Lesson 02 Demo 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

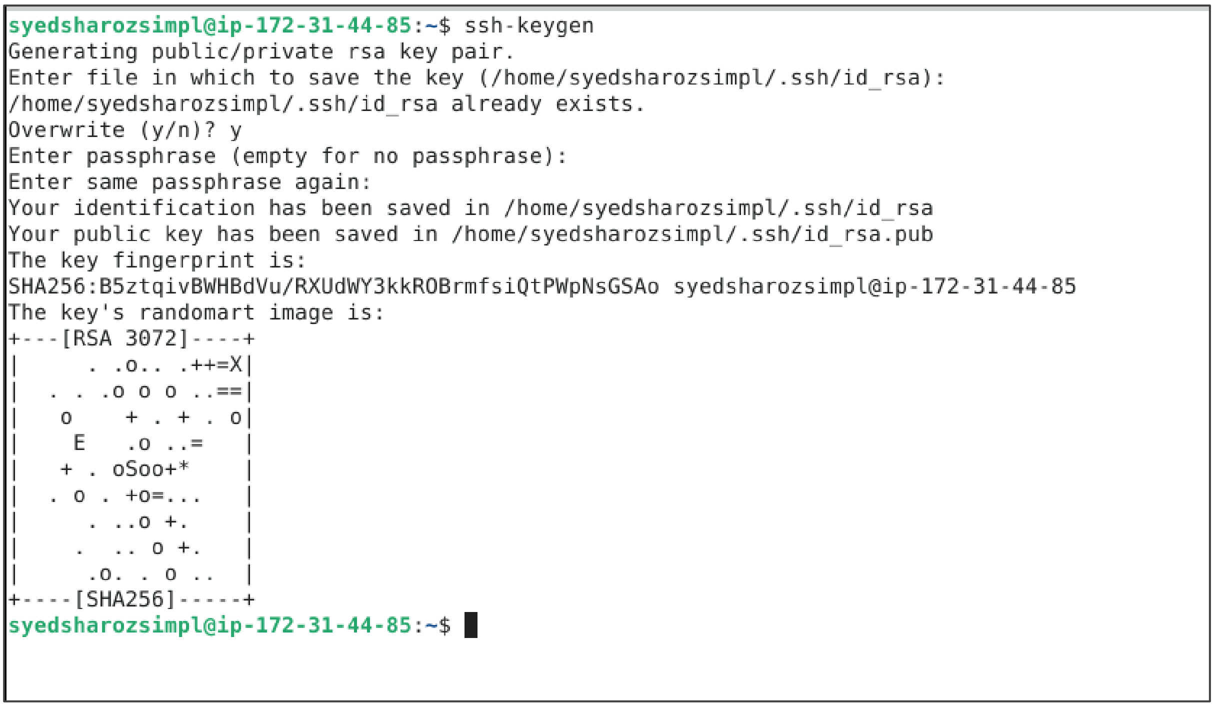
Prerequisites: You need to have Ansible installed to proceed with this demo

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

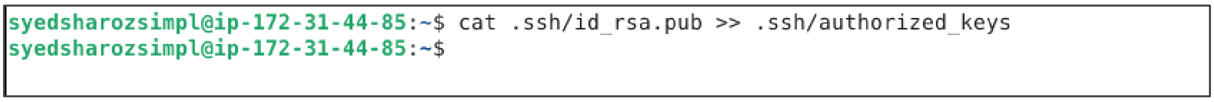


```
syedsharozsimpl@ip-172-31-44-85:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/syedsharozsimpl/.ssh/id_rsa):
/home/syedsharozsimpl/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/syedsharozsimpl/.ssh/id_rsa
Your public key has been saved in /home/syedsharozsimpl/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:B5ztqivBWHBdVu/RXUdWY3kkROBrmfsiQtPWpNsGSAo syedsharozsimpl@ip-172-31-44-85
The key's randomart image is:
+---[RSA 3072]----+
|     . .o.. .++=X|
|  . . .o o o ..==|
|   o    + . + . o|
|    E   .o ..=   |
|   + . oSoo+*    |
|  . o . +o=...   |
|     . ..o +.    |
|    .  .. o +.   |
|     .o. . o ..  |
+----[SHA256]-----+
syedsharozsimpl@ip-172-31-44-85:~$ 
```

