Lab Exercise 01

Creating Static Host Inventory

Objective: To create a static host inventory for managing and automating infrastructure tasks efficiently across multiple servers using Ansible

Tools required: Ubuntu OS

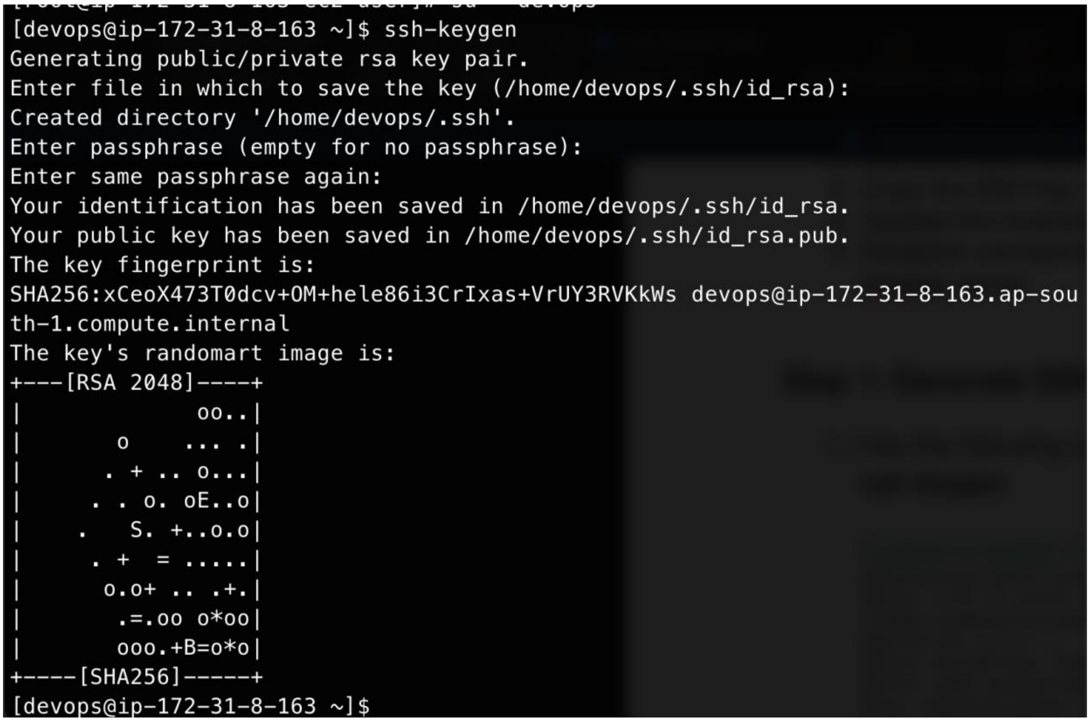
Prerequisites: You need to have Ansible installed to proceed with this demo

Steps to be followed:

1. Generate SSH key pair on the main node
2. Copy the SSH key to the two other nodes
3. Update the inventory or host file with the host IP address
4. Establish connectivity between the hosts specified in the host file and the Ansible server

