos. The source of these modules is hosted by Ansible on GitHub in the Ansible-modules-core.

Extras Modules

These modules are currently shipped with Ansible but might be shipped separately in the future. They are also mostly maintained by the Ansible Community. Non-core modules are still fully usable but may receive slightly lower response rates for issues and pull requests.

Popular “extras” modules may be promoted to core modules over time. The source for these modules is hosted by Ansible on GitHub in the Ansible-modules-extras.

13. What is an Ansible Task, exactly?

Ansible Tasks allow you to break up bits of configuration policy into smaller files. These are blocks of code that can be used to automate any process. For example, if you wish to install a package or update a software, you can follow the below code snippet:

Install <package_name>, update <software_name>