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:

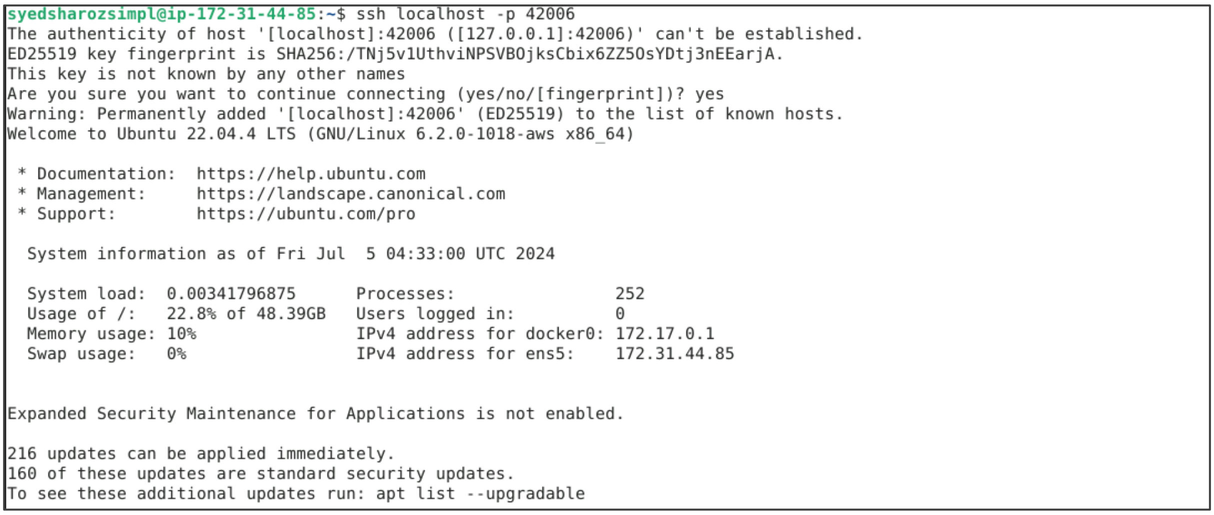
cat .ssh/id_rsa.pub >> .ssh/authorized_keys



```
syedsharozsimpl@ip-172-31-44-85:~$ cat .ssh/id_rsa.pub >> .ssh/authorized_keys
syedsharozsimpl@ip-172-31-44-85:~$
```

2.2 Use the following command to check the SSH connection with the localhost:

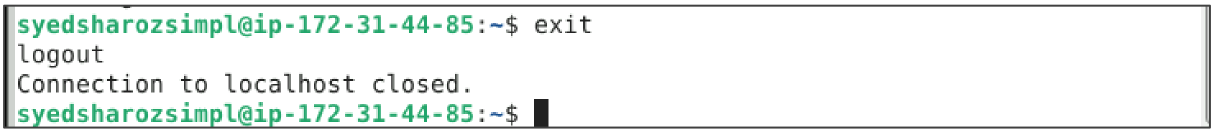
ssh localhost -p 42006



```
syedsharozsimpl@ip-172-31-44-85:~$ ssh localhost -p 42006
The authenticity of host '[localhost]:42006 ([127.0.0.1]:42006)' can't be established.
ED25519 key fingerprint is SHA256:/TNj5v1UthviNPSVBOjksCbix6ZZ5OsYDtj3nEEarjA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[localhost]:42006' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.2.0-1018-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

  System information as of Fri Jul  5 04:33:00 UTC 2024

  System load:  0.00341796875      Processes:                252
  Usage of /:   22.8% of 48.39GB   Users logged in:          0
  Memory usage: 10%                IPv4 address for docker0: 172.17.0.1
  Swap usage:   0%                 IPv4 address for ens5:    172.31.44.85


Expanded Security Maintenance for Applications is not enabled.

216 updates can be applied immediately.
160 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable
```

2.3 Now, use the following command to exit from the localhost:

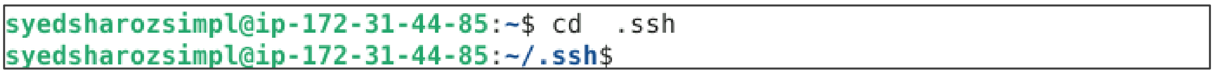
exit



```
syedsharozsimpl@ip-172-31-44-85:~$ exit
logout
Connection to localhost closed.
syedsharozsimpl@ip-172-31-44-85:~$
```