Step 1: Generate SSH key pair on the main node

1.1 Use the following command to generate the SSH key on the Ansible server:
ssh-keygen

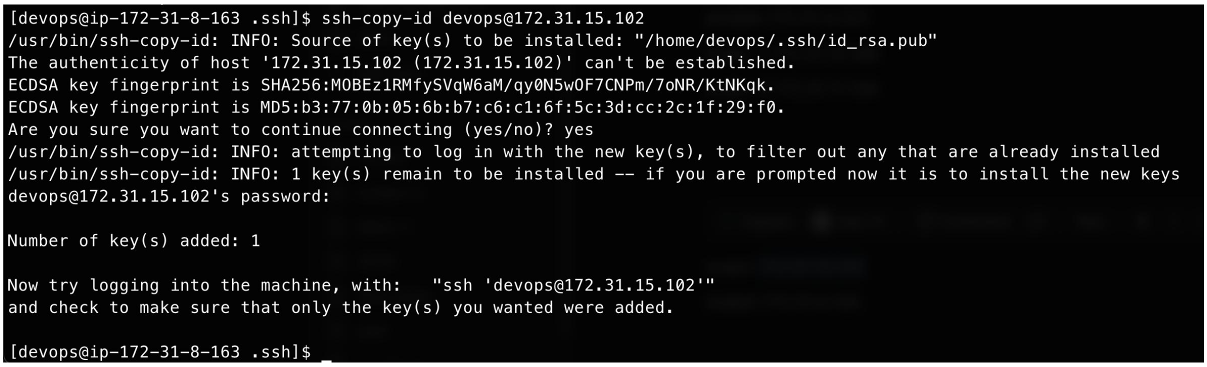


```
[root@ip-172-31-8-163 ec2-user]# su - devops
[devops@ip-172-31-8-163 ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/devops/.ssh/id_rsa):
Created directory '/home/devops/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/devops/.ssh/id_rsa.
Your public key has been saved in /home/devops/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:xCeoX473T0dcv+OM+hele86i3CrIxas+VrUY3RVKkWs devops@ip-172-31-8-163.ap-sou
th-1.compute.internal
The key's randomart image is:
+---[RSA 2048]----+
|             oo..|
|       o    ... .|
|      . + .. o...|
|     . . o. oE..o|
|    .   S. +..o.o|
|     . +  = .....|
|      o.o+ .. .+.|
|       .=.oo o*oo|
|       ooo.+B=o*o|
+----[SHA256]-----+
[devops@ip-172-31-8-163 ~]$
```

Step 2: Copy the SSH key to the other two nodes

2.1 Use the following command to copy the public key to a file named **authorized_keys** in localhost:

cat .ssh/id_rsa.pub >> .ssh/authorized_keys



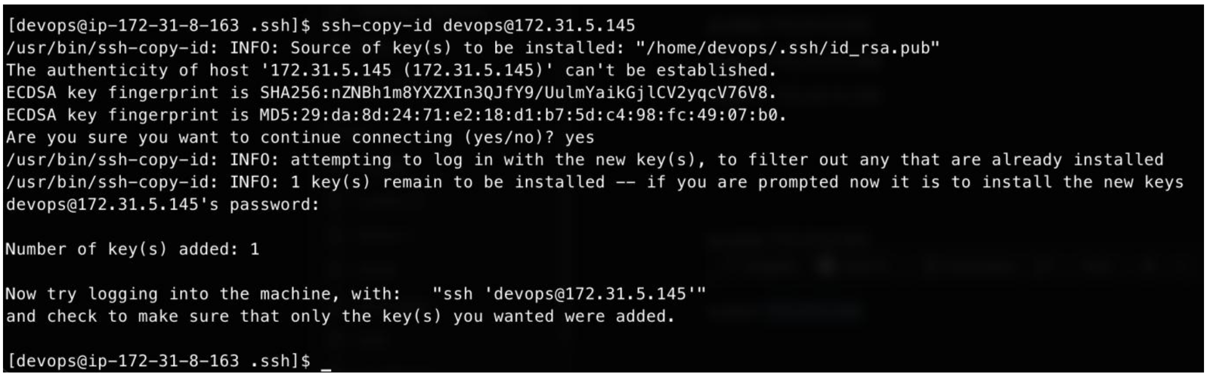
```
[devops@ip-172-31-8-163 .ssh]$ ssh-copy-id devops@172.31.15.102
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/devops/.ssh/id_rsa.pub"
The authenticity of host '172.31.15.102 (172.31.15.102)' can't be established.
ECDSA key fingerprint is SHA256:MOBEz1RMfySVqW6aM/qy0N5wOF7CNPm/7oNR/KtNKqk.
ECDSA key fingerprint is MD5:b3:77:0b:05:6b:b7:c6:c1:6f:5c:3d:cc:2c:1f:29:f0.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
devops@172.31.15.102's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'devops@172.31.15.102'"
and check to make sure that only the key(s) you wanted were added.

[devops@ip-172-31-8-163 .ssh]$ _
```

