Module 3: Ansible Modules and Roles

Demo Document - 1

edureka!

Demo – 1: Ansible Ad-Hoc Commands

Problem Statement:

PART 1 - Configure Ansible on CentOS 7

Solution Steps:

1. To install Ansible for CentOS 7, first ensure that the **CentOS 7** is updated with latest packages and **EPEL** repository is installed as shown below.

yum update -y

```
[root@ip-172-31-18-124 ~]# yum update -y
Loaded plugins: fastestmirror
Determining fastest mirrors
 * base: d36uatko69830t.cloudfront.net
 * extras: d36uatko69830t.cloudfront.net
 * updates: d36uatko69830t.cloudfront.net
base
extras
updates
(1/4): base/7/x86_64/group_gz
(2/4): extras/7/x86_64/primary_db
(3/4): updates/7/x86_64/primary_db
(4/4): base/7/x86_64/primary_db
Resolving Dependencies
--> Running transaction check
---> Package bind-export-libs.x86_64 32:9.11.4-16.P2.el7_8.2 will be updated
```

yum install epel-release -y

```
[root@ip-172-31-18-124 ~]# yum install epel-release -y
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: d36uatko69830t.cloudfront.net
 * extras: d36uatko69830t.cloudfront.net
 * updates: d36uatko69830t.cloudfront.net
Resolving Dependencies
--> Running transaction check
---> Package epel-release.noarch 0:7-11 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                          Arch                              Version
================================================================================
Installing:
 epel-release                     noarch                            7-11

Transaction Summary
================================================================================
```

2. Once the repository is installed, install Ansible with Yum using below command..

   yum install ansible -y

```
[root@ip-172-31-18-124 ~]# yum install ansible -y
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
epel/x86_64/metalink
 * base: d36uatko69830t.cloudfront.net
 * epel: mirror.steadfastnet.com
 * extras: d36uatko69830t.cloudfront.net
 * updates: d36uatko69830t.cloudfront.net
epel
(1/3): epel/x86_64/group_gz
(2/3): epel/x86_64/updateinfo
(3/3): epel/x86_64/primary_db
Resolving Dependencies
--> Running transaction check
---> Package ansible.noarch 0:2.9.21-1.el7 will be installed
--> Processing Dependency: python-httplib2 for package: ansible-2.9.21-1.el7.noarch
```

3. Once the installation is completed, check the ansible version as shown below..

```
[root@ip-172-31-18-124 ~]# ansible --version
ansible 2.9.21
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/module
  ansible python module location = /usr/lib/python2.7/site-package
  executable location = /bin/ansible
  python version = 2.7.5 (default, Nov 16 2020, 22:23:17) [GCC 4.8
[root@ip-172-31-18-124 ~]#
```

4. Ansible keeps track of all the servers that it knows about through a "hosts" file. We need to set up this file first before we can begin to communicate with other computers.
   Open the file with root privileges like this:

   sudo vi /etc/ansible/hosts

5. IP addresses used here in this demo is 172.31.25.35. To configure this, you would add this block to your hosts file: **/etc/ansible/hosts** as shown below.

```
#   - 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

# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10
localhost
172.31.25.35
```

Problem Statement:

PART 2 - Run simple ansible modules.

NOTE: You should have enabled Password less authentication with remote servers before running the below ansible modules.

1. Ping all the servers you have configured by typing as shown below.

   ansible -m ping all

   Ansible will return output like this:

```
[root@ip-172-31-18-124 ~]# ansible -m ping all
172.31.25.35 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[root@ip-172-31-18-124 ~]#
```

   This is a basic test to make sure that Ansible has a connection to all its hosts..

   The -m ping of the command is an instruction to Ansible to use the "ping" module.These are basically commands that you can run on your remote hosts..

The ping module operates in many ways like the normal ping utility in Linux, but instead it checks for Ansible connectivity

The all portion means "**all hosts".** You could just as easily specify a group:

2. The shell module lets us send a terminal command to the remote host and retrieve the results. For instance, to find out the memory usage on our remote machine, we could use:

ansible -m shell -a 'free -m' all

As you can see, you pass arguments into a script by using the -a switch. Here is what the output might look like:

```
[root@ip-172-31-18-124 ~]# ansible -m shell -a 'free -m' all
172.31.25.35 | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:            985         111         746          12         126         734
Swap:             0           0           0
[root@ip-172-31-18-124 ~]#
```