Day 15 – Ansible examples - playbooks, inventory

Friday, December 29, 2023 2:17 PM

Adding the host key (pem) to the control node: (Where ansible runs)

```
ubuntu@ip-172-31-13-84:~$ vim ec2_key.pem
ubuntu@ip-172-31-13-84:~$ chmod 400 ec2_key.pem
ubuntu@ip-172-31-13-84:~$ ssh-agent bash
ubuntu@ip-172-31-13-84:~$ cp ec2_key.pem ~/.ssh/
ubuntu@ip-172-31-13-84:~$ ssh-add ~/.ssh/ec2_key.pem
Identity added: /home/ubuntu/.ssh/ec2_key.pem (/home/ubuntu/.ssh/ec2_key.pem)
ubuntu@ip-172-31-13-84:~$
```

Default location of Inventory file : **/etc/ansible/hosts**

Ping all resources listed in default inventory

```
buntu@ip-172-31-13-84:~$ ansible all -m ping
```

Create inventory file anywhere that can be used to add hosts, databases

```
ubuntu@ip-172-31-13-84:~$ cat inventory.txt
[webservers]
server1 ansible_host=3.104.121.36
server2 ansible_host=54.252.185.143
```

Ping the webservers, listed in inventory.txt

```
ubuntu@ip-172-31-13-84:~$ ansible webservers -m ping -i inventory.txt
```

Ad-hoc commands:

Running ansible commands to be executed on the host machines (webservers) in the CLI.
The webservers are listed in the inventory:
The target can be: **all** services listed, **webservers** or individual server identified by names: (server1, server2 etc...)

```
ubuntu@ip-172-31-13-84:~$ ansible webservers -i inventory.txt -a "touch hello.txt"
```

```
buntu@ip-172-31-13-84:~$ ansible webservers -i inventory.txt -a "mv hello.txt hi.txt"
```

```
buntu@ip-172-31-13-84:~$ ansible server1 -i inventory.txt -a "mv hi.txt server1.txt"
```

Ansible Playbooks examples:

1. Create file in $HOME folder of all the hosts

```
ubuntu@ip-172-31-13-84:~$ cat first_ansible.yml
 - hosts: webservers
   tasks:
     - name: create a file in home directory
       command: /bin/touch ~/ansible.txt
```

Run ansible script: **$ansible-playbook first_ansible.yml -i inventory.txt**

2. Install jdk-17 in all the hosts (webservers)

```
ubuntu@ip-172-31-13-84:~$ cat install_jdk_ansible.yml
- name: playbook to install jdk-17
  hosts: webservers
  tasks:
    - name: update apt
      command: sudo apt update


    - name: update apt-get
      command: sudo apt-get update

    - name: install java using ansible
      become: yes
      apt:
        name: "{{packages}}"
        state: present
      vars:
        packages:
           - openjdk-17-jdk
```

Run ansible script: **$ansible-playbook install_jdk_ansible.yml -i inventory.txt**

3. Download jar file from S3, and deploy springboot application

```
ubuntu@ip-172-31-13-84:~$ cat deploy_springboot_ansible.yml
- name: download and rename springboot jar file fomr S3
  hosts: webservers
  tasks:
    - name: download exec jar from AWS S3
      get_url:
        url: https://vatsabucketdemo.s3.amazonaws.com/employee-springboot-demo-1.0.jar
        dest: ~/

    - name: rename springboot jar file
      command: /bin/mv employee-springboot-demo-1.0.jar springboot.jar


- name: run and deploy springboot application
  hosts: webservers
  tasks:
    - name: deploy the springboot application
      command: /bin/java -jar springboot.jar
```

Run ansible script: **$ansible-playbook deploy_springboot_ansible.yml -i inventory.txt**

4. Create EC-2 instances in AWS.
    a. Install Python3, Boto in the instance
    b. Copy the Access & Secret keys of the IAM user.

```
ubuntu@ip-172-31-13-84:~$ cat create_ec2_ansible.yml
- name: create EC2 instance with Ansible
  hosts: localhost
  tasks:
  - name: Launch EC2 instance using Ansible
    ec2:
      key_name: aws_ansible_instance
      image: ami-04f5097681773b989
      instance_type: t2.micro
      region: ap-southeast-2
      count: 1
      vpc_subnet_id: default
      assign_public_ip: yes
      aws_access_key: [redacted]
      aws_secret_key: [redacted]
```