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

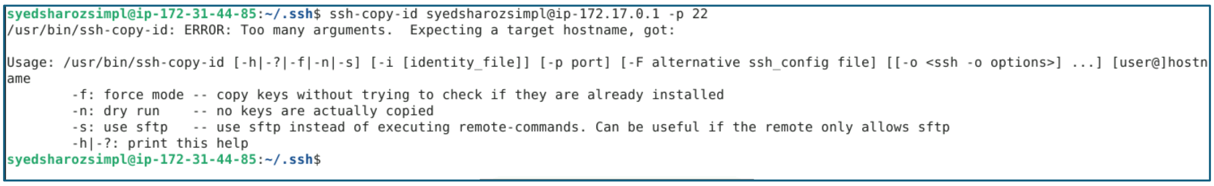
cd .ssh



```
syedsharozsimpl@ip-172-31-44-85:~$ cd  .ssh
syedsharozsimpl@ip-172-31-44-85:~/.ssh$
```

2.5 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

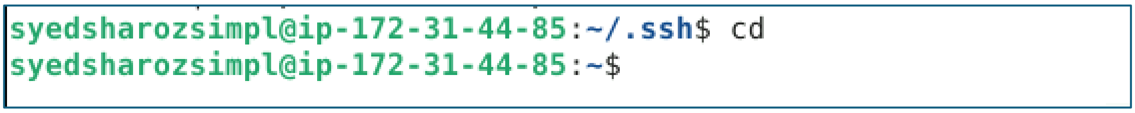


```
syedsharozsimpl@ip-172-31-44-85:~/.ssh$ ssh-copy-id syedsharozsimpl@ip-172.17.0.1 -p 22
/usr/bin/ssh-copy-id: ERROR: Too many arguments.  Expecting a target hostname, got:

Usage: /usr/bin/ssh-copy-id [-h|-?|-f|-n|-s] [-i [identity_file]] [-p port] [-F alternative ssh_config file] [[-o <ssh -o options>] ...] [user@]hostn
ame
        -f: force mode -- copy keys without trying to check if they are already installed
        -n: dry run    -- no keys are actually copied
        -s: use sftp   -- use sftp instead of executing remote-commands. Can be useful if the remote only allows sftp
        -h|-?: print this help
syedsharozsimpl@ip-172-31-44-85:~/.ssh$
```

Note: You must use a **username@ip** with your node and IP username, which are provided in the lab credential.

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

cd

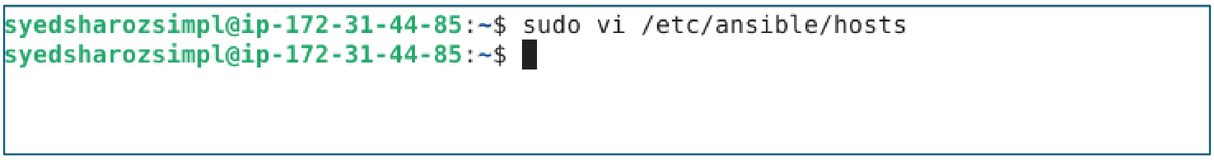


```
syedsharozsimpl@ip-172-31-44-85:~/.ssh$ cd
syedsharozsimpl@ip-172-31-44-85:~$
```