2.2 Run the following command to go to the **.ssh** directory of the Ansible server:

cd .ssh

2.3 Run the following command to copy the public key to another node that will connect to the Ansible server:

ssh-copy-id username@ip -p 22

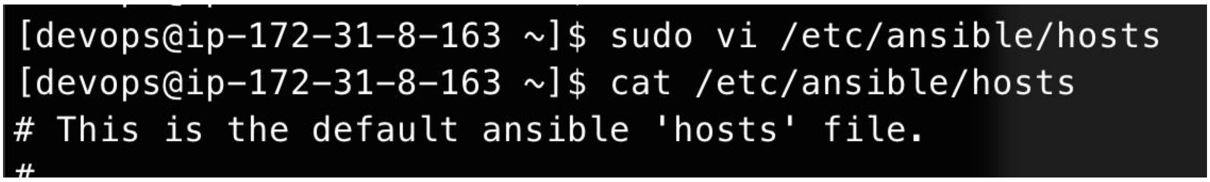


```
[devops@ip-172-31-8-163 .ssh]$ ssh-copy-id devops@172.31.5.145
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/devops/.ssh/id_rsa.pub"
The authenticity of host '172.31.5.145 (172.31.5.145)' can't be established.
ECDSA key fingerprint is SHA256:nZNBh1m8YXZXIn3QJfY9/UulmYaikGjlCV2yqcV76V8.
ECDSA key fingerprint is MD5:29:da:8d:24:71:e2:18:d1:b7:5d:c4:98:fc:49:07:b0.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
devops@172.31.5.145's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'devops@172.31.5.145'"
and check to make sure that only the key(s) you wanted were added.

[devops@ip-172-31-8-163 .ssh]$ _
```

2.4 Execute the following command to exit the **.ssh** directory of the Ansible server:

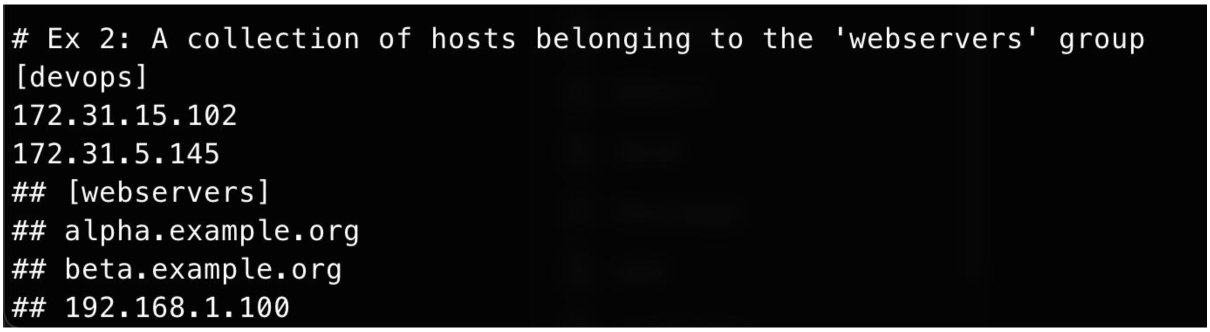
Step 3: Update the inventory or host file with the host IP address

3.1 Use the following command to open the Ansible inventory file and add the host localhost to it:
sudo vi/etc/ansible/hosts



```
[devops@ip-172-31-8-163 ~]$ sudo vi /etc/ansible/hosts
[devops@ip-172-31-8-163 ~]$ cat /etc/ansible/hosts
# This is the default ansible 'hosts' file.
#
```

3.2 When the file opens, add the three lines of code below to the end of the file:
[dbbservers]
localhost:22
172.31.5.76:22

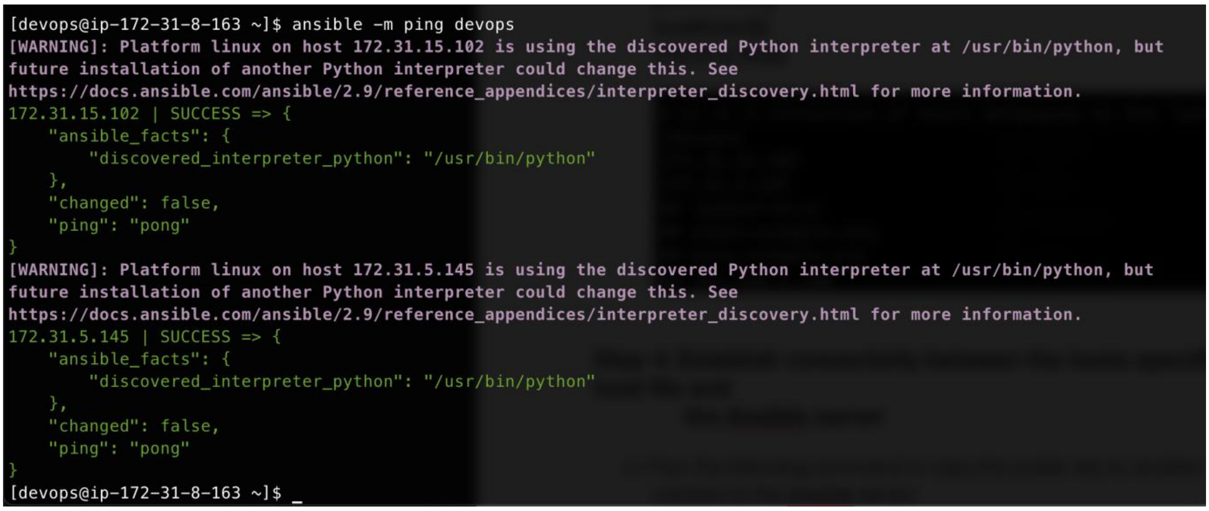


```
# Ex 2: A collection of hosts belonging to the 'webservers' group
[devops]
172.31.15.102
172.31.5.145
## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
```

Step 4: Establish connectivity between the hosts specified in the host file and
the Ansible server

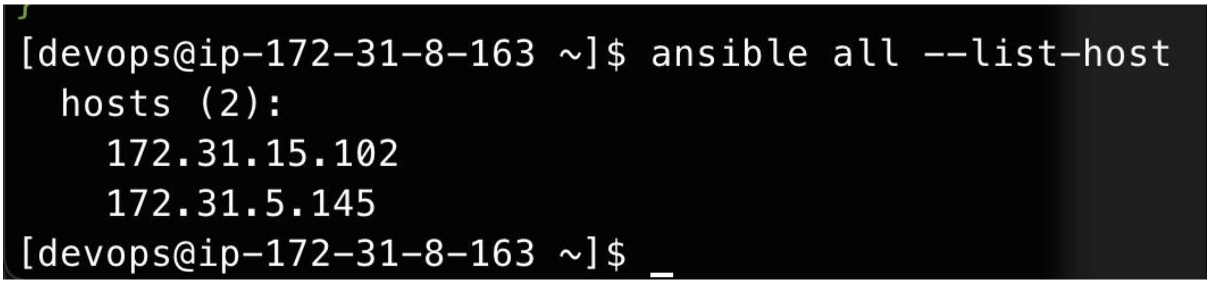
4.1 Run the following command to copy the public key to another node that will connect to the Ansible server:
ansible -m ping dbbservers

Note: Press **esc**, then write **:wq** and press **enter** to save the file.



```
[devops@ip-172-31-8-163 ~]$ ansible -m ping devops
[WARNING]: Platform linux on host 172.31.15.102 is using the discovered Python interpreter at /usr/bin/python, but
future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
172.31.15.102 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Platform linux on host 172.31.5.145 is using the discovered Python interpreter at /usr/bin/python, but
future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
172.31.5.145 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[devops@ip-172-31-8-163 ~]$ _
```

4.2 Use the following command to check the number of hosts in the host file:
ansible all --list-hosts



```
}
[devops@ip-172-31-8-163 ~]$ ansible all --list-host
  hosts (2):
    172.31.15.102
    172.31.5.145
[devops@ip-172-31-8-163 ~]$ _
```

By following these steps, you have successfully created a static host inventory for managing and automating infrastructure tasks efficiently across multiple servers using Ansible.