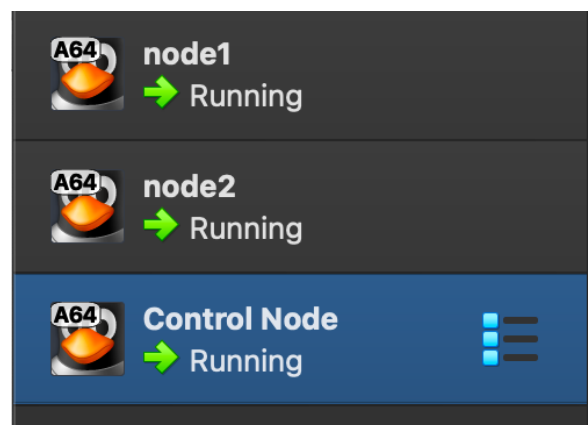


PROJECT INFRASTRUCTURE AS A CODE ANSIBLE

In this project, i'm setting up three servers in VirtualBox : one will act as my **Ansible control node**, and the other two will be my **managed nodes**. Once the infrastructure is ready, I'll leverage Ansible to **automate the Nginx installation** across my managed servers. This will be a great way to solidify my understanding of Ansible for infrastructure as code.

- A. Create 3 Servers in Virtual box and Set Network to Bridge Adapter, install SSH and enable SSH :



- Enable SSH and check if SSH is running on all VMs :

```
node1 [Running]
root@server1:/home/dyno# sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-06-03 12:27:57 UTC; 48s ago
   TriggeredBy: ● ssh.socket
   Docs: man:sshd(8)
         man:sshd_config(5)
   Main PID: 1284 (sshd)
   Tasks: 1 (limit: 2060)
   Memory: 2.1M (peak: 2.4M)
   CPU: 14ms
   CGroup: /system.slice/ssh.service
           └─1284 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Jun 03 12:27:57 server1 systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Jun 03 12:27:57 server1 sshd[1284]: Server listening on :: port 22.
Jun 03 12:27:57 server1 systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
root@server1:/home/dyno#

node2 [Running]
root@server2:/home/dyno# sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-06-03 12:30:54 UTC; 1min 50s ago
   TriggeredBy: ● ssh.socket
   Docs: man:sshd(8)
         man:sshd_config(5)
   Process: 783 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 788 (sshd)
   Tasks: 1 (limit: 2060)
   Memory: 2.1M (peak: 2.4M)
   CPU: 14ms
   CGroup: /system.slice/ssh.service
           └─788 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Jun 03 12:30:54 server2 systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Jun 03 12:30:54 server2 sshd[788]: Server listening on :: port 22.
Jun 03 12:30:54 server2 systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
root@server2:/home/dyno#

Control Node [Running]
root@controlnode:/home/controlnode# sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-06-03 12:36:58 UTC; 22s ago
   TriggeredBy: ● ssh.socket
   Docs: man:sshd(8)
         man:sshd_config(5)
   Main PID: 1821 (sshd)
   Tasks: 1 (limit: 2060)
   Memory: 1.1M (peak: 1.4M)
   CPU: 9ms
   CGroup: /system.slice/ssh.service
           └─1821 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Jun 03 12:36:58 controlnode systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Jun 03 12:36:58 controlnode sshd[1821]: Server listening on :: port 22.
Jun 03 12:36:58 controlnode systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
root@controlnode:/home/controlnode#
```

- Installing Ansible in Control Node

```
Control Node [Running]
Selecting previously unselected package python3-libcloud.
Preparing to unpack .../68-python3-libcloud_3.4.1-5_all.deb ...
Unpacking python3-libcloud (3.4.1-5) ...
Selecting previously unselected package python3-ntlm-auth.
Preparing to unpack .../69-python3-ntlm-auth_1.5.0-1_all.deb ...
Unpacking python3-ntlm-auth (1.5.0-1) ...
Selecting previously unselected package python3-requests-ntlm.
Preparing to unpack .../70-python3-requests-ntlm_1.1.0-3_all.deb ...
Unpacking python3-requests-ntlm (1.1.0-3) ...
Selecting previously unselected package python3-selinux.
Preparing to unpack .../71-python3-selinux_3.5-2ubuntu2.1_arm64.deb ...
Unpacking python3-selinux (3.5-2ubuntu2.1) ...
Selecting previously unselected package python3-xmltodict.
Preparing to unpack .../72-python3-xmltodict_0.13.0-1_all.deb ...
Unpacking python3-xmltodict (0.13.0-1) ...
Selecting previously unselected package python3-wiresh.
Preparing to unpack .../73-python3-wiresh_0.4.3-2_all.deb ...
Unpacking python3-wiresh (0.4.3-2) ...
Selecting previously unselected package python3-passlib.
Preparing to unpack .../74-python3-passlib_1.7.4-4_all.deb ...
Unpacking python3-passlib (1.7.4-4) ...
Setting up python3-lockfile (1:0.12.2-3) ...
Setting up python3-passlib (1.7.4-4) ...
Setting up python3-ntlm-auth (1.5.0-1) ...
Setting up python3-resolveip (1.0.1-1) ...
Setting up python3-kerberos (1.1.14-3build9) ...
Setting up python3-simplejson (3.19.2-1build2) ...
Setting up python3-xmltodict (0.13.0-1) ...
Setting up python3-dnspython (2.6.1-1ubuntu1) ...
Setting up python3-selinux (3.5-2ubuntu2.1) ...
Setting up python3-requests-ntlm (1.1.0-3) ...
Setting up python3-requests-ntlm (1.1.0-3) ...
Setting up python3-libcloud (3.4.1-5) ...
Setting up python3-wiresh (0.4.3-2) ...
Setting up ansible (5.2.0+dfsg-1ubuntu2) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@controlnode:/home/controlnode#
```

- Create ssh keygen and copy ssh into Node1 and Node2

```
Control Node [Running]
controlnode@controlnode:~$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/controlnode/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/controlnode/.ssh/id_rsa
Your public key has been saved in /home/controlnode/.ssh/id_rsa.pub
The key's fingerprint is:
SHA256:1GMCJUMnMeiU0wL28Ch1x/wupAqM7pNoJDTXRd3xCU controlnode@controlnode
The key's randomart image is:
+---[RSA 4096]-----+
|  .o    ...oE..    |
|.ooB... .o ...    |
|==+==+.+=         |
|&=00= o+ .        |
|&B.  o .S         |
|+0.+ .            |
|+=               |
|...              |
+---[SHA256]-----+
controlnode@controlnode:~$
```

```
Control Node [Running]
controlnode@controlnode:~$ ssh-copy-id dyno@172.20.10.2
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/controlnode/.ssh/id_rsa.pub"
The authenticity of host '172.20.10.2 (172.20.10.2)' can't be established.
ED25519 key fingerprint is SHA256:2RwYvWYhxp53UJEK77CHF4Ar/Hd1XImb0IdaVXoHFXs.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? 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
dyno@172.20.10.2's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'dyno@172.20.10.2'"
and check to make sure that only the key(s) you wanted were added.
controlnode@controlnode:~$
```

```
Control Node [Running]
controlnode@controlnode:~$ ssh-copy-id dyno@172.20.10.3
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/controlnode/.ssh/id_rsa.pub"
The authenticity of host '172.20.10.3 (172.20.10.3)' can't be established.
ED25519 key fingerprint is SHA256:2nuPQqNBXc6eTpNuMq7/Vhf0sPJY4xekun3EMbPybs.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? 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
dyno@172.20.10.3's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'dyno@172.20.10.3'"
and check to make sure that only the key(s) you wanted were added.
controlnode@controlnode:~$
```

- Check if Ansible has been installed or not :

```
Control Node [Running]
controlnode@controlnode:~$ ansible --version
ansible [core 2.16.3]
  config file = None
  configured module search path = ['/home/controlnode/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/controlnode/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Feb 4 2025, 14:48:35) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
controlnode@controlnode:~$ _
```

- Create file inventory.ini and fill with configuration ini :

```
Control Node [Running]
GNU nano 7.2 /home/controlnode/ansible/inventory.ini *
[webservers]
172.20.10.2
172.20.10.3

[all:vars]
ansible_user=dyno
ansible_ssh_private_key_file=~/.ssh/id_rsa
_
```

- Check if Ansible can connect to Node1 dan Node2 :

```
Control Node [Running]
controlnode@controlnode:~$ ansible all -i ~/ansible/inventory.ini -m ping
172.20.10.3 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
172.20.10.2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
controlnode@controlnode:~$ _
```

- Create install-nginx.yml file for the commands that need to be executed with .yml

```
Control Node [Running]
GNU nano 7.2 /home/controlnode/ansible/install-nginx.yml
- name: Install nginx di webservers
  hosts: webservers
  become: yes
  tasks:
    - name: Update apt cache
      apt:
        update_cache: yes
    - name: Install nginx
      apt:
        name: nginx
        state: present
```

- Fill in the configuration with Sudo visudo on Node1 and Node2 to allow root access without a password :

```
node1 [Running]
GNU nano 7.2 /etc/sudoers.tmp *
Defaults    use_pty

# This preserves proxy settings from user environments of root
# equivalent users (group sudo)
#Defaults:sudo env_keep += "http_proxy https_proxy ftp_proxy all_proxy no_proxy"

# This allows running arbitrary commands, but so does ALL, and it means
# different sudoers have their choice of editor respected.
#Defaults:sudo env_keep += "EDITOR"

# Completely harmless preservation of a user preference.
#Defaults:sudo env_keep += "GREP_COLOR"

# While you shouldn't normally run git as root, you need to with etckeeper
#Defaults:sudo env_keep += "GIT_AUTHOR_* GIT_COMMITTER_*"

# Per-user preferences; root won't have sensible values for them.
#Defaults:sudo env_keep += "EMAIL DEBEMAIL DEBFULLNAME"

# "sudo scp" or "sudo rsync" should be able to use your SSH agent.
#Defaults:sudo env_keep += "SSH_AGENT_PID SSH_AUTH_SOCK"

# Ditto for GPG agent
#Defaults:sudo env_keep += "GPG_AGENT_INFO"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "include" directives:

#includedir /etc/sudoers.d

dunyo   ALL=(ALL) NOPASSWD:ALL_
```

- Run the Ansible Playbook and check if nginx is installed on Node1 and Node2 (Successful ✓✓)

```
Control Node [Running]
controlnode@controlnode:~/ansible$ ansible-playbook -i inventory.ini install-nginx.yml

PLAY [Install nginx on web servers] *****

TASK [Gathering Facts] *****
ok: [172.20.10.3]
ok: [172.20.10.2]

TASK [Update apt cache] *****
changed: [172.20.10.3]
changed: [172.20.10.2]

TASK [Install nginx] *****
changed: [172.20.10.2]
changed: [172.20.10.3]

PLAY RECAP *****
172.20.10.2 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
172.20.10.3 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

controlnode@controlnode:~/ansible$ ansible web servers -i inventory.ini -a "curl -I http://localhost"
172.20.10.3 | CHANGED | rc=0 >>
HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Tue, 03 Jun 2025 13:38:56 GMT
Content-Type: text/html
Content-Length: 615
Last-Modified: Tue, 03 Jun 2025 13:34:01 GMT
Connection: keep-alive
ETag: "689ef9c9-267"
Accept-Ranges: bytes
  bytes  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
   Dload  Upload  Total    Spent    Left  Speed
  0   615    0     0    0     0      0      0      0      0  0 --:--:-- --:--:-- --:--:--    0
172.20.10.2 | CHANGED | rc=0 >>
HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Tue, 03 Jun 2025 13:38:56 GMT
Content-Type: text/html
Content-Length: 615
Last-Modified: Tue, 03 Jun 2025 13:34:00 GMT
Connection: keep-alive
ETag: "689ef9c8-267"
Accept-Ranges: bytes
  bytes  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
   Dload  Upload  Total    Spent    Left  Speed
  0   615    0     0    0     0      0      0      0      0  0 --:--:-- --:--:-- --:--:--    0
controlnode@controlnode:~/ansible$
```

Thank You