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



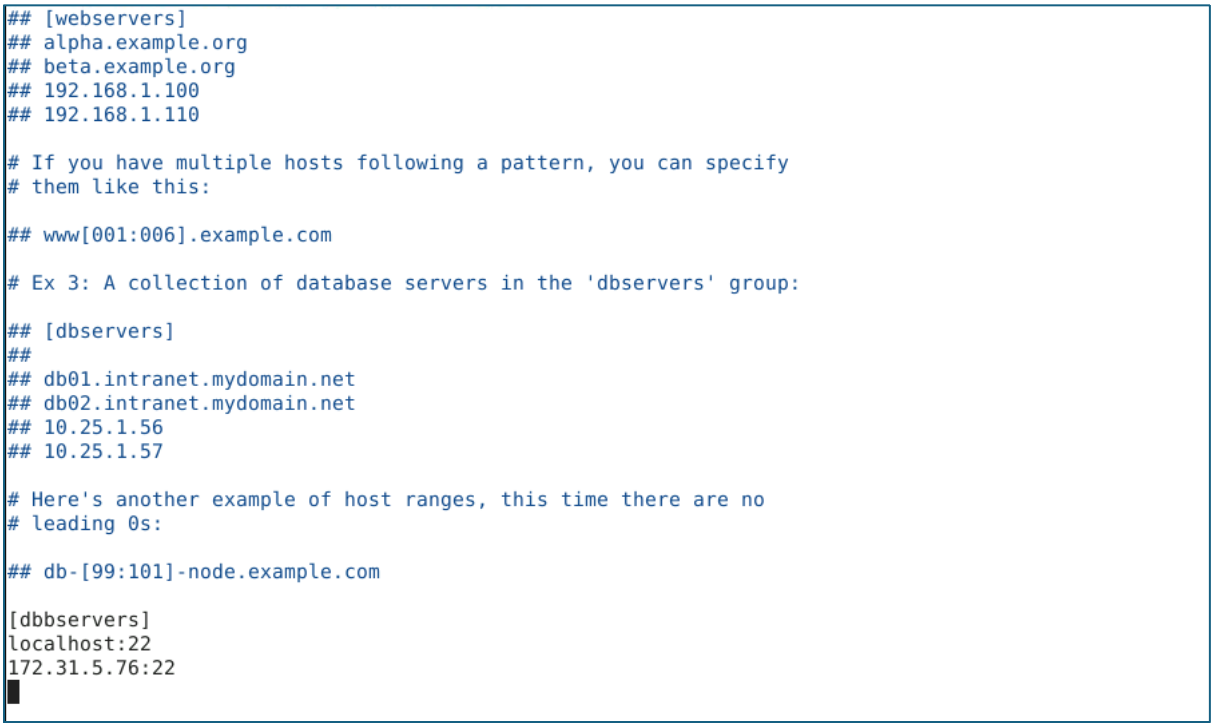
```
syedsharozsimpl@ip-172-31-44-85:~$ sudo vi /etc/ansible/hosts
syedsharozsimpl@ip-172-31-44-85:~$
```

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



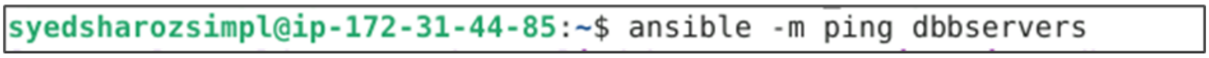
```
## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern, you can specify
# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group:

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Here's another example of host ranges, this time there are no
# leading 0s:

## db-[99:101]-node.example.com

[dbbservers]
localhost:22
172.31.5.76:22
```

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

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

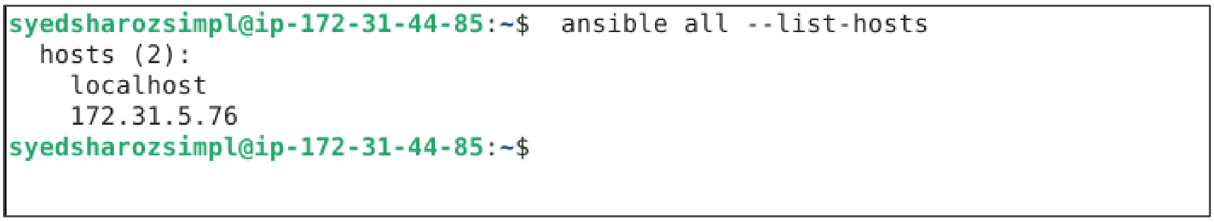
4.1 Run the following command to copy the public key to another node that will connect to the Ansible server:

ansible -m ping dbbservers



```
syedsharozsimpl@ip-172-31-44-85:~$ ansible -m ping dbbservers
```

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



```
syedsharozsimpl@ip-172-31-44-85:~$  ansible all --list-hosts
  hosts (2):
    localhost
    172.31.5.76
syedsharozsimpl@ip-172-31-44-85:~$
